

**Membuat Game Berbasis
Website Menggunakan Bahasa
Javascript dan PHP**

Membuat Game Berbasis Website Menggunakan Bahasa Javascript dan PHP

**Dodik Arwin Dermawan, S.ST., S.T., M.T.
Chamdan Mashuri, S.Kom., M.Kom.
Ginjar Setyo Permadi, S.Kom., M.Kom.
Duta Alif Gunawan
Dini Widiasih S.T**



Membuat Game Berbasis Website Menggunakan Bahasa Javascript dan PHP

© Penerbit Perkumpulan Rumah Cemerlang Indonesia (PRCI)

Penulis:

Dodik Arwin Dermawan, S.ST., S.T., M.T.
Chamdan Mashuri, S.Kom., M.Kom.
Ginanjari Setyo Permadi, S.Kom., M.Kom.
Duta Alif Gunawan
Dini Widiasih S.T

Editor: Duta Alif Gunawan

Cetakan Pertama: Januari 2022

Cover: Rusli

Tata Letak: Tim Kreatif PRCI

Hak Cipta 2022, pada Penulis. Diterbitkan pertama kali oleh:

Perkumpulan Rumah Cemerlang Indonesia
ANGGOTA IKAPI JAWA BARAT
Pondok Karisma Residence Jalan Raflesia VI D.151
Panglayungan, Cipedes Tasikmalaya – 085223186009

Website: www.rcipress.rcipublisher.org

E-mail: rumahcemerlangindonesia@gmail.com

Copyright © 2022 by Perkumpulan Rumah Cemerlang Indonesia
All Right Reserved

- Cet. I –: Perkumpulan Rumah Cemerlang Indonesia, 2022
; 14,8 x 21 cm
ISBN: 978-623-5847-91-7

Hak cipta dilindungi undang-undang
Dilarang memperbanyak buku ini dalam bentuk dan dengan
cara apapun tanpa izin tertulis dari penulis dan penerbit

Undang-undang No.19 Tahun 2002 Tentang
Hak Cipta Pasal 72

Undang-undang No.19 Tahun 2002 Tentang Hak Cipta
Pasal 72

Barang siapa dengan sengaja melanggar dan tanpa hak melakukan perbuatan sebagaimana dimaksud dalam pasal ayat (1) atau pasal 49 ayat (1) dan ayat (2) dipidana dengan pidana penjara masing-masing paling sedikit 1 (satu) bulan dan/atau denda paling sedikit Rp.1.000.000,00 (satu juta rupiah), atau pidana penjara paling lama 7 (tujuh) tahun dan/atau denda paling banyak Rp.5.000.000.000,00 (lima miliar rupiah).

Barang siapa dengan sengaja menyiarkan, memamerkan, mengedarkan, atau menjual kepada umum suatu ciptaan atau barang hasil pelanggaran hak cipta terkait sebagai dimaksud pada ayat (1) dipidana dengan pidana penjara paling lama 5 (lima) tahun dan/atau denda paling banyak Rp.500.000.000,00 (lima ratus juta rupiah).

KATA PENGANTAR

Perkembangan perangkat pendukung permainan berbasis komputer, atau lebih dikenal dengan game menjadikan jenis dari game semakin kompleks dan bervariasi. Demikian pula dengan bahasa pemrograman yang digunakan untuk membangun game. Buku **Membuat Game Berbasis Website Menggunakan Javascript & PHP**, disusun untuk dapat digunakan sebagai referensi dalam membangun game, dengan menggunakan Bahasa pemrograman Javascript dan PHP.

Buku ini menjelaskan langkah - langkah awal dalam membangun game sederhana, diantaranya struktur HTML, struktur Div Class, style CSS, dan struktur Java Script. Penjelasan yang gamblang disertai dengan gambar pendukung, akan memudahkan pembaca untuk memahami dari setiap langkah yang ada di dalam buku ini.

Akhir kata kami berharap buku ini dapat memberi manfaat bagi pembaca. Kami menyadari bahwa buku **Membuat Game Berbasis Website Menggunakan Javascript & PHP** jauh dari kata sempurna. Oleh karenanya saran dan kritik yang membangun sangat kami harapkan.

Penulis

DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI	ii
DAFTAR TABEL.....	iv
DAFTAR GAMBAR.....	v
PENDAHULUAN	1
JavaScript.....	11
Pengertian Javascript.....	11
Kelebihan Javascript.....	12
Kekurangan Javascript	13
Fungsi dan cara kerja javascript terhadap website.....	13
Penggunaan umum pada javascript.....	13
Menuliskan syntax JavaScript pada HTML.....	14
Statement pada Javascript	16
Variabel pada Javascript.....	17
Comment pada Javascript	18
Tipe Data pada Javascript	18
String	18
Assignment Operator pada javascript	23
Operator Perbandingan pada JavaScript	25
Operasi Logika dalam Javascript.....	27
Looping pada Javascript	30
Function.....	31
Variabel Scope.....	35
Mode Strict pada Javascript.....	37
Object pada Browser	49
DOM(Document Object Model).....	59
DOM (Document Object Model) Selector.....	61
DOM (Document Object Model) Event.....	65
Membuat game dengan Bahasa Pemrograman Javascript	71

PHP(Hypertext Preprocessor).....	98
Pengertian PHP	99
Kelebihan Bahasa PHP	99
Kekurangan Bahasa PHP	101
Cara Download dan Instalasi Xampp	101
Struktur Dasar PHP	106
Konstanta dan Variabel dalam PHP	109
Tipe Data dalam Bahasa PHP	111
Kondisi	121
Fungsi dalam PHP	126
Argumen pada Fungsi PHP	127
Data Default Argument pada PHP	130
Fungsi PHP mengembalikan nilai/return	131
Looping/Perulangan pada PHP	132
Membuat Game pada Bahasa Pemrograman PHP	139
DAFTAR PUSTAKA.....	146
GLOSARIUM.....	148
INDEKS	151
BIOGRAFI PENULIS	152

DAFTAR TABEL

Tabel 2.1	Macam macam operasi logika javascript.....	28
Tabel 2.2	Macam macam property dan method yang sering digunakan	53
Tabel 3.1	Aturan penulisan variabel dalam PHP.....	109
Tabel 3.2	Definisi dari macam macam perulangan	133

DAFTAR GAMBAR

Gambar 1.1 Gambar logo C++	4
Gambar 1.2 Gambar logo Java.....	5
Gambar 1.3 Gambar logo C#.....	6
Gambar 1.4 Logo javascript	7
Gambar 1.5 Logo PHP.....	9
Gambar 2.1 Contoh pesan peringatan terhadap user	14
Gambar 2.2 Contoh syntax embedded script.....	15
Gambar 2.3 Contoh syntax external script	15
Gambar 2.4 Contoh script alert	16
Gambar 2.5 Hasil script alert.....	17
Gambar 2.6 Contoh script variabel menggunakan var	17
Gambar 2.7 Contoh script comment	18
Gambar 2.8 Contoh script penggunaan tipe data string	19
Gambar 2.9 Hasil script tipe data string.....	19
Gambar 2.10 Contoh script tipe data number.....	20
Gambar 2.11 Hasil script tipe data number	20
Gambar 2.12 contoh script tipe data array	21
Gambar 2.13 Hasil tipe data array pada browser.....	22
Gambar 2.14 Script tipe data object html	22
Gambar 2.15 Script javascript pada tipe data object.....	23
Gambar 2.16 Contoh script assignment operator	24
Gambar 2.17 Macam macam assignment operator	24
Gambar 2.18 Script operator javascript operator sama dengan	25
Gambar 2.19 Script javascript operator identik.....	26
Gambar 2.20 Script javascript operator tidak sama dengan..	27
Gambar 2.21 Script javascript menggunakan operasi logika	28
Gambar 2.22 Struktur if else javascript	29
Gambar 2.23 Contoh script menggunakan if else javascript..	30
Gambar 2.24 Script looping pada javascript.....	30

Gambar 2.25 Hasil script looping javascript.....	31
Gambar 2.26 Struktur function pada javascript.....	32
Gambar 2.27 Script function custom.....	33
Gambar 2.28 Script function dengan menggunakan if else pada javascript	34
Gambar 2.29 Script function menghitung angka.....	35
Gambar 2.30 Script variabel scope	36
Gambar 2.31 Kesalahan dalam penulisan variabel	37
Gambar 2.32 Metode strict metode global.....	38
Gambar 2.33 Metode stric metode local	39
Gambar 2.34 Metode strict tanpa mendeklarasikan variabel	40
Gambar 2.35 Penggunaan mode strict menghapus variabel	41
Gambar 2.36 Penggunaan mode strict menghapus function .	42
Gambar 2.37 Penggunaan mode strict menduplikasikan parameter	43
Gambar 2.38 Mode strict penggunaan property read only....	44
Gambar 2.39 Mode strict penggunaan property read only....	45
Gambar 2.40 Penggunaan mode strict menghapus properti .	46
Gambar 2.41 Penggunaan mode stric kata eval.....	47
Gambar 2.42 Penggunaan mode strict menggunakan argument kata.....	48
Gambar 2.43 Penggunaan mode strict pernyataan with	49
Gambar 2.44 Tampilan klik kanan pada touchbar atau mouse	50
Gambar 2.45 Tampilan halaman console pada browser	51
Gambar 2.46 Macam macam property dan method pada javascript	52
Gambar 2.47 Mengakses property atau method melalui window.....	54
Gambar 2.48 Mengakses property atau method tanpa melalui window.....	55
Gambar 2.49 Script javascript menggunakan object browser	56

Gambar 2.50 Script javascript menggunakan object browser siap untuk dijalankan	57
Gambar 2.51 Hasil script ketika dijalankan halaman input pertama	58
Gambar 2.52 Hasil script ketika dijalankan halaman inputan kedua	58
Gambar 2.53 Hasil script ketika dijalankan halaman inputan ketiga	58
Gambar 2.54 Hasil akhir script javascript object browser	59
Gambar 2.55 Struktur DOM	60
Gambar 2.56 Script javascript menggunakan Method <code>getElementsByTagName</code>	62
Gambar 2.57 Hasil script menggunakan method <code>getElementsByTagName</code>	62
Gambar 2.58 Script javascript menggunakan Method <code>getElementById</code>	63
Gambar 2.59 Hasil script javascript menggunakan Method <code>getElementById</code>	64
Gambar 2.60 Script javascript menggunakan Method <code>getElementsByClassName</code>	64
Gambar 2.61 Hasil script javascript dengan menggunakan Method <code>getElementsByClassName</code>	65
Gambar 2.62 Script javascript penggunaan <code>OnClick</code>	66
Gambar 2.63 Script javascript penggunaan <code>OnChange</code>	67
Gambar 2.64 Hasil script javascript <code>OnChange</code>	67
Gambar 2.65 Hasil script javascript <code>OnMouseOver</code> dan <code>OnMouseOut</code>	68
Gambar 2.66 Script javascript <code>OnKeyDown</code>	69
Gambar 2.67 Hasil script javascript <code>OnKeyDown</code>	69
Gambar 2.68 Script javascript menggunakan <code>OnLoad</code>	70
Gambar 2.69 Hasil script javascript menggunakan <code>OnLoad</code> ..	70
Gambar 2.70 Dokumen html pembuatan game	71

Gambar 2.71 Membuat div pada html	72
Gambar 2.72 Membuat style untuk element h1	73
Gambar 2.73 Hasil dari style h1	73
Gambar 2.74 Style pada div class container.....	74
Gambar 2.75 Hasil dari style container.....	75
Gambar 2.76 Style div class tanah.....	76
Gambar 2.77 Hasil styling div tanah.....	77
Gambar 2.78 Style div tikus	77
Gambar 2.79 Hasil styling div tikus	78
Gambar 2.80 styling tanah dengan penambahan value overflow	79
Gambar 2.81 Hasil jika ditambahkan overvlow dengan value hidden	79
Gambar 2.82 Script styling tikus untuk muncul diatas tanah	80
Gambar 2.83 Hasil styling div tanah dan div tikus	80
Gambar 2.84 Menuliskan class muncul pada tanah.....	81
Gambar 2.85 Hasil setelah diberi nilai muncul pada tanah	82
Gambar 2.86 Penambahan value transition pada class tikus	83
Gambar 2.87 Penulisan div pada html.....	84
Gambar 2.88 Hasil dari penulisan div ulang	85
Gambar 2.89 Menambahkan value float pada style tanah	86
Gambar 2.90 Hasil penambahan value float dengan class tanah.....	86
Gambar 2.91 Script javascript untuk tanah dan tikus.....	87
Gambar 2.92 Script mengatur munculnya tikus.....	88
Gambar 2.93 Script untuk mengatur waktu minimal dan maksimal.....	89
Gambar 2.94 Membuat tombol start pada html untuk memulai game.....	89
Gambar 2.95 Styling element button	90
Gambar 2.96 Membuat sebuah fungsi pada javascript.....	91
Gambar 2.97 Memberikan variabel untuk memulai.....	92

Gambar 2.98 Membuat papan skor untuk game	92
Gambar 2.99 Memberi style pada element h1 dan h2	93
Gambar 2.100 Membuat query pada class papan-skor	94
Gambar 2.101 membuat script function pukul.....	94
Gambar 2.102 Memastikan semua variabel terdekalarasi	95
Gambar 2.103 Script untuk membuat cursor ketika masuk ke game menjadi palu	96
Gambar 2.104 Script untuk memberikan efek palu ketika pukul tikus	97
Gambar 2.105 Hasil akhir game pemrograman javascript.....	97
Gambar 3.1 Halaman download xampp	102
Gambar 3.2 Halaman control panel xampp.....	103
Gambar 3.3 Halaman control panel xampp start apache dan mysql.....	104
Gambar 3.4 halaman xampp pada browser.....	105
Gambar 3.5 Halaman versi PHP dalam xampp	105
Gambar 3.6 Proses meletakkan dalam file melalui Local Disk(C:).....	106
Gambar 3.7 Proses meletakkan file php dalam folder xampp	107
Gambar 3.8 Cara menampilkan pada browser cara pertama	108
Gambar 3.9 Cara menampilkan pada browser cara kedua..	108
Gambar 3.10 Hasil dari script echo	109
Gambar 3.11 Jika salah atau typo memasukan nama file pada pencarian browser	109
Gambar 3.12 Script penggunaan konstanta	110
Gambar 3.13 Hasil script konstanta dalam PHP	110
Gambar 3.14 Script penggunaan variabel.....	111
Gambar 3.15 Hasil script penggunaan variabel.....	111
Gambar 3.16 Script menggunakan tipe data Integer	113
Gambar 3.17 Hasil script menggunakan tipe data integer...	113

Gambar 3.18 Script PHP menggunakan tipe data float	114
Gambar 3.19 Hasil script menggunakan tipe data float	115
Gambar 3.20 Script menggunakan tipe data boolean	116
Gambar 3.21 Hasil script menggunakan tipe data boolean .	116
Gambar 3.22 Script menggunakan tipe data string.....	117
Gambar 3.23 Hasil script menggunakan tipe data string	117
Gambar 3.24 Script menggunakan tipe data object	118
Gambar 3.25 Hasil script menggunakan tipe data object....	118
Gambar 3.26 Script menggunakan tipe data array.....	119
Gambar 3.27 Script php tipe data array pemanggilan data.	119
Gambar 3.28 Hasil dari script menggunakan tipe data array	120
Gambar 3.29 Script menggunakan tipe data null.....	120
Gambar 3.30 Hasil script tipe data null.....	121
Gambar 3.31 Script php kondisi if.....	122
Gambar 3.32 Hasil script php kondisi if.....	122
Gambar 3.33 Script php kondisi if else if else.....	123
Gambar 3.34 Hasil script php kondisi if elseif else.....	123
Gambar 3.35 Script php kondisi if else.....	124
Gambar 3.36 Hasil script kondisi if else.....	124
Gambar 3.37 Script menggunakan kondisi switch-case	125
Gambar 3.38 Hasil script menggunakan kondisi switch case	126
Gambar 3.39 Script fungsi PHP.....	127
Gambar 3.40 Hasil script fungsi PHP	127
Gambar 3.41 Script php menggunakan argumen	128
Gambar 3.42 Hasil script php menggunakan argumen	129
Gambar 3.43 Script PHP menggunakan 2 argumen.....	129
Gambar 3.44 Hasil script PHP menggunakan 2 argumen....	130
Gambar 3.45 Script PHP menggunakan data default	130
Gambar 3.46 Hasil script PHP menggunakan data default ..	131
Gambar 3.47 Script PHP menggunakan return.....	132

Gambar 3.48 Hasil script PHP menggunakan return	132
Gambar 3.49 Script PHP menggunakan while	134
Gambar 3.50 Hasil script php menggunakan while	135
Gambar 3.51 Script PHP menggunakan do while	135
Gambar 3.52 hasil script php menggunakan do while	136
Gambar 3.53 Script PHP menggunakan for	137
Gambar 3.54 Hasil script php menggunakan do while.....	138
Gambar 3.55 Script php menggunakan foreach	138
Gambar 3.56 Hasil script PHP menggunakan foreach	139
Gambar 3.57 Struktur html untuk pembuatan game PHP ..	140
Gambar 3.58 Script membuat logika komputer	140
Gambar 3.59 Membuat input pada user	141
Gambar 3.60 Membuat logika game.....	142
Gambar 3.61 Membuat hasil menang atau kalah terhadap user dan komputer	143
Gambar 3.62 Styling game pada PHP	144
Gambar 3.63 Halaman game suit PHP	145
Gambar 3.64 Contoh hasil antara user dan komputer	145



PENDAHULUAN

Perbedaan Pembuatan Game dengan Desain Game Sebelum kita belajar tentang cara membuat game lewat script sebaiknya kita tahu dulu arti dari pembuatan game dan desain game. Jika anda ingin membuat game sendiri alangkah baiknya anda memahami apa arti pembuatan game dan desain game itu sendiri. Sebelum anda membuat game anda perlu membuat desain game, desain game sendiri bisa kita buat dengan video. Gimana cara kita membuat desain game yang bagus dan menarik dimata masyarakat bukan hanya lokal saja tetapi bisa sampai ke manca Negara. Dalam membuat desain game tersebut kita harus membuat detail contoh kita membuat game petualang bagaimana cara perang menghadapi lawan bagaimana karakter kita semakin kuat untuk menghadapi lawan dan membuat karakter kita menjadi semakin menarik seiring dengan naiknya level karakter tersebut dan lain-lain.

Setelah kita sudah membuat desain game itu sendiri secara menarik. Langkah selanjutnya adalah membuat game itu sendiri melalui konsep bisa melewati coding atau bisa anda pakai aplikasi seperti rpg maker langsung drag and drop.

Langkah-Langkah dalam Pembuatan Game

Membuat game sendiri terlihat sulit sekali bagi pemula atau bagi mahasiswa menjelang skripsi yang menggunakan aplikasi game untuk skripsi pasti pusing dan akan menyerahkan kepada orang ahli dalam bidang membangun game untuk mengerjakan aplikasi game anda, dan tentu saja meminta tolong kepada orang yang ahli membangun game biayanya tentu tidaklah murah.

Kali ini kita akan bahas bagaimana sih cara membangun game yang baik dan benar. Tapi, perlu diperhatikan proses pengerjaan game berbeda beda. Tergantung dari genre game apa yang anda buat semakin detail game yang anda buat semakin lama waktunya dan pastinya software yang anda pakai juga berpengaruh berikut merupakan langkah langkah dalam membuat game :

Langkah 1 :

Melakukan Riset dan Membangun Konseptualisasi Pada Game

Langkah pertama yang harus kita lakukan dalam membuat game adalah melakukan riset dan membangun konsep supaya game anda menjadi kenyataan dan sesuai target yang anda inginkan.

Langkah 2 :

Membuat sebuah dokumen tentang desain terkait dengan game yang sedang anda buat

Langkah kedua ini sangatlah penting dalam membuat game karena dokumen desain ini akan menentukan

- a. Anda harus menentukan bagaimana tampilan pada game terlihat menarik. Tetapi, biasanya orang orang akan menyewa orang yang ahli dalam game untuk mendesain game anda. Desain game pun tentu tidaklah murah.
- b. Anda harus menentukan bahasa pemrograman mana yang cepat dan efisien untuk membangun game anda.
- c. Anda harus memiliki rencana untuk 5 tahun kedepan atau 10 tahun kedepan untuk game anda. Apakah anda ingin menambah fitur pada game anda supaya terlihat menarik.

Langkah 3 :

Menentukan Software untuk Pembuatan Game Anda

Langkah ini adalah merupakan langkah terakhir proses desain dokumen pada game yang kita buat. Kita harus menentukan software mana yang cocok untuk pembuatan game anda. Pada pemula untuk proses pembuatan game cukup software yang dimana software tersebut memfasilitasi text editor untuk memprogram game tersebut dan compiler untuk menjalankan hasil dari kodingan kita. Sedangkan, jika sudah expert atau ahli biasanya dalam membuat game butuh software yang lebih spesifik lagi.

Langkah 4 :

Mencoba Game anda dan mempromosikan game yang anda buat

Setelah pembuatan game anda selesai, pastikan anda memastikan di game anda terdapat bug atau eror apa tidak. Proses untuk mengecek ada bugnya atau tidak dan memperbaiki jika ada bug pasti membutuhkan waktu yang tidak sedikit.

Setelah kita mengecek game tersebut tidak memiliki bug atau eror langkah selanjutnya adalah mempromosikan game anda supaya orang tertarik dan ingin mencoba buatan game yang anda buat. Berikut merupakan cara untuk mempromosikan game :

- Membuat video tentang game anda yang menarik dan promosi berlangganan di media social yang besar seperti instagram, youtube atau facebook.
- Membuat sebuah website yang menarik tentang game anda supaya pengunjung website tertarik dan ingin membeli game anda lalu mengunduhnya.

- Jika game anda untuk game handphone, lebih baik daftarkan ke app store atau ke play store.

Bahasa Pemrograman Untuk Membuat Game

Untuk membuat game berbasis android, apple atau pc dibutuhkan bahasa pemrograman. Ada banyak bahasa pemrograman yang bisa untuk membuat game entah itu android, apple atau pc. Jika, anda sudah menguasai satu bahasa pemrograman, untuk membuat game akan terasa mudah. Berikut merupakan bahasa pemrograman untuk pembuatan game :



Gambar 1.1 Gambar logo C++

1. C++ merupakan bahasa pemrograman pada tingkat tinggi, bahasa pemrograman ini didirikan oleh perusahaan bernama Bell lab pada tahun 1980-an. Pada awalnya bahasa ini adalah bahasa C, Namun seseorang bernama Bjarne Stroustrup beserta tim mengembangkan bahasa C menjadi C++. Pada bahasa pemrograman C++ memiliki memiliki sistem support

yang berguna pada pemrograman tingkat rendah (low level coding).

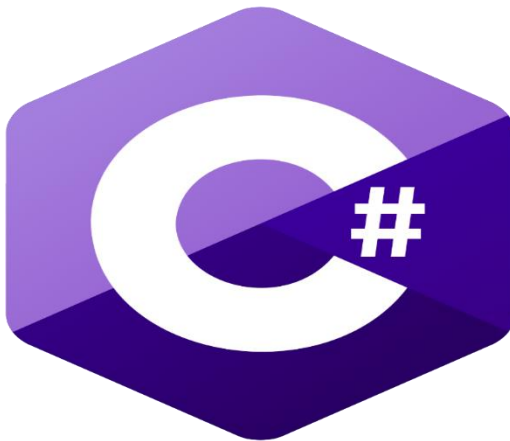


Gambar 1.2 Gambar logo Java

2. Pada tahun 1991, seseorang bernama James Gosling dan Patrick Naughton bekerja pada perusahaan Sun Microsystem, pada waktu mereka sedang mengerjakan sebuah proyek yaitu membuat bahasa pemrograman dengan memiliki tujuan yaitu untuk mempermudah kegiatan manusia. Proyek ini diberi nama kode “Green Project”. Mereka mengerjakan proyek dengan dibantu oleh para programmer, sebagian programmer tersebut terdapat programmer yang membangun bahasa C++. Konsep yang digunakan pada proyek ini adalah bahasa yang sudah OOP (*Object Oriented Programming*). 18 bulan kemudian terciptalah sebuah bahasa pemrograman OOP yang memiliki sifat portable. Kemudian James Gosling memberikan nama pada bahasa pemrograman ini yaitu “Oak”. James Gosling memberikan nama tersebut karena mendapatkan inspirasi dari nama pohon dari seberang kantornya.

Namun nama “Oak” sudah dipakai orang lain sehingga mereka terpaksa harus mengubah nama tersebut.

Pada suatu waktu, James Gosling beserta para insinyur mengunjungi sebuah kedai kopi dan bersantai disana. James Gosling akhirnya mendapatkan ide untuk membuat nama bahasa pemrograman tersebut yaitu “java”. Arti dari java sendiri adalah kopi yang sering diminum oleh “Bapak Java”. Kemudian, James Gosling menduga Kopi berasal dari Indonesia pulau Jawa. Akhirnya memutuskan nama tersebut menjadi bahasa pemrograman Java.



Gambar 1.3 Gambar logo C#

3. Perusahaan Microsoft pada tahun 1990-an membuat sebuah program yaitu Microsoft Visual J++ yang bertujuan untuk supaya sistem operasi Windows dapat menggunakan bahasa pemrograman Java dan *Microsoft Component Object Model (COM)*. Namun

dalam pengembangan Microsoft Visual J++ terjadi masalah yaitu pemegang hak cipta bahasa pemrograman java, Sun Microsystem. Akibatnya, pengembangan Microsoft Visual J++ dihentikan. Kemudian Microsoft beralih untuk menggantikan Microsoft Visual J++. Akhirnya, Microsoft seseorang bernama Andres Helsberg yang merupakan seorang mantan karyawan dari perusahaan Borland yang membangun bahasa Turbo Pascal. Bahasa pemrograman C# diperkenalkan oleh Microsoft pada tahun 2000-an. Bahasa pemrograman C# sebagai bahasa pemrograman modern dan sudah berbasis OOP (Object Oriented Programming). Bahasa ini digunakan untuk pengembangan platform Microsoft Net Framework.



Gambar 1.4 Logo javascript

4. Javascript merupakan bahasa pemrograman *client-side* yang dikembangkan oleh seorang karyawan di perusahaan Netscape yang bernama Brandan Eich. Pada tahun 1995, sebelum menjadi bahasa

pemrograman javascript. Bahasa ini pertama kali dibentuk adalah bernama bahasa pemrograman Mocha. Setelah dikembangkan nama bahasa pemrograman tersebut berganti nama menjadi bahasa pemrograman Mona. Setelah itu dikembangkan lagi dan berganti nama menjadi bahasa pemrograman LiveScript lalu dikembangkan lagi dan akhirnya menjadi bahasa pemrograman yang bernama Javascript. Bahasa tersebut pertama kali digunakan oleh kalangan Netscape dan mempunyai fungsi yang sangat terbatas. Namun, bahasa tersebut dikembangkan lagi oleh komunitas developer yang menggunakan bahasa pemrograman tersebut.

Bahasa pemrograman javascript diresmikan pada tahun 1996 dan menjadi bahasa pemrograman ECMAScript. 2 tahun kemudian tepatnya pada tahun 1998 bahasa ECMAScript telah berkembang menjadi ECMAScript 2. Dan pada tahun 1999, ECMAScript 2 telah berkembang menjadi ECMAScript 3. ECMAScript 3 tersebut dikembangkan lagi sampai bahasa pemrograman javascript yang kita kenal selama ini.

Dari tahun ke tahun bahasa javascript mengalami perkembangan secara pesat. Pada tahun 2016 website di seluruh dunia telah menggunakan bahasa pemrograman javascript sebanyak 92%. Dalam waktu kurang lebih 20 tahun bahasa tersebut sudah memiliki banyak fungsi terutama bagi developer website pasti sering dengar tentang bahasa pemrograman javascript.



Gambar 1.5 Logo PHP

5. PHP(Hypertext Preprocessor) merupakan sebuah bahasa yang ditujukan untuk pengembangan website. Pada tahun 1995, seseorang yang berasal dari Negara Denmark-Kanada yaitu bernama Rasmus Lerdorf. Pada masa perkembangan php. Pada awalnya singkatan PHP adalah *Personal Home Page*, dan nama pemrograman ini adalah *Form Interpreted(FI)*. Form Interpreted ini hanyalah sebuah kumpulan script. Kumpulan script berfungsi sebagai mengolah data pada website. Kemudian, Rasmus Lerdorf membuat kode dan menamakan kode tersebut yaitu PHP/FI, yang memiliki sifat terbuka. Sejak saat itu para programmer antusias mengembangkan bahasa pemrograman PHP.

Pada tahun tahun 1997, bahasa pemrograman php memiliki perkembangan dan PHP/FI berubah nama

menjadi PHP/FI 2.0. Bahasa PHP telah di implementasikan terhadap bahasa C. Sejak perlisian PHP/FI 2.0 terdapat modul modul ekstensi yang menunjukkan peningkatan kemampuan dari PHP/FI secara signifikan.

Pada tahun 1997, terjadi penulisan ulang interpreter PHP yang bertujuan supaya PHP menjadi lebih baik dan menjadi lebih cepat dari versi sebelumnya. Penulisan ulang tersebut dilakukan oleh sebuah perusahaan bernama Zend. Pada tahun 1998, Perusahaan tersebut mengeluarkan versi terbaru dari PHP yang dimana hasil dari penulisan Interpreter pada PHP yaitu adalah PHP 3.0, versi tersebut juga merubah dari singkatan PHP itu sendiri yaitu singkatan PHP berubah menjadi *Hypertext Processing*.

Selanjutnya kita belajar bahasa pemrograman secara dasar terlebih dahulu sebelum membuat game. Pada buku ini bahasa pemrograman terdapat javascript dan php untuk pembuatan game.

JavaScript

Pada judul kali ini kita akan membahas javascript, javascript merupakan bagian dasar untuk membuat suatu website. Javascript berbeda dengan html dan css, javascript merupakan bahasa pemrograman yang digunakan untuk website. fungsi dari javascript sendiri bukan sebagai mempercantik atau memperindah website melainkan membuat sebuah website lebih interaktif dan fungsionaris.

Pada bab ini kita akan mempelajari tentang apa itu javasript, variable dalam javascript, if else dalam javascript, array dalam javascript, fungsi operator, dll.

Pengertian Javascript

Javascript merupakan bahasa pemrograman website yang memiliki sifat *Client Side Programming Language*. Sifat ini memiliki proses untuk pihak client. Bahasa pemrograman *Client Side Programming Language* berbeda dengan bahasa pemrograman *Server Side* contohnya seperti bahasa pemrograman PHP.

Untuk menjalankan javascript, kita hanya perlu membutuhkan software text editor dan browser sebagai menampilkan hasil dari koding yang anda buat. Javascript memliki beberapa fitur yaitu:

- High Level Programming Language
- Client-side
- Berorientasi objek

Javascript pertama kali dikembangkan bertujuan untuk membuat interaksi antara client dengan website menjadi lebih cepat tanpa harus menunggu loading server dari website tersebut.

Dalam perkembangan javascript, javascript memiliki permasalahan seperti kode pemrograman yang memiliki sifat client side. Karena itu javascript merupakan bahasa pemrograman tersulit pada saat itu. Semakin bertambahnya tahun javascript mulai bangkit kembali. Javascript ditawarkan oleh komunitas programmer untuk membuat library javascript yaitu jQuery.

Kelebihan Javascript

Bahasa pemrograman Javascript memiliki kelebihan dari bahasa pemrograman lainnya, untuk beberapa kasus tertentu. Berikut merupakan kelebihan yang dimiliki oleh bahasa Javascript, diantaranya:

- Untuk memahami bahasa pemrograman javascript lebih mudah daripada bahasa pemrograman lainnya.
- Bahasa pemrograman javascript tidak perlu memerlukan sebuah compiler seperti bahasa pemrograman lainnya. Untuk melihat hasilnya cukup kita melakukan pada web browser. Web browser tersebut akan otomatis menginterpretasi dengan script html yang anda buat.
- Ketika script javascript mengalami eror maka untuk mencari kode eror tersebut menjadi lebih mudah.
- Bahasa javascript bisa kita pakai di berbagai browser.
- Ketika anda membangun sebuah website dan menggunakan bahasa javascript, website tersebut

tidak hanya memiliki tampilan yang menarik saja tetapi website tersebut akan menjadi lebih interaktif.

Kekurangan Javascript

Bahasa javascript selain memiliki kelebihan juga memiliki kekurangan. Bahasa pemrograman javascript dari tahun ke tahun semakin berkembang. Namun, oleh sebagian orang bahasa tersebut disalahgunakan oleh para hacker dan scammer dalam mencari celah keamanan. Berikut merupakan macam macam kekurangan yang dimiliki oleh bahasa pemrograman javascript, diantaranya:

- Bahasa javascript memiliki resiko yang sangat tinggi terhadap eksploitasi
- Bahasa javascript bisa disalah gunakan oleh para hacker untuk dapat mengaktifkan kode yang sangat berbahaya di komputer.
- Bahasa javascript bisa di render ke masing masing perangkat yang berbeda dan perangkat tersebut akan mengarah ke inkonsistensi.

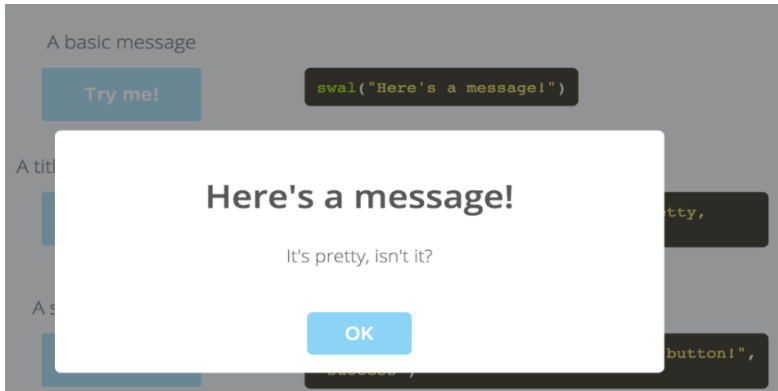
Fungsi dan cara kerja javascript terhadap website

Bahasa javascript merupakan bahasa yang bersifat seperti halnya html dan css yaitu *client-side programming*. Bahasa javascript bisa anda tambahkan ke halaman website itu sendiri atau bisa ke file javascript yang kalian buat sendiri. Bahasa javascript bisa anda nonaktifkan tetapi hanya di sebagian website browser.

Penggunaan umum pada javascript

Fungsi dari penggunaan bahasa pemrograman javascript berubah ubah seiring dari tahun ke tahun. Pada awalnya

bahasa javascript digunakan untuk berinteraksi kepada pengguna (*user*) dengan memberikan sebuah umpan untuk mendapatkan komentar dari user. Kita bisa menggunakan element `alert()` untuk menotifikasi user jika ada sesuatu masalah.



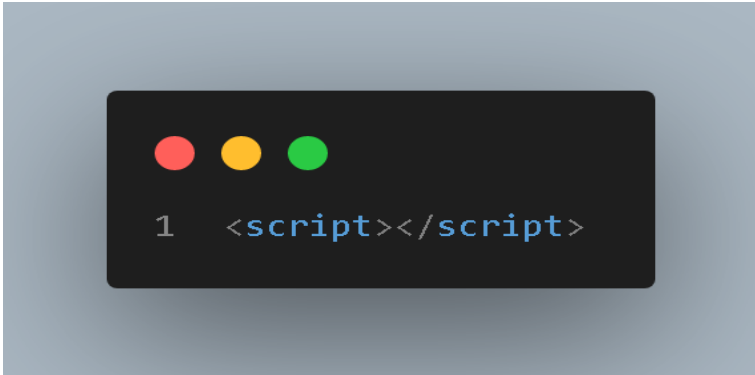
Gambar 2.1 Contoh pesan peringatan terhadap user

Menuliskan syntax JavaScript pada HTML

Pada sub bab kali ini kita akan membahas tentang bagaimana aturan cara menulis javascript. Berikut merupakan cara penulisan javascript.

Embedded Script

Cara yang pertama Embedded yaitu menuliskan syntax script ke dalam html kita bisa menuliskan syntax ini bisa ke dalam element `<head>` atau `<body>` tetapi seringkali programmer menuliskan syntax script ke dalam element `<body>`.



Gambar 2.2 Contoh syntax embedded script

External Script

Cara yang kedua External yaitu menuliskan element script kedalam body kemudian diberi tambahan `src=""`. untuk menyambungkan file html ke file javascript yang kita tuju. Berikut merupakan contoh penerapannya.



Gambar 2.3 Contoh syntax external script

Gambar 2.3 merupakan external script dengan file yang kita sambungkan adalah file bernama main. Untuk nama format

berbeda dengan html css ditulis dengan nama .html dan .css. Sedangkan javascript kita tulis dengan .js.

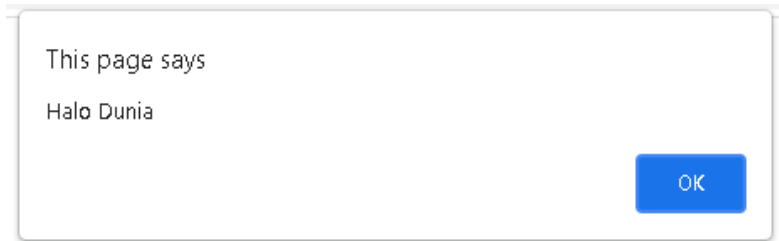
Statement pada Javascript

Bahasa javascript memiliki banyak statement dengan syntax yang sesuai. Statement yang kita tulis akan memberi perintah kepada browser apa yang harus dilakukan oleh si browser. Salah satu contohnya yaitu alert. Berikut merupakan contohnya.



Gambar 2.4 Contoh script alert

Pada script diatas terdapat akhiran tanda titik koma (;) ini biasa di bahasa pemrograman seperti PHP atau C++ bagi yang sudah mengajari bahasa pemrograman PHP atau C++ mungkin terasa lebih mudah untuk mempelajari javascript. Berikut merupakan hasil dari script jika ditampilkan oleh browser.



Gambar 2.5 Hasil script alert

Variabel pada Javascript

Variabel merupakan sebuah wadah untuk menyimpan sebuah value. Value terdiri dari number, string dan boolean. Cara pendeklarasian variable javascript adalah “var”. Berikut merupakan contoh penerapannya.

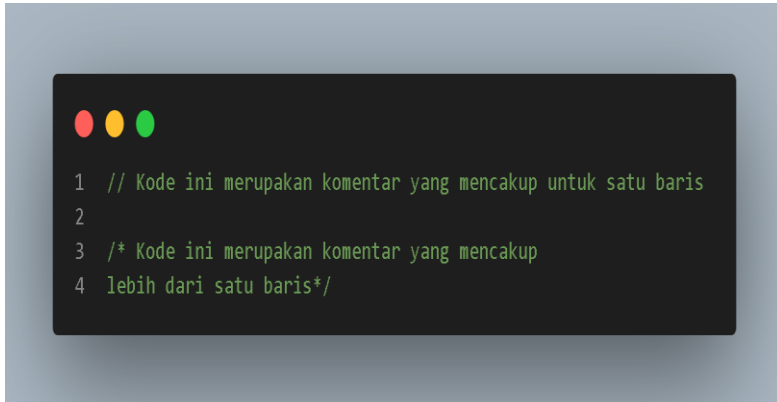


Gambar 2.6 Contoh script variabel menggunakan var

Dalam bahasa pemrograman javascript untuk mendeklarasikan variabel tidak cuman var aja tetapi masih ada variabel “let” dan juga variabel “const”.

Comment pada Javascript

Pada javascript memiliki fitur yaitu komentar yang berfungsi sebagai pengingat kode. Komentar html dan javascript memiliki cara penulisan berbeda. Dalam javascript memiliki cara 2 penulisan. Berikut merupakan contoh penulisannya.



```
1 // Kode ini merupakan komentar yang mencakup untuk satu baris
2
3 /* Kode ini merupakan komentar yang mencakup
4 lebih dari satu baris*/
```

Gambar 2.7 Contoh script comment

Tipe Data pada Javascript

Pada sub bab ini kita akan membahas tentang tipe data pada javascript. Terdapat 5 tipe data dalam javascript. Berikut merupakan contohnya.

- String
- Integer
- Boolean
- Array
- Object

String

Tipe data string memiliki fungsi untuk sebagai mendeklarasikan kata atau kalimat. Berikut merupakan

contoh tipe data string.

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title>Tipe data string</title>
5 </head>
6 <body>
7   <h1>Tipe Data string</h1>
8   <div id="pertama"></div>
9
10  <script>
11    //Tipe Data String
12    var contoh = "Halo nama saya joko sumarno";
13
14    //menampilkan tipe data string pada element pertama
15    document.getElementById("pertama").innerHTML=contoh;
16  </script>
17 </body>
18 </html>
```

Gambar 2.8 Contoh script penggunaan tipe data string

Gambar 2.8 merupakan contoh penerapan dari tipe data string berikut merupakan hasil script jika kita tampilkan ke browser maka akan jadi seperti ini.



Tipe Data string

Halo nama saya joko sumarno

Gambar 2.9 Hasil script tipe data string

Integer

Tipe data number atau integer (int) merupakan tipe data yang menunjukkan angka bilangan bulat. Berikut merupakan

contohnya.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title>Tipe data Number</title>
5 </head>
6 <body>
7   <h1>Tipe Data Number</h1>
8   <div id="angka"></div>
9
10  <script>
11    //Tipe Data Number
12    var nomer = "46";
13
14    //menampilkan tipe data number pada element pertama
15    document.getElementById("angka").innerHTML=nomer;
16  </script>
17 </body>
18 </html>
```

Gambar 2.10 Contoh script tipe data number

Dari gambar diatas jika kita tampilkan di browser maka hasilnya akan seperti ini



Gambar 2.11 Hasil script tipe data number

Array

Array merupakan tipe data yang berfungsi sebagai menyimpan banyak data bisa berbentuk string maupun number dalam satu variabel. Cara menghitung array dimulai dari angka 0 dan seterusnya. Berikut merupakan contoh dari tipe data array dalam javascript.

A screenshot of a code editor with a dark background and light text. The code is a mix of HTML and JavaScript. It starts with a DOCTYPE declaration, followed by an HTML document structure with a title 'Tipe data Number' and a body containing an h1 'Tipe Data Number' and a div with id 'macam'. A script block follows, containing a JavaScript array declaration: 'var tumbuhan = ["Mawar", "Melati", "Anggrek", "Katkus"];'. Then, it uses 'document.getElementById("macam").innerHTML=tumbuhan[1];' to set the content of the 'macam' div to the second element of the array. The code is numbered from 1 to 18.

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title>Tipe data Number</title>
5 </head>
6 <body>
7   <h1>Tipe Data Number</h1>
8   <div id="macam"></div>
9
10  <script>
11    //Tipe Data Array
12    var tumbuhan = ["Mawar", "Melati", "Anggrek", "Katkus"];
13
14    //menampilkan tipe data array pada element pertama
15    document.getElementById("macam").innerHTML=tumbuhan[1];
16  </script>
17 </body>
18 </html>
```

Gambar 2.12 contoh script tipe data array

Pada gambar diatas merupakan tipe data array disini kita mempunyai 4 nilai string dan kita memanggil nama variable

dan memberi [1], kode [1] ini yang dimaksud adalah memanggil nomer 1 pada variable tumbuhan dan array menghitung data dimulai dari angka 0. Berikut merupakan hasil jika ditampilkan oleh browser.

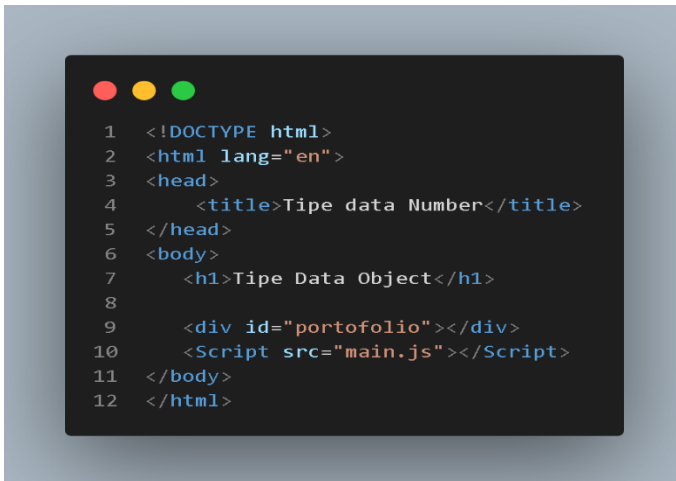
Tipe Data Array

Melati

Gambar 2.13 Hasil tipe data array pada browser

Object

Object merupakan tipe data yang fungsinya hampir sama seperti array yaitu menyimpan variabel. Berikut merupakan contohnya.

A screenshot of a code editor with a dark background and light text. The code is HTML and is numbered from 1 to 12. It defines a document type, language, title, and body content. The body content includes an h1 tag with the text 'Tipe Data Object', a div tag with id 'portofolio', and a script tag with src 'main.js'.

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title>Tipe data Number</title>
5 </head>
6 <body>
7   <h1>Tipe Data Object</h1>
8
9   <div id="portofolio"></div>
10  <Script src="main.js"></Script>
11 </body>
12 </html>
```

Gambar 2.14 Script tipe data object html

Gambar diatas kita membuat document html dan kita menggunakan external script berikut merupakan script javascript yang akan dibuat.

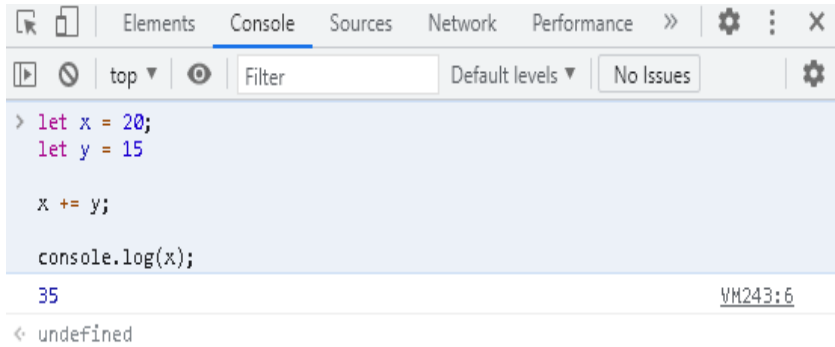
A screenshot of a code editor with a dark background and light-colored text. The code is written in JavaScript and demonstrates how to create an object and use it to update the content of an HTML element. The code is as follows:

```
1 //Tipe data object
2 var datadiri = {
3     nama : "Joko Sumarno",
4     umur : "20",
5     domisili : "Surabaya",
6     pekerjaan : "Programmer"
7 };
8
9 //menampilkan tipe data object pada id portofolio
10 document.getElementById("portofolio").innerHTML="Nama saya " + datadiri.nama +
11 " umur saya " + datadiri.umur + " saya tinggal di " + datadiri.domisili +
12 " pekerjaan saya sebagai " + datadiri.pekerjaan;
```

Gambar 2.15 Script javascript pada tipe data object

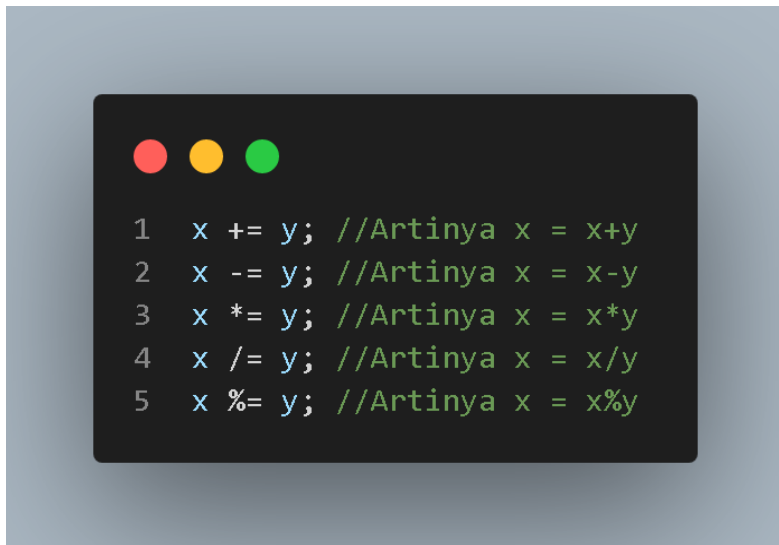
Assignment Operator pada javascript

Pada operator ini berfungsi sebagai menginisialisasi sebuah variable. Pada operator ini menggunakan tanda sama dengan (=) . Tanda ini berfungsi sebagai menginisialisasi variable. Berikut merupakan contoh dari assignment operator, kita memakai variable x dan y.



Gambar 2.16 Contoh script assignment operator

Pada gambar 2.16 kita memakai operator $x += y$ yang artinya $x = x+y$. Tidak cuman $x += y$, ada beberapa macam operasi matematika yang ada di javascript.



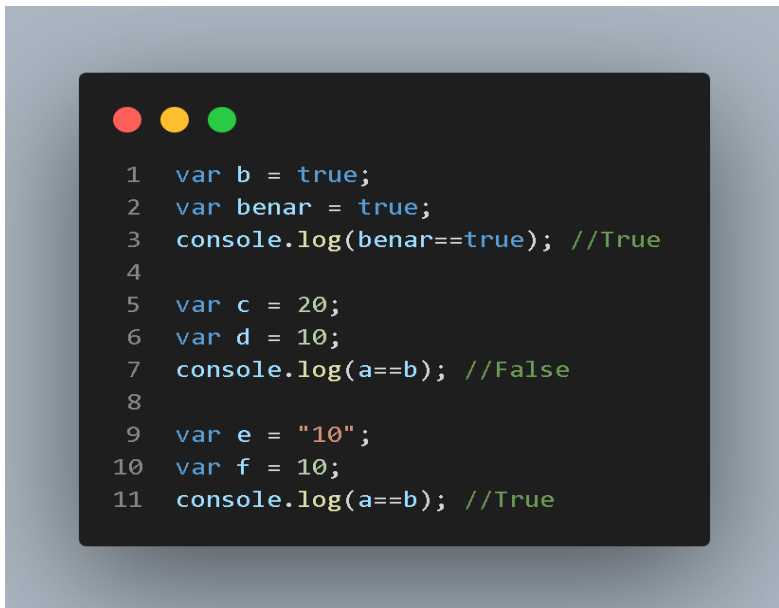
Gambar 2.17 Macam macam assignment operator

Operator Perbandingan pada JavaScript

Dalam bahasa pemrograman javascript terdapat operator yang berfungsi sebagai membandingkan sebuah variable dengan variable lainnya. Hasil dari perbandingan tersebut akan menghasilkan tipe data Boolean. Berikut contoh dari operator perbandingan dalam javascript.

Operator Sama Dengan (==)

Operator ini merupakan operator yang berfungsi sebagai membandingkan variable dengan variable lainnya dan hasil akhirnya akan menjadi tipe data Boolean. Berikut merupakan contoh dari operator sama dengan.



```
1  var b = true;
2  var benar = true;
3  console.log(benar==true); //True
4
5  var c = 20;
6  var d = 10;
7  console.log(a==b); //False
8
9  var e = "10";
10 var f = 10;
11 console.log(a==b); //True
```

Gambar 2.18 Script operator javascript operator sama dengan

Operator Identik (===)

Operator identik memiliki fungsi yang hampir sama dengan operator sama dengan yaitu membandingkan dua buah variabel dan hasil akhirnya menunjukkan tipe data Boolean. Namun Perbedaannya adalah Operator identik jauh lebih spesifik daripada operator sama dengan contohnya kita masukkan data variabel a dan variabel b. Variabel a memiliki value 10 dan variabel memiliki nilai string "10" pada operator sama dengan jika dibandingkan akan bernilai true, pada operator indentik akan bernilai false.

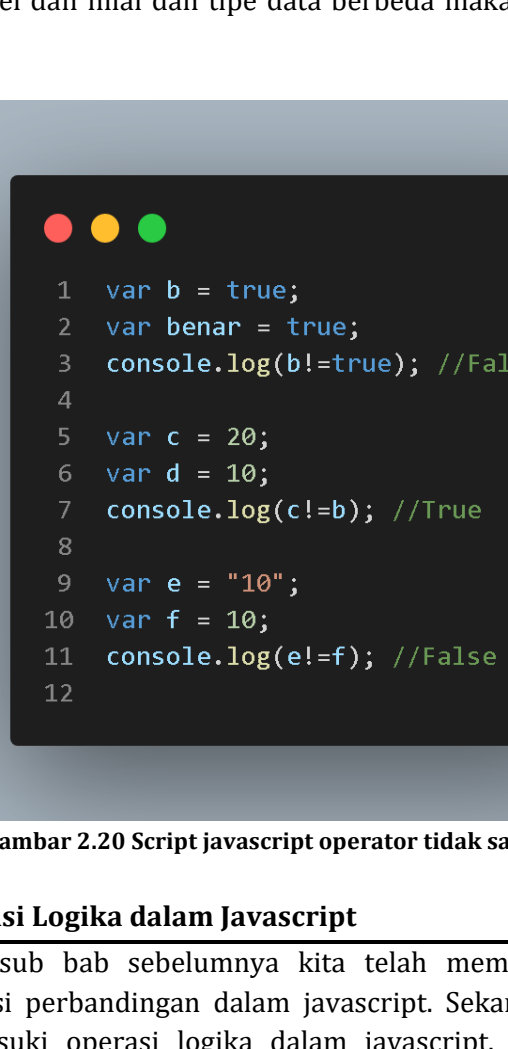
A screenshot of a code editor with a dark background and light-colored text. The code is written in JavaScript and demonstrates the use of the strict equality operator (===). It consists of 15 lines of code, each numbered from 1 to 15. The code defines several variables and then uses console.log to output the results of strict equality comparisons. The comparisons are: benar===true (True), a===b (False), a===b (False), and a===b (True). The code is as follows:

```
1 var b = true;
2 var benar = true;
3 console.log(benar===true); //True
4
5 var c = 20;
6 var d = 10;
7 console.log(a===b); //False
8
9 var e = "10";
10 var f = 10;
11 console.log(a===b); //False
12
13 var g = "20";
14 var h = "20";
15 console.log(a===b); //True
```

Gambar 2.19 Script javascript operator identik

Operator Tidak Sama Dengan (!=)

Operator tidak sama dengan merupakan kebalikan dari operator sama dengan. Jika kita memiliki lebih dari satu variabel dan nilai dan tipe data berbeda maka hasilnya akan true.

A screenshot of a JavaScript console window with a dark background and light text. At the top left, there are three colored circles: red, yellow, and green. The console contains 12 lines of code, each preceded by a line number. The code demonstrates the != operator with three examples: 1. var b = true; var benar = true; console.log(b!=true); //False. 2. var c = 20; var d = 10; console.log(c!=b); //True. 3. var e = "10"; var f = 10; console.log(e!=f); //False. The output of each console.log statement is shown in green text to the right of the code line.

```
1  var b = true;
2  var benar = true;
3  console.log(b!=true); //False
4
5  var c = 20;
6  var d = 10;
7  console.log(c!=b); //True
8
9  var e = "10";
10 var f = 10;
11 console.log(e!=f); //False
12
```

Gambar 2.20 Script javascript operator tidak sama dengan

Operasi Logika dalam Javascript

Pada sub bab sebelumnya kita telah membahas tentang operasi perbandingan dalam javascript. Sekarang kita telah memasuki operasi logika dalam javascript. Dalam bahasa javascript terdapat tiga buah operator logika. Berikut adalah macam macam dari operator logika.

Tabel 2.1 Macam macam operasi logika javascript

Operator	Deskripsi
&&	Operator ini memiliki fungsi menghasilkan nilai true apabila semua kondisi terpenuhi.
	Operator ini memiliki fungsi menghasilkan nilai true apabila salah satu kondisi bernilai true.
!	Operator ini memiliki fungsi bernilai kebalikan nilai suatu kondisi.

Berikut merupakan contoh penerapan operator logika menurut table diatas.

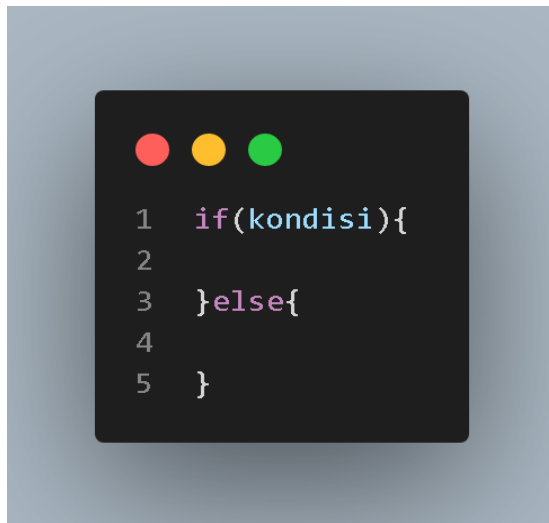
A screenshot of a JavaScript console window with a dark background and light green text. The code demonstrates the use of logical operators: && (AND), || (OR), and ! (NOT). It includes variable declarations for 'a' and 'b', and console.log statements with comments explaining the expected results of the logical operations.

```
1 let a = 10;
2 let b = 20;
3
4 //Operator logika &&
5 console.log(a < 15 && b > 15); // (true && true) -> true
6 console.log(a > 20 && b > 10); // (false && true) -> false
7
8 //Operator logika ||
9 console.log(a < 15 || b>15) // (true || true) ->true
10 console.log(a > 20 || b > 10); // (false || true) -> true
11
12 //Operator logika !
13 console.log(!(a < 15)) // !(true) -> false
14 console.log(!(b > 25))// !(false) ->true
```

Gambar 2.21 Script javascript menggunakan operasi logika

If else

Bagi kalian anak SMK jurusan RPL atau kuliah di jurusan IT yang sedang belajar atau sudah mempelajari bahasa pemrograman lain mungkin tidak asing dengan operator yang satu ini. Secara fungsi memiliki kesamaan yaitu jika suatu kondisi terpenuhi maka akan lanjut ke program selanjutnya jika kondisi tidak terpenuhi maka program tersebut akan berhenti. Berikut merupakan contoh dari operator if else.



Gambar 2.22 Struktur if else javascript

Gambar 2.22 merupakan struktur dari operator if else. Selanjutnya mari kita terapkan operator if else pada console

browser.

```
> //buat variabel Number angka dan isi dengan nilai 17
var angka=24;

// jika sisa hasil bagi (mod) variabel angka dengan 2 adalah 0,
// maka jalankan perintah console.log()
if (angka%3==0)
    console.log("Angka adalah bilangan genap");
else
    console.log("Angka adalah bilangan ganjil");
```

Angka adalah bilangan genap VM109:7

Gambar 2.23 Contoh script menggunakan if else javascript

Looping pada Javascript

Loop merupakan sebuah perulangan yang bertujuan untuk memudahkan kita dalam membuat sebuah daftar, atau perhitungan secara dinamis tanpa menulis kode secara manual. Sekarang mari kita terapkan dan kita membuat looping menggunakan for berikut contohnya.

```
> for(i = 0; i < 10; i++){
    document.write("<p>Perulangan ke- "+i+"</p>")
}
< undefined
```

Gambar 2.24 Script looping pada javascript

Pada program diatas kita telah membuat looping menggunakan for. Disini kita menggunakan variabel *i* untuk contohnya. Yang pertama adalah *i = 0* artinya adalah variabel *i* akan dimulai dari angka 0. Yang kedua adalah *i < 10* artinya adalah disaat kita melakukan looping batas akhirnya adalah kurang dari 10 dimana batas akhir tersebut menuju ke angka 9. Yang ketiga adalah *i++* yang memiliki arti *i = i+1* berikut merupakan hasil dari program tersebut.

Perulangan ke- 0
Perulangan ke- 1
Perulangan ke- 2
Perulangan ke- 3
Perulangan ke- 4
Perulangan ke- 5
Perulangan ke- 6
Perulangan ke- 7
Perulangan ke- 8
Perulangan ke- 9

Gambar 2.25 Hasil script looping javascript

Function

Pada bahasa javascript sering kita melihat kode `console.log()`. Pada `console.log()` memiliki sebuah function yaitu `.log()`. Fungsi ini berfungsi sebagai menampilkan data pada console browser. Kalian terpikir tidak bagaimana cara kerja sebuah fungsi pada javascript ?, Berikut merupakan penjelasan cara kerja fungsi.

Fungsi adalah sebuah potongan kode jika anda tidak menulisnya atau memanggilnya maka kode tersebut tidak dapat mengerjakan sesuai dengan tugasnya. Contohnya fungsi dalam javascript yaitu `alert()`. Kode `alert()` dalam javascript berfungsi sebagai menunjukkan pesan kepada user. Fungsi `alert()` pada javascript sudah tersedia dalam browser. Namun, jika anda tidak memanggil fungsi tersebut maka tidak dapat menampilkan pesan.

Semua fungsi pada bahasa pemrograman memiliki struktur fungsi yang sama yaitu anda dapat menulis nama fungsi terlebih dahulu lalu dilanjutkan dengan tanda () sebagai argument. Terdapat dua jenis fungsi yang pertama adalah memakai argument yang kedua adalah tidak memakai argument. Setelah itu kita menuliskan tanda {} isi dari kurung kurawal adalah pernyataan atau perintah. Berikut merupakan struktur dari function.

```
nama_fungsi(argumen)  
{  
    ... pernyataan / perintah;  
    ... pernyataan / perintah;  
    ... pernyataan / perintah;  
}
```

Gambar 2.26 Struktur function pada javascript

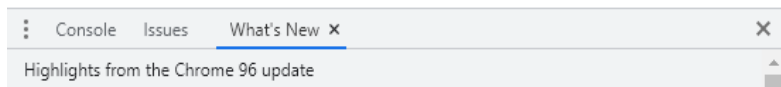
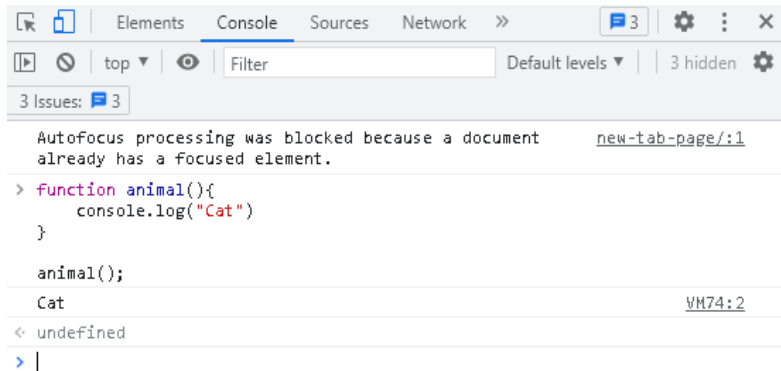
Function dalam javascript memiliki 2 jenis, yaitu custom function dan native function.

Native function merupakan sebuah fungsi yang sudah di fasilitasi oleh javascript dan sebagai developer kita hanya memakai function tersebut. Contoh dari fungsi javascript yaitu

- alert()
- confirm()
- date()
- parseInt()

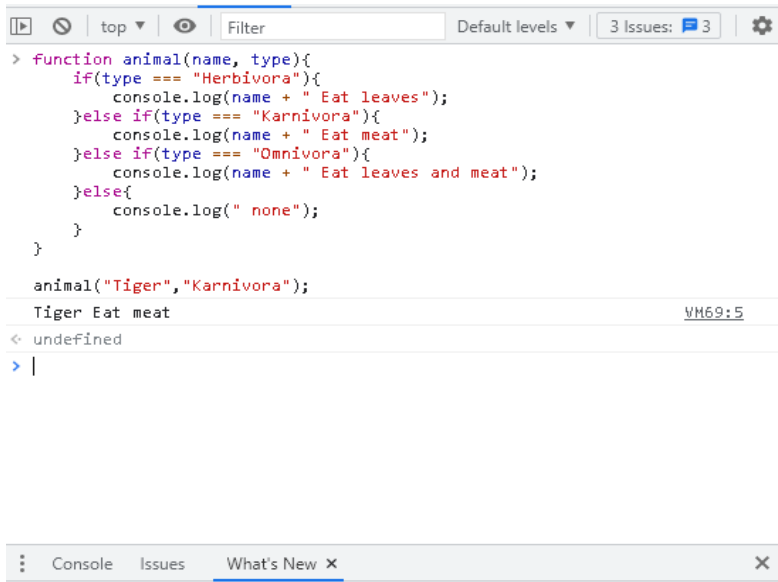
Custom function merupakan sebuah fungsi yang bisa anda buat sendiri sesuai program yang anda buat. Cara untuk

membuat custom function adalah anda memasukan keyword function dan jangan lupa menulis struktur dari function tersebut. Berikut salah satu contoh custom function:



Gambar 2.27 Script function custom

Untuk menambahkan argument pada function javascript, anda perlu menambahkan variabel di dalam tanda kurung tersebut. Namun, anda tidak perlu untuk menambahkan beberapa variabel yaitu var, let, maupun const berikut salah satu contohnya



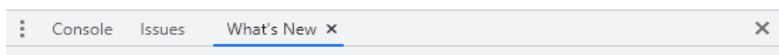
```
> function animal(name, type){
  if(type === "Herbivora"){
    console.log(name + " Eat leaves");
  }else if(type === "Karnivora"){
    console.log(name + " Eat meat");
  }else if(type === "Omnivora"){
    console.log(name + " Eat leaves and meat");
  }else{
    console.log(" none");
  }
}

animal("Tiger", "Karnivora");
Tiger Eat meat
< undefined
```

Gambar 2.28 Script function dengan menggunakan if else pada javascript

Berikutnya adalah function pada javascript bisa dapat melakukan perhitungan matematika dan hasilnya juga anda masukan ke dalam variabel. Berikut merupakan contohnya.

```
> function multiply(x,y,z){
  return x+y*z;
}
let result = multiply(6,8,4)
console.log(result)
38 VM88:5
< undefined
>
```



Gambar 2.29 Script function menghitung angka

Variabel Scope

Pada sub-bab sebelumnya anda telah mempelajari function pada javascript. Pada sub-bab kali ini kita membahas variabel scope pada javascript . Variabel scope ini sangat penting karena anda ketika membuat sebuah program dimana sebuah variabel dapat diakses oleh seluruh script dan keadaan dimana variabel tersebut hanya untuk cakupan fungsi dan turunan fungsi tersebut.

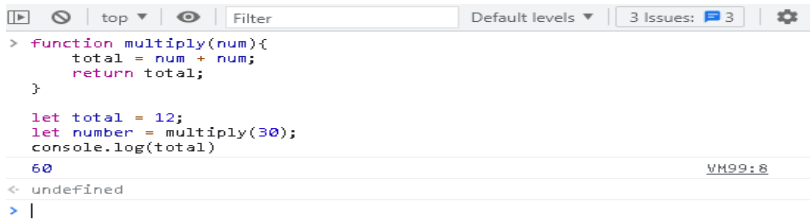
Variabel scope memiliki dua jenis yaitu yang pertama adalah variabel “globally scoped”. Variabel ini memiliki akses pada seluruh script. Yang kedua adalah variabel “locally scoped”. Variabel ini memiliki akses pada function tertentu. Berikut merupakan gambar simulasi dari variabel scope.



```
1 //variabel ini kita buat variabel global
2 const x = "x";
3
4 function animal() {
5     //variabel ini adalah variabel local yang dimana,
6     //variabel ini dapat mengakses animal dan herbivora
7     const y = "y";
8
9     function herbivora(){
10         //variabel ini memiliki akses hanya herbivora
11         const z = "z";
12     }
13 }
```

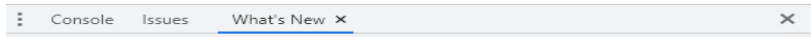
Gambar 2.30 Script variabel scope

Tapi perlu anda ingat, anda harus berhati-hati dalam menggunakan variabel di dalam fungsi. Karena akan mendapatkan hasil yang anda tidak pikirkan. Berikut salah satu contohnya.



```
> function multiply(num){
  total = num + num;
  return total;
}

let total = 12;
let number = multiply(30);
console.log(total)
60 VM99:8
< undefined
> |
```



Gambar 2.31 Kesalahan dalam penulisan variabel

Anda pasti mengira output dari script gambar javascript diatas adalah 12 ternyata output mengeluarkan hasil yang berbeda. Kenapa ?, karena disebabkan kita memasukkan fungsi multiply(). Kita tidak memasukkan variabel total sebagai variabel lokal. Kita juga tidak memasukan var, let maupun const pada variabel total pada fuction multiply() Sehingga, variabel total otomatis menjadi variabel global.

Mode Strict pada Javascript

Mode Stric pada bahasa pemrograman javascript pertama kali diperkenalkan oleh ECMAScript 5, mode stric merupakan sebuah cara ke ikut sertaan dalam varian javascript. Mode stric ini berfungsi sebagai jika kita menggunakan javascript dan kita melihat hasil dari script kita di browser dan script kita tidak bekerja, maka paling cocok menggunakan mode stric. Mode stric terdapat pada semua versi browser. Contohnya browser chrome mode stric ada pada versi 13, internet explorer versi 10 sampai versi sekarang dan browser safari mulai versi 5.1.

Pengertian Mode Strict

Mode Strict merupakan perubahan bahasa javascript menjadi semakin ketat. Semakin ketat ini adalah bahasa javascript engine menjadi bahasa javascript modern. Salah satu contohnya adalah kita diwajibkan menggunakan var, let maupun const sebelum mendeklarasikan sebuah variabel dan masih banyak lainnya.

Cara Mengaktifkan dan Hal yang tidak boleh pada Mode Strict pada Javascript

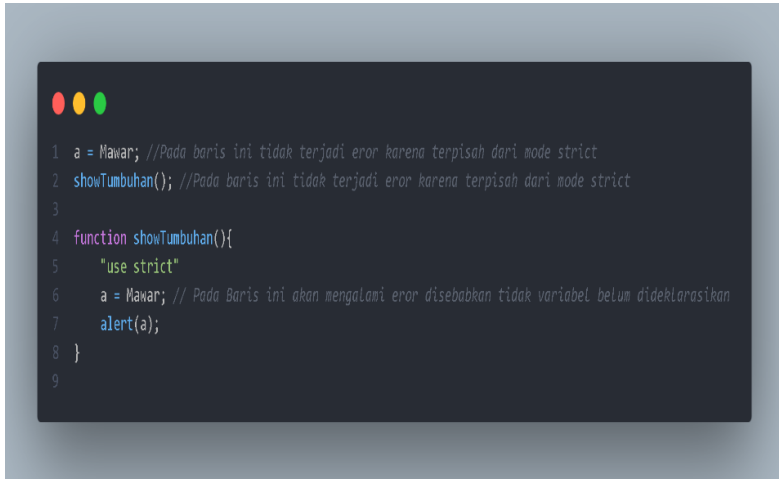
Cara mengaktifkan mode strict pada javascript cukup mudah yaitu anda menulis "use strict" pada script anda. Penulisan "use strict" ditulis dalam bentuk biasa dan tidak dalam bentuk statement. Jika anda menulis "use strict" di awal maka menjadi global variabel jika anda menulis "use strict" di dalam fungsi maka menjadi lokal variabel. Berikut merupakan contoh penulisan mode strict.



```
1 "use strict"
2 a = 2; //ini akan mengalami eror karena variabel belum dideklarasikan
3 showAngka(); //ini akan mengalami eror karena function belum dideklarasikan
4
5 function showAngka(){
6     var a=2;
7     alert(a);
8 }
9
```

Gambar 2.32 Metode strict metode global

Pada gambar 2.32 merupakan contoh mode strict dengan metode global yaitu penulisan diawal.

A screenshot of a code editor with a dark background and light-colored text. The code is written in PHP and includes comments explaining the strict mode behavior. The code is as follows:

```
1 a = Mawar; //Pada baris ini tidak terjadi eror karena terpisah dari mode strict
2 showTumbuhan(); //Pada baris ini tidak terjadi eror karena terpisah dari mode strict
3
4 function showTumbuhan(){
5     "use strict"
6     a = Mawar; // Pada Baris ini akan mengalami eror disebabkan tidak variabel belum dideklarasikan
7     alert(a);
8 }
9
```

Gambar 2.33 Metode stric metode local

Pada gambar 2.33 merupakan contoh script mode strict dengan metode local yaitu penulisannya di dalam function.

Pada kedua gambar diatas merupakan contoh penulisan mode strict secara global dan secara lokal. Berikut merupakan hal yang tidak boleh dalam penulisan mode strict.

- Pada script mode strict ketika anda menggunakan sebuah variabel, anda harus mendeklarasikan variabel tersebut. Jika tidak akan menyebabkan eror.

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10 <h3>"use strict":</h3>
11 <h4>Jika anda menggunakan objek tanpa mendeklarasikan objek tersebut, bisa menyebabkan error </h4>
12
13 <script>
14   "use strict";
15   a = {
16     p3:5,
17     p4:10
18   }; // ini akan menyebabkan error karena variabel a tidak terdefinisi
19 </script>
20 </body>
21 </html>
```

Gambar 2.34 Metode strict tanpa mendeklarasikan variabel

- Pada script mode strict ketika anda menggunakan sebuah variabel, anda tidak boleh menghapus variabel tersebut.



```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <title></title>
5     <meta charset="UTF-8">
6     <meta name="viewport" content="width=device-width, initial-scale=1">
7   </head>
8   <body>
9     <h2>use strict :</h2>
10    <h3>Menghapus fungsi pada mode strict, tidak diperbolehkan</h3>
11
12    <script>
13      "use strict";
14      var b = 30.6;
15      delete b; //Ini akan menyebabkan eror
16    </script>
17  </body>
18 </html>
```

Gambar 2.35 Penggunaan mode strict menghapus variabel

- Pada script javascript menggunakan mode strict, anda tidak boleh menghapus function karena bisa menyebabkan terjadinya eror.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10  <h2>"use strict":</h2>
11  <h3>menghapus sebuah function, tidak diperbolehkan</h3>
12
13  <script>
14    "use strict";
15    function c(p5, p6);
16      delete c;           // Ini akan menyebabkan terjadinya eror
17  </script>
18 </body>
19 </html>
```

Gambar 2.36 Penggunaan mode strict menghapus function

- Pada script javascript dengan menggunakan metode strict, anda tidak diperbolehkan menduplikasikan parameter. Jika diduplikasikan akan menyebabkan eror.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10  <h2>"use strict":</h2>
11  <h3>menghapus sebuah function, tidak diperbolehkan</h3>
12
13  <script>
14    "use strict";
15    function c(p5, p6) {}; // Ini akan menyebabkan terjadinya eror
16  </script>
17 </body>
18 </html>
```

Gambar 2.37 Penggunaan mode strict menduplikasikan parameter

- Pada script javascript dengan menggunakan metode strict, anda tidak diperbolehkan memasukkan sebuah property yaitu read-only karena dapat menyebabkan eror.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10  <h2>"use strict":</h2>
11  <h3>Memasukkan properti read-only, tidak diperbolehkan</h3>
12
13  <script>
14    "use strict";
15    var obj = {};
16    Object.defineProperty(obj, "a", {value:1, writable:false});
17
18    obj.a = 3.14;
19  </script>
20 </body>
21 </html>
```

Gambar 2.38 Mode strict penggunaan property read only

- Pada script javascript dengan menggunakan metode strict, anda tidak diperbolehkan memasukkan sebuah property yaitu get-only karena dapat menyebabkan eror.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10  <h2>"use strict":</h2>
11  <h3>Memasukkan properti get-only, tidak diperbolehkan</h3>
12
13  <script>
14    "use strict";
15    var obj = {get a() {return 0} };
16
17    obj.a = 3.14; //akan menyebabkan error
18  </script>
19 </body>
20 </html>
```

Gambar 2.39 Mode strict penggunaan property read only

- Pada script javascript dengan menggunakan metode strict, anda tidak diperbolehkan menghapus properti. Berikut merupakan salah satu contohnya.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10  <h2>"use strict":</h2>
11  <h3>Menghapus properti yang tidak dapat dihapus tidak diperbolehkan</h3>
12
13  <script>
14    "use strict";
15    delete Object.prototype; // ini akan menyebabkan eror
16  </script>
17 </body>
18 </html>
```

Gambar 2.40 Penggunaan mode strict menghapus properti

- Pada script javascript dengan menggunakan metode strict, anda tidak diperbolehkan menggunakan kata eval sebagai variabel. Berikut merupakan salah satu contohnya.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10  <h2>"use strict":</h2>
11  <h3>>Kata eval tidak dapat digunakan sebagai variabel</h3>
12
13  <script>
14    "use strict";
15    var eval = 22/7; //ini akan menyebabkan eror
16  </script>
17 </body>
18 </html>
```

Gambar 2.41 Penggunaan mode stric kata eval

- Pada script javascript dengan menggunakan metode strict, anda tidak diperbolehkan menggunakan variabel dengan argument kata. Berikut merupakan salah satu contohnya.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10  <h2>"use strict":</h2>
11  <h3>Argumen kata tidak dapat digunakan sebagai variabel</h3>
12
13  <script>
14    "use strict";
15    var arguments = 3.14; // ini akan menyebabkan error
16  </script>
17 </body>
18 </html>
```

Gambar 2.42 Penggunaan mode strict menggunakan argument kata

- Pada script javascript dengan menggunakan metode strict, anda tidak diperbolehkan menggunakan pernyataan with . Berikut merupakan salah satu contohnya.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10  <h2>"use strict":</h2>
11  <h3>Pernyataan with tidak diperbolehkan</h3>
12
13  <script>
14    "use strict";
15    with (Math){x = cos(2)};
16  </script>
17 </body>
18 </html>
```

Gambar 2.43 Penggunaan mode strict pernyataan with

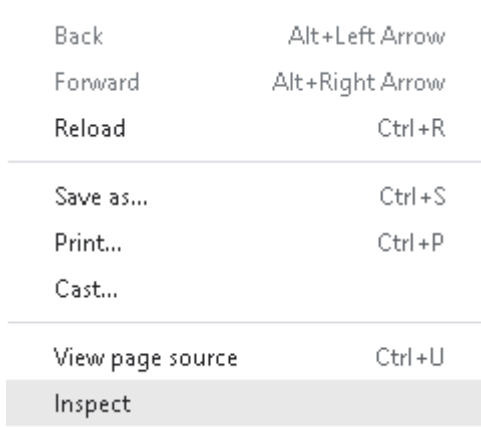
Object pada Browser

Penggunaan javascript, untuk membangun sebuah website. Tentu saja diperlukan agar website yang anda buat menjadi lebih interaktif dan fungsional. Dalam membangun website, bahasa javascript harus bisa mengontrol element element yang ada di dalam website tersebut. Bahasa javascript dalam mengontrol sebuah website, dibutuhkan object yang sudah tersedia pada browser yaitu window.

Dalam bahasa pemrograman javascript, browser merupakan sebuah object windows. Pada object window memiliki banyak method dan properties untuk membantu membuat sebuah website, salah satu contohnya yaitu alert().

Anda dapat melihat kumpulan method dan properties yang dimiliki object window pada browser. Berikut merupakan langkah-langkahnya.

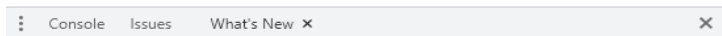
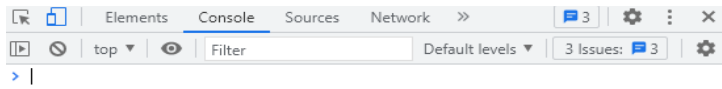
1. Buka browser yang anda miliki lalu klik kanan pada mouse dan jika tidak memakai mouse klik kanan pada tombol touchbar pada halaman pertama, akan muncul beberapa menu. Setelah itu klik inspect.



Back	Alt+Left Arrow
Forward	Alt+Right Arrow
Reload	Ctrl+R
<hr/>	
Save as...	Ctrl+S
Print...	Ctrl+P
Cast...	
<hr/>	
View page source	Ctrl+U
Inspect	

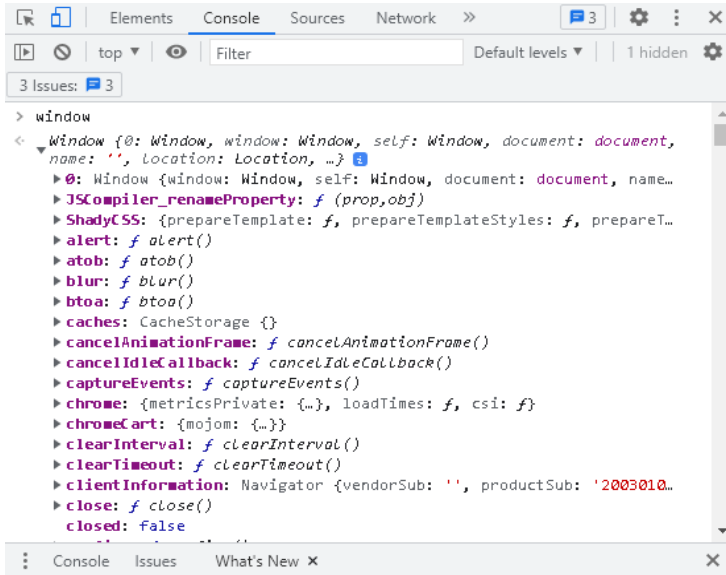
Gambar 2.44 Tampilan klik kanan pada touchbar atau mouse

2. Lalu, klik console pada halaman inspect.



Gambar 2.45 Tampilan halaman console pada browser

3. Lalu ketik window pada halaman console tersebut, dan akan muncul banyak properties, method, dan juga event. Berikut merupakan contohnya.



Gambar 2.46 Macam macam property dan method pada javascript

Pada gambar 2 44 terdapat banyak sekali property dan method pada object browser, syntax javascript bisa mengontrol browser. Anda juga tidak perlu menghafal semua property dan method pada object browser. Berikut merupakan property dan method yang sering digunakan untuk mengembangkan website.

Tabel 2.2 Macam macam property dan method yang sering digunakan

Property/Method	Deskripsi
alert()	Berfungsi sebagai menampilkan pesan peringatan jika user website tidak mengikuti perintah dalam website.
confirm()	Berfungsi sebagai menampilkan pesan konfirmasi jika user melakukan tindakan tertentu pada website.
close()	Berfungsi sebagai menutup tab pada halaman website.
history	Berfungsi untuk melakukan navigasi history pada URL browser.
prompt()	Berfungsi untuk menampilkan pesan untuk user website dan teks input. Dan fungsi dari method ini adalah mengembalikan nilai string sesuai dengan response dari user website.
location	Berfungsi sebagai mengambil URL pada address bar browser.

- Akses property atau method lewat window



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10  <h1>Contoh akses object window</h1>
11
12  <script>
13    // mengakses alert melalui window
14    window.alert("Halo joko")
15
16  </script>
17 </body>
18 </html>
```

Gambar 2.47 Mengakses property atau method melalui window

- Akses property atau method tanpa memasukkan window



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10  <h1>Contoh akses object window</h1>
11
12  <script>
13    // mengakses alert melalui window
14    alert("Halo joko")
15
16  </script>
17 </body>
18 </html>
```

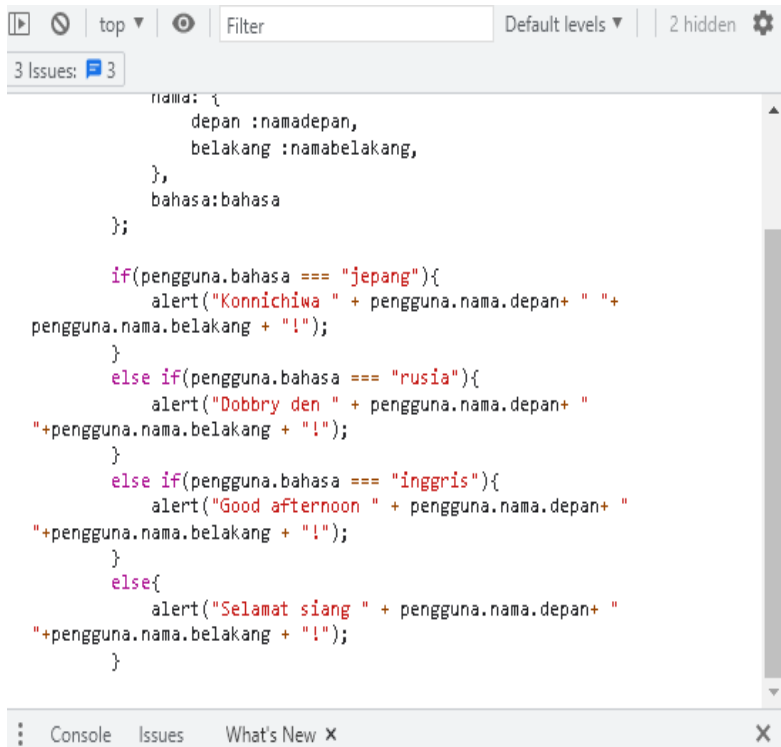
Gambar 2.48 Mengakses property atau method tanpa melalui window

Sekarang kita coba implementasikan untuk membuat sebuah program sederhana dari penjelasan pada object browser dan beberapa fundamental atau dasar dasar dari bahasa pemrograman javascript. Disini kita membuat program javascript menggunakan property dan method dari object window yaitu alert() dan prompt(). Berikut merupakan contoh programnya.

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10   <script>
11     const namadepan = prompt("Masukan nama depan");
12     const namabelakang = prompt("Masukan nama belakang");
13     const bahasa = prompt("Bahasa apa yang anda bisa");
14
15     const pengguna = {
16       nama: {
17         depan :namadepan,
18         belakang :namabelakang,
19       },
20       bahasa:bahasa
21     };
22
23     if(pengguna.bahasa === "jepang"){
24       alert("Konnichiwa " + pengguna.nama.depan+ " "+ pengguna.nama.belakang + "!");
25     }
26     else if(pengguna.bahasa === "rusia"){
27       alert("Dobbry den " + pengguna.nama.depan+ " "+pengguna.nama.belakang + "!");
28     }
29     else if(pengguna.bahasa === "inggris"){
30       alert("Good afternoon " + pengguna.nama.depan+ " "+pengguna.nama.belakang + "!");
31     }
32     else{
33       alert("Selamat siang " + pengguna.nama.depan+ " "+pengguna.nama.belakang + "!");
34     }
35   </script>
36 </body>
37 </html>
```

Gambar 2.49 Script javascript menggunakan object browser

Pada script javascript 2.47 kita memakai logika if else untuk membuat bahasa yang digunakan dan isi dari logika if else tersebut adalah alert yang berfungsi untuk menampilkan data. Setelah itu kita copy isi dari element script ke console browser untuk menjalankan program.

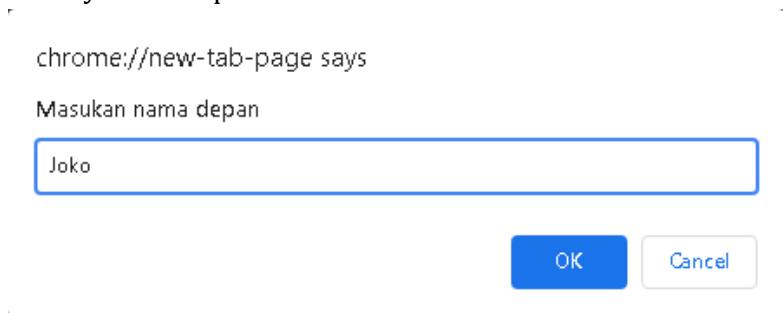


```
nama: {
  depan :namadepan,
  belakang :namabelakang,
},
bahasa:bahasa
});

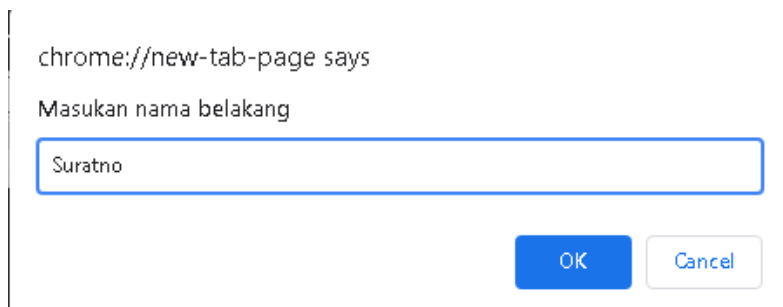
if(pengguna.bahasa === "jepang"){
  alert("Konnicchiwa " + pengguna.nama.depan+ " "+
pengguna.nama.belakang + "!");
}
else if(pengguna.bahasa === "rusia"){
  alert("Dobbry den " + pengguna.nama.depan+ "
"+pengguna.nama.belakang + "!");
}
else if(pengguna.bahasa === "inggris"){
  alert("Good afternoon " + pengguna.nama.depan+ "
"+pengguna.nama.belakang + "!");
}
else{
  alert("Selamat siang " + pengguna.nama.depan+ "
"+pengguna.nama.belakang + "!");
}
```

Gambar 2.50 Script javascript menggunakan object browser siap untuk dijalankan

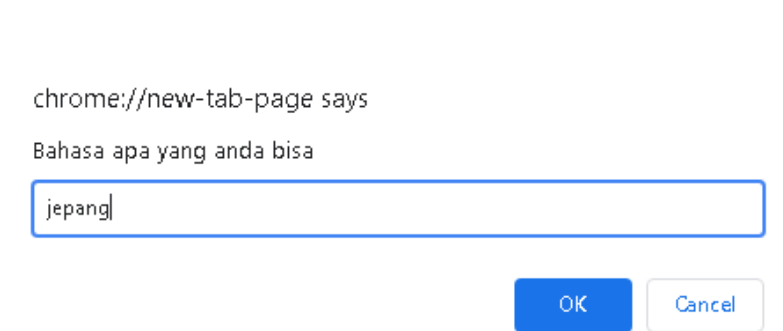
Setelah itu kita jalankan program pada browser tersebut dan hasilnya akan seperti ini.



Gambar 2.51 Hasil script ketika dijalankan halaman input pertama



Gambar 2.52 Hasil script ketika dijalankan halaman inputan kedua



Gambar 2.53 Hasil script ketika dijalankan halaman inputan ketiga

Dan hasil akhirnya akan menjadi seperti ini

```
chrome://new-tab-page says
```

```
Konnichiwa Joko Suratno!
```



Gambar 2.54 Hasil akhir script javascript object browser

DOM(Document Object Model)

Pada sub bab sebelumnya kita telah mempelajari tentang object window. Pada sub bab ini kita akan mempelajari tentang DOM Scripting. Yang akan kita pelajari adalah tentang atribut elemen, teks Nampak pada halaman website, dan dapat memanipulasi element.

Pengertian DOM

DOM (Document Object Model) merupakan sebuah cara pada bahasa javascript untuk melihat dokumen pada html. DOM memiliki sebuah interface dan platform yang berfungsi sebagai mengubah konten pada html, struktur html, dan style pada program html dan script. Pada sebuah web anda tidak diwajibkan untuk menggunakan DOM yang berfungsi sebagai menampilkan program html.

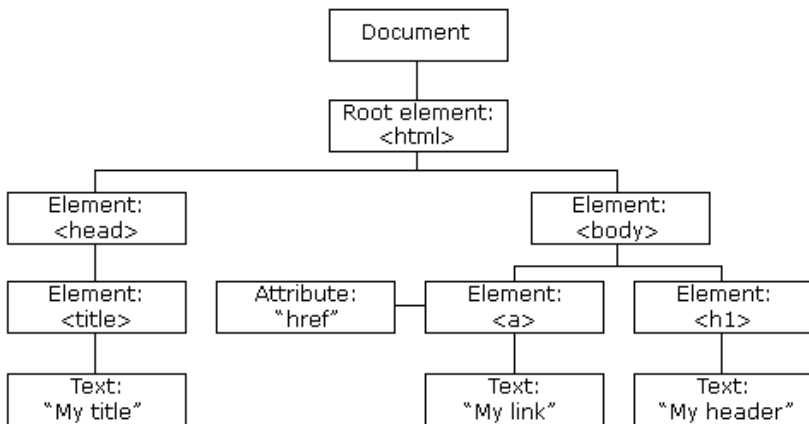
Sejarah DOM

Pada tahun 1990-an perusahaan bernama W3C membangun DOM. Selanjutnya pada tahun 1998 tepatnya pada bulan Oktober perusahaan W3C meluncurkan DOM yaitu bernama DOM 1. Pada pembuatan DOM oleh perusahaan W3C dibagi menjadi beberapa tingkatan. Pada suatu halaman yang mendukung dalam suatu level, harus bisa dapat menerapkan

semua persyaratan pada level tersebut dan juga pada level yang ada dibawahnya.

Sebelum perusahaan W3C memiliki standarisasi DOM, pada masing masing website yang ada di browser memiliki DOM sendiri pada web tersebut. Akibat dari memiliki masing masing memiliki DOM sendiri adalah apabila sebuah website ingin menampilkan dengan cara cross-browser compatible, ia harus dirancang untuk setiap web yang akan didukung. DOM dalam versi standart akan mempermudah dalam mengembangkan website tersebut.

Struktur DOM



Gambar 2.55 Struktur DOM

Dalam penggunaan DOM bahasa pemrograman javascript anda dapat mengetahui :

- Pada semua element yang ada pada HTML merupakan sebuah object
- Pada DOM, HTML dapat kita definisikan sebuah property dari element element yang ada pada HTML.

- Pada DOM, HTML dapat kita definisikan bahwa method dapat berfungsi sebagai mengakses semua element element yang ada pada HTML.
- Pada DOM, HTML dapat kita definisikan bahwa semua event adalah untuk semua element element yang ada pada HTML.

Pada element-element yang berada pada HTML dapat kita atur menggunakan DOM. DOM juga bisa memanipulasi seluruh element HTML melalui bahasa pemrograman javascript. Berikut merupakan salah satu contoh Bahasa pemrograman javascript DOM adalah **document.getElementById**.

Fungsi dari syntax `document.getElementById` adalah mencari id pada script HTML. Selain, `document.getElementById` dalam DOM bahasa pemrograman javascript tersedia fungsi fungsi yang lain diantaranya `document.getElementByName`, `document.getElementsByClassName`, dan masih banyak lagi. Fitur fitur pada DOM inilah yang akan kita pelajari pada sub bab ini.

DOM (Document Object Model) Selector

Pada sub bab kali ini akan membahas macam macam selector pada DOM bahasa pemrograman javascript. Berikut merupakan beberapa selector dari DOM.

- Method `getElementsByTagName`
Pada selector ini Berfungsi sebagai cara mengakses sebuah element pada script html dengan berdasarkan pada tag html. Setelah mengakses DOM akan menyeleksi semua element yang ditentukan dan akan menghasilkan sebuah document HTML. Collection atau dalam bentuk sederhananya adalah semua kumpulan element element yang ada pada html memiliki sebuah index, nilai index

collection sendiri dimulai dari angka 0. Berikut merupakan salah satu contoh penerapan Method `getElementsByTagName`.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10  <h1>Judul 1</h1>
11  <h2>Judul 2</h2>
12  <script>
13    let h1 = document.getElementsByClassName('h1');
14    h1[0].innerHTML = 'ubah halaman judul 3';
15    h1[1].innerHTML = 'ubah halaman judul 4';
16  </script>
17 </body>
18 </html>
```

Gambar 2.56 Script javascript menggunakan Method `getElementsByTagName`

Dan Jika kita tampilkan pada browser maka akan menjadi seperti berikut ini.

ubah halaman judul 3

Judul 2

Gambar 2.57 Hasil script menggunakan method `getElementsByTagName`

- Method `getElementById`
Pada selector DOM ini memiliki fungsi yaitu dapat mengakses sebuah element element yang berada pada script html melalui id. Berikut merupakan contoh method `getElementById`.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10  <h2 id="sub">Halaman judul pertama</h2>
11  <script>
12    let sub = document.getElementById('sub');
13    sub.innerHTML = 'ubah halaman judul';
14  </script>
15 </body>
16 </html>
```

Gambar 2.58 Script javascript menggunakan Method `getElementById`

Jika kita tampilkan pada browser maka hasilnya akan menjadi seperti ini.

ubah halaman judul

Gambar 2.59 Hasil script javascript menggunakan Method `getElementById`

- Method `getElementsByName`
Pada Method ini berfungsi sebagai cara untuk mengakses elemnt yang berada pada script html berdasarkan dengan atribut class. Pada selector ini dapat menghasilkan html collection. Berikut merupakan salah satu contoh `getElementsByName`.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10  <h1 class="judul">judul 1</h1>
11  <h1 class="judul">judul 2</h1>
12  <script>
13    let h1 = document.getElementsByClassName('judul');
14    h1[0].style.color = 'blue';
15    h1[1].style.color = 'red';
16  </script>
17 </body>
18 </html>
```

Gambar 2.60 Script javascript menggunakan Method `getElementsByName`

Dan hasilnya akan menjadi seperti ini.

judul 1

judul 2

Gambar 2.61 Hasil script javascript dengan menggunakan Method `getElementsByClassName`

DOM (Document Object Model) Event

Pada sub bab kali ini akan membahas DOM Event, DOM Event merupakan perubahan yang terjadi pada element-element yang ada pada HTML. Jika kita implementasikan salah satu contohnya adalah jika kalian membangun sebuah website yang memiliki button atau tombol, dan kalian ingin memberikan sebuah pesan untuk user website ingin melanjutkan ke halaman selanjutnya atau tidak pada user tersebut. Berikut merupakan beberapa event pada bahasa pemrograman javascript.

- **OnClick**

OnClick merupakan sebuah event pada javascript yang berfungsi sebagai merubah suatu kejadian pada website ketika sebuah element pada html di klik. Berikut merupakan contoh script dari OnClick.

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10  <p>Contoh dari implementasi onclick</p>
11  <p id="demo" OnClick="myFunction()">Silahkan klik disini</p>
12
13  <script>
14    function myfunction(){
15      document.getElementById('demo').innerHTML="Anda silahkan klik disini";
16    }
17  </script>
18 </body>
19 </html>
```

Gambar 2.62 Script javascript penggunaan Onclick

- **OnChange**
OnChange merupakan sebuah event DOM pada javascript berfungsi sebagai merubah suatu kejadian yang terjadi pada element html yang berubah. Berikut merupakan contoh dari OnChange.

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10  <p>Silahkan memilih mobil dari list berikut ini</p>
11  <select id="mySelect" OnChange="myFunction()">
12    <option value="Pajero">Pajero</option>
13    <option value="Fortuner">Fortuner</option>
14    <option value="Xenia">Xenia</option>
15    <option value="Xpander">Xpander</option>
16  </select>
17  <p id="demo"></p>
18  <script>
19    function myFunction(){
20      var a = document.getElementById('mySelect').value;
21      document.getElementById("demo").innerHTML = "Kamu memilih "+a;
22    }
23  </script>
24 </body>
25 </html>
```

Gambar 2.63 Script javascript penggunaan OnChange

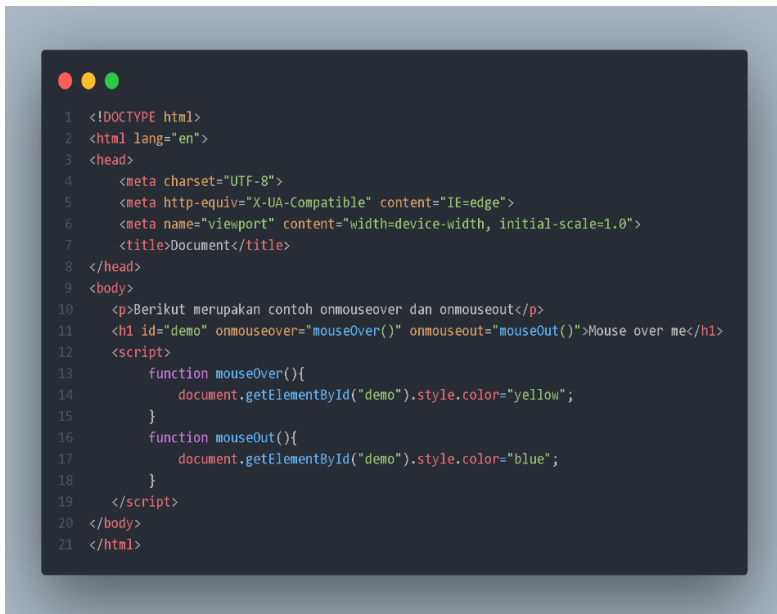
Dan hasil dari script diatas jika ditampilkan dalam browser maka seperti ini

Silahkan memilih mobil dari list berikut ini

Kamu memilih Xpander

Gambar 2.64 Hasil script javascript OnChange

- **OnMouseOver dan OnMouseOut**
OnMouseOver dan OnMouse Out merupakan sebuah kejadian yang terjadi pada element yang ada dalam html dan diletakkan cursor mouse. OnMouseOut merupakan sebuah kejadian dimana cursor mouse meninggalkan element pada html. Berikut merupakan contoh script dari OnMouseOver dan OnMouseOut.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10  <p>Berikut merupakan contoh onmouseover dan onmouseout</p>
11  <h1 id="demo" onmouseover="mouseover()" onmouseout="mouseout()">Mouse over me</h1>
12  <script>
13    function mouseOver(){
14      document.getElementById("demo").style.color="yellow";
15    }
16    function mouseOut(){
17      document.getElementById("demo").style.color="blue";
18    }
19  </script>
20 </body>
21 </html>
```

Gambar 2.65 Hasil script javascript OnMouseOver dan OnMouseOut

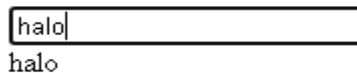
- **OnkeyDown**
OnkeyDown merupakan sebuah event DOM pada bahasa pemrograman javascript. OnkeyDown memiliki fungsi yaitu jika ada suatu kejadian ketika ada kegiatan pengetikan pada element html.

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body>
10  <p>Berikut merupakan contoh dari onkeydown</p>
11  <input type="text" id="input" onkeydown="myfunction()">
12  <div id="text"></div>
13  <script>
14    function myfunction(){
15      var inputText = document.getElementById("input").value;
16      document.getElementById("text").innerHTML = inputText;
17    }
18  </script>
19 </body>
20 </html>
```

Gambar 2.66 Script javascript OnkeyDown

Dan jika kita tampilkan pada browser maka hasilnya akan menjadi seperti berikut ini.

Berikut merupakan contoh dari onkeydown



Gambar 2.67 Hasil script javascript OnkeyDown

- OnLoad
Onload merupakan sebuah event DOM pada pemrograman javascript. Pada Onload ini ketika sebuah kejadian pada

user membuka halaman sebuah website. Berikut merupakan contoh script dari onload.



```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>Document</title>
8 </head>
9 <body onload="myFunction()">
10  <h1>Halo dunia</h1>
11  <script>
12    function myFunction(){
13      alert("Page is loaded");
14    }
15  </script>
16 </body>
17 </html>
```

Gambar 2.68 Script javascript menggunakan OnLoad

Dan hasil script diatas akan menjadi seperti ini.

This page says

Page is loaded



Gambar 2.69 Hasil script javascript menggunakan OnLoad

Membuat game dengan Bahasa Pemrograman Javascript

Setelah kita mempelajari dasar dasar pada bahasa pemrograman javascript. Selanjutnya kita akan membuat game menggunakan bahasa pemrograman javascript. Pada penggunaan game javascript ini, kita menggunakan html, css, dan javascript. Judul dari game yang akan kita buat adalah game pukul tikus tanah. Berikut merupakan game.

1. Hal yang pertama anda lakukan adalah menyiapkan file html, css dan javascript terlebih dahulu pastikan ke 3 file tersebut di dalam satu folder supaya mudah untuk meng codingnya.
2. Setelah itu buka file html anda yang akan dibuat game pukul tikus tanah dan kita akan memakai html 5. Berikut merupakan contohnya.

A screenshot of a code editor with a dark background and light-colored text. The code is HTML5 boilerplate for a game titled "Game tikus tanah". The code is numbered from 1 to 12. The code includes a DOCTYPE declaration, an HTML lang attribute, a head section with a title, a meta charset attribute, a meta viewport attribute, and a link to a style sheet. The body and html tags are also present.

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <title>Game tikus tanah</title>
5     <meta charset="UTF-8">
6     <meta name="viewport" content="width=device-width, initial-scale=1">
7     <link href="style.css" rel="stylesheet">
8   </head>
9   <body>
10
11 </body>
12 </html>
```

Gambar 2.70 Dokumen html pembuatan game

3. Selanjutnya kita mengisi element body pada script html yaitu h1, div dengan class yaitu container di dalamnya kita akan kasi div dengan kelas tanah, dan di dalam div kelas tanah kita tambahkan div dengan kelas tikus. Berikut merupakan contohnya.

A screenshot of a code editor with a dark background and light-colored text. The code is HTML and is numbered from 1 to 20. It shows the structure of an HTML document for a game page. The code includes a DOCTYPE declaration, an HTML lang attribute, a head section with a title, meta charset, meta viewport, and a link to a style sheet. The body section contains an h1 element and a nested div structure with classes 'container', 'tanah', and 'tikus'.

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <title>Game tikus tanah</title>
5     <meta charset="UTF-8">
6     <meta name="viewport" content="width=device-width, initial-scale=1">
7     <link href="style.css" rel="stylesheet">
8   </head>
9   <body>
10    <h1>Pukul Tikus Tanah</h1>
11
12    <div class="container">
13      <div class="tanah">
14        <div class="tikus">
15
16          </div>
17        </div>
18      </div>
19    </body>
20 </html>
```

Gambar 2.71 Membuat div pada html

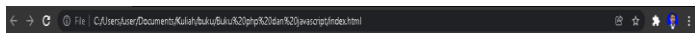
4. Setelah itu kita masuk ke dalam file style.css untuk mengedit halaman website. Pada file css kita mulai mengedit pada element h1.

A screenshot of a code editor window with a dark background and three colored window control buttons (red, yellow, green) at the top left. The code is as follows:

```
1  h1{
2    font-size: 65px;
3    font-family: sans-serif;
4    text-transform: uppercase;
5    text-align: center;
6  }
```

Gambar 2.72 Membuat style untuk element h1

Jika kita tampilkan pada browser maka akan menjadi seperti ini.



PUKUL TIKUS TANAH

Gambar 2.73 Hasil dari style h1

5. Setelah itu kita membuat style element container sebagai pembungkus konten pada html.

A code editor window with a dark background and three colored window control buttons (red, yellow, green) at the top left. The code is as follows:

```
1  .container{  
2      margin: auto;  
3      width: 600px;  
4      border: 1px solid black;  
5  }
```

Gambar 2.74 Style pada div class container

Jika kita tampilkan pada browser maka akan menjadi seperti ini.



Gambar 2.75 Hasil dari style container

- Setelah itu kita menulis class tanah pada file css disini kita menuliskan class tanah dan class tanah dengan diberi after berikut merupakan contohnya

A screenshot of a code editor with a dark background and light-colored text. The code is CSS for a class named '.tanah'. It includes a main rule for '.tanah' and an 'after' pseudo-element rule. The main rule sets width to 500px, height to 200px, and position to relative. The 'after' rule sets content to an empty string, width to 200px, height to 100px, display to block, background to a URL (gambartanah.png) with bottom center no-repeat, background-size to 60%, position to absolute, and bottom to 0. The code is numbered from 1 to 15.

```
1 .tanah{
2     width: 500px;
3     height: 200px;
4     position: relative;
5 }
6 .tanah::after{
7     content: '';
8     width: 200px;
9     height: 100px;
10    display: block;
11    background: url(gambartanah.png) bottom center no-repeat;
12    background-size: 60%;
13    position: absolute;
14    bottom: 0;
15 }
```

Gambar 2.76 Style div class tanah

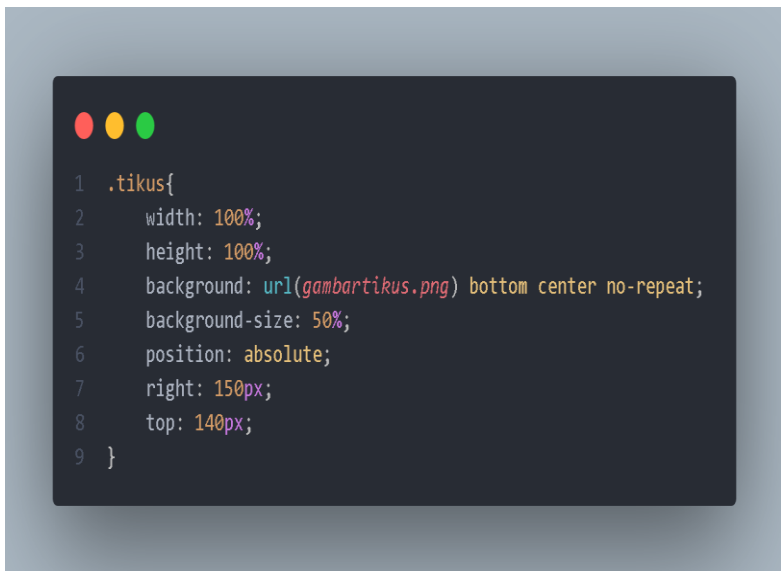
Dan jika kita tampilkan pada browser maka akan menjadi seperti ini.

PUKUL TIKUS TANAH



Gambar 2.77 Hasil styling div tanah

7. Selanjutnya kita akan menuliskan style tikus dan penambahan value pada class tanah berikut merupakan contohnya.



Gambar 2.78 Style div tikus

Jika kita script 2.78 ditampilkan maka akan menjadi seperti ini

PUKUL TIKUS TANAH



Gambar 2.79 Hasil styling div tikus

Pada hasil script diatas kita mengatur gambar tikus pada bagian bawah gambar tanah, selanjutnya kita mengatur supaya gambar tikus hilang. Berikut merupakan scriptnya.

```
1 tanah{
2     width: 500px;
3     height: 200px;
4     position: relative;
5     overflow: hidden;
6 }
```

Gambar 2.80 styling tanah dengan penambahan value overflow

Pada style element tanah kita menambahkan value overflow dengan nilai hidden untuk menghilangkan gambar tikus dari kotak.

PUKUL TIKUS TANAH



Gambar 2.81 Hasil jika ditambahkan overvlow dengan value hidden

8. Jika kita ingin tampilkan tikus diatas tanah. Berikut merupakan contohnya.



```
1 .tanah.muncul .tikus{
2   top : 0;
3 }
```

Gambar 2.82 Script styling tikus untuk muncul diatas tanah

Jika kita tampilkan pada browser maka hasilnya akan seperti ini.

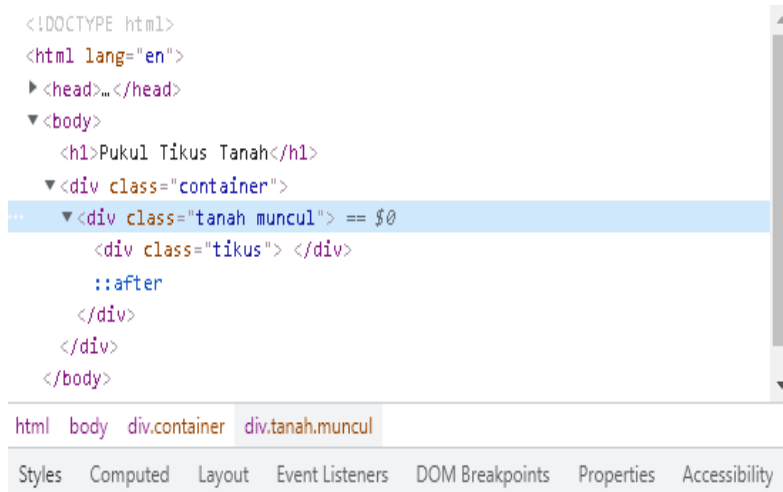
PUKUL TIKUS TANAH



Gambar 2.83 Hasil styling div tanah dan div tikus

Disini hasilnya tidak menunjukkan perubahan. Karena kita belum menuliskan class muncul pada tanah. Cara untuk

melihat hasilnya kita menuju ke inspect. Berikut merupakan caranya.



```
<!DOCTYPE html>
<html lang="en">
  <head>...</head>
  <body>
    <h1>Pukul Tikus Tanah</h1>
    <div class="container">
      <div class="tanah muncul" == $0
        <div class="tikus"> </div>
        ::after
      </div>
    </div>
  </body>
```

html body div.container div.tanah.muncul

Styles Computed Layout Event Listeners DOM Breakpoints Properties Accessibility

Gambar 2.84 Menuliskan class muncul pada tanah

Pada class tanah kita tambahkan class muncul. Berikut merupakan hasilnya.



Gambar 2.85 Hasil setelah diberi nilai muncul pada tanah

- Setelah itu kita tambahkan value transition untuk menambahkan animasi ketika tikus muncul berikut merupakan contohnya.

A screenshot of a code editor with a dark background and light-colored text. The code is CSS for a class named '.tikus'. It includes properties for width, height, background image, background size, position, and transition. The transition property is 'top 0.3s'.

```
1 .tikus{
2   width: 100%;
3   height: 100%;
4   background: url(gambartikus.png) bottom center no-repeat;
5   background-size: 50%;
6   position: absolute;
7   right: 150px;
8   top: 140px;
9   transition: top 0.3s;
10 }
```

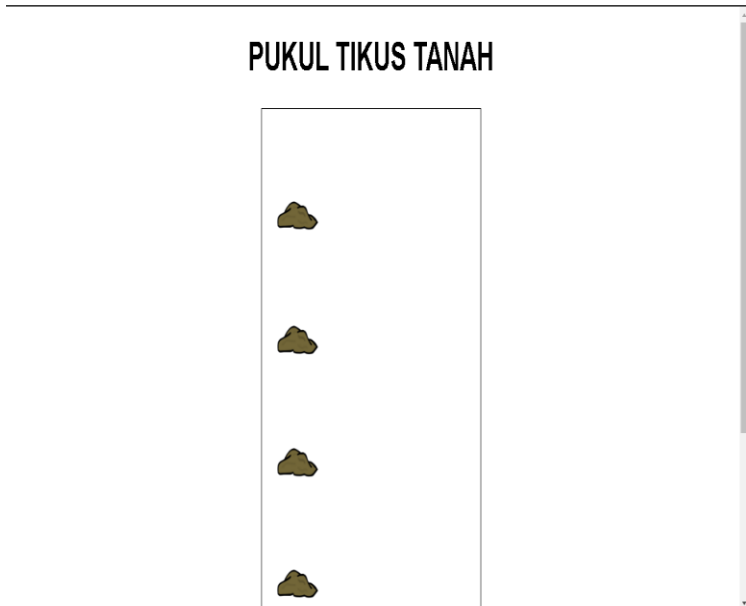
Gambar 2.86 Penambahan value transition pada class tikus

- Kita kembali pada halaman html dan menulis ulang div tanah dan div tikus. Dan juga kita menuliskan element script untuk menghubungkan file html dan file javascript. Berikut merupakan contohnya.

```
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <title>Game tikus tanah</title>
5     <meta charset="UTF-8">
6     <meta name="viewport" content="width=device-width, initial-scale=1">
7     <link href="style.css" rel="stylesheet">
8   </head>
9   <body>
10    <h1>Pukul Tikus Tanah</h1>
11
12    <div class="container">
13      <div class="tanah">
14        <div class="tikus">
15          </div>
16      </div>
17      <div class="tanah">
18        <div class="tikus">
19          </div>
20      </div>
21      <div class="tanah">
22        <div class="tikus">
23          </div>
24      </div>
25      <div class="tanah">
26        <div class="tikus">
27          </div>
28      </div>
29      <div class="tanah">
30        <div class="tikus">
31          </div>
32      </div>
33      <div class="tanah">
34        <div class="tikus">
35          </div>
36      </div>
37    </div>
38    <script src="main.js"></script>
39  </body>
40 </html>
```

Gambar 2.87 Penulisan div pada html

Dan hasil dari script diatas jika ditampilkan pada browser akan menjadi seperti ini.



Gambar 2.88 Hasil dari penulisan div ulang

Pada hasil diatas gambar tanah terlihat tegak lurus sekarang kita mau ubah posisi men datar. Berikut merupakan contoh script.

```
1  .tanah{
2      width: 200px;
3      height: 200px;
4      position: relative;
5      overflow: hidden;
6      float: left;
7  }
```

Gambar 2.89 Menambahkan value float pada style tanah

Jika kita tampilkan pada browser maka akan menjadi seperti ini.

PUKUL TIKUS TANAH




Gambar 2.90 Hasil penambahan value float dengan class tanah

11. Selanjutnya kita mengisi file javascript dan disini kita akan langsung membuat tanah dan tikus lebih interaktif. Berikut merupakan contoh scripnya.

```
1 const tanah = document.querySelectorAll('.tanah');
2 const tikus = document.querySelectorAll('.tikus');
3
4 let tanahSebelumnya;
5
6 function randomTanah(tanah) {
7     const t = Math.floor(Math.random() * tanah.length);
8     const tRandom = tanah[t];
9     if (tRandom == tanahSebelumnya) {
10         randomTanah(tanah);
11     }
12     tanahSebelumnya = tRandom;
13     return tRandom;
14 }
15 function munculkanTikus() {
16     const tRandom = randomTanah(tanah);
17     const wRandom = randomWaktu(300, 1000);
18     tRandom.classList.add('muncul');
```

Gambar 2.91 Script javascript untuk tanah dan tikus

12. Selanjutnya kita mengatur waktu kemunculan tikus. Berikut merupakan contoh script javascript.



```
1  setTimeout(() => {
2      tRandom.classList.remove('muncul');
3      if (!selesai) {
4          munculkanTikus();
5      }
6  }, wRandom);
7  }
```

Gambar 2.92 Script mengatur munculnya tikus

13. Selanjutnya kita membuat sebuah rumus untuk menentukan waktu minimal dan maksimal untuk keluar si tikus. Berikut merupakan contohnya.



```
1 function randomWaktu(min, max) {
2   return Math.round(Math.random() * (max - min) + min);
3 }
```

Gambar 2.93 Script untuk mengatur waktu minimal dan maksimal

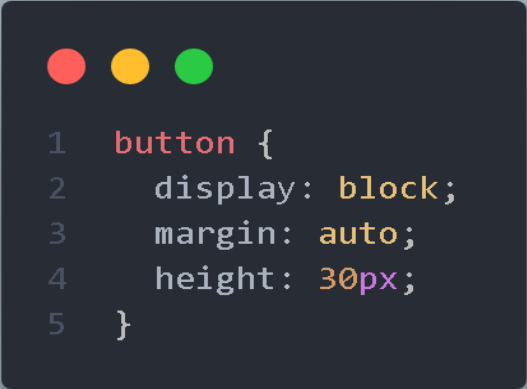
14. Selanjutnya kita membuat tombol untuk memulai gamenya. Berikut merupakan contoh scriptnya.



```
1 <button type="button" class="mulai" onclick="mulai()">Start</button>
```

Gambar 2.94 Membuat tombol start pada html untuk memulai game

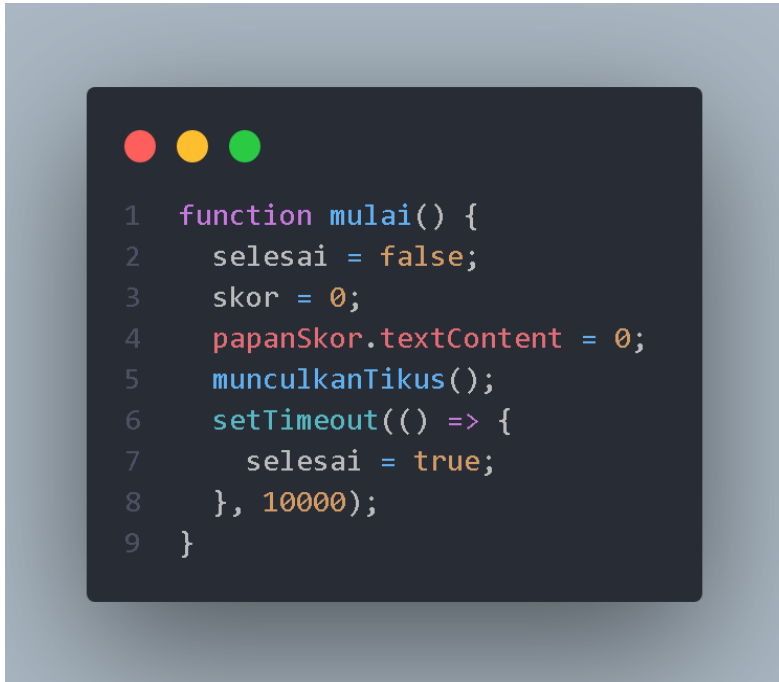
15. Selanjutnya kita style element button pada html. Berikut merupakan contohnya



```
1  button {
2    display: block;
3    margin: auto;
4    height: 30px;
5  }
```

Gambar 2.95 Styling element button

16. Selanjutnya, kita akan memberi fungsi kepada button. Berikut merupakan contohnya.

A screenshot of a code editor with a dark background and light-colored text. The code is a JavaScript function named 'mulai()' with the following lines:

```
1 function mulai() {
2   selesai = false;
3   skor = 0;
4   papanSkor.textContent = 0;
5   munculkanTikus();
6   setTimeout(() => {
7     selesai = true;
8   }, 10000);
9 }
```

The code is numbered from 1 to 9 on the left side. The function sets a 'selesai' variable to false, a 'skor' variable to 0, and sets the 'textContent' of 'papanSkor' to 0. It then calls 'munculkanTikus()' and sets a timeout for 10000 milliseconds to set 'selesai' to true.

Gambar 2.96 Membuat sebuah fungsi pada javascript

Dan jangan lupa untuk memberikan sebuah variabel selesai berikut merupakan contohnya.



Gambar 2.97 Memberikan variabel untuk memulai

17. Selanjutnya kita akan membuat papan skor terlebih dahulu. Kita membuat dari file html terlebih dahulu berikut merupakan contohnya.



Gambar 2.98 Membuat papan skor untuk game

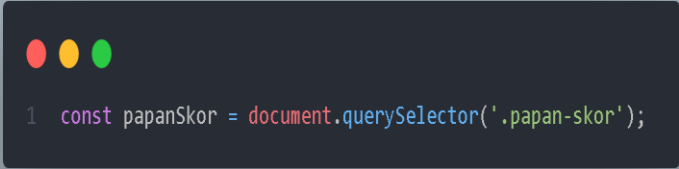
18. Setelah itu kita memberi style pada element h2 tersebut berikut merupakan contohnya.

A code editor window with a dark background and three colored window control buttons (red, yellow, green) at the top left. The code is as follows:

```
1 h1, h2 {  
2   font-size: 60px;  
3   text-align: center;  
4   font-family: sans-serif;  
5   text-transform: uppercase;  
6   margin: 10px;  
7   color: #fff;  
8 }
```

Gambar 2.99 Memberi style pada element h1 dan h2


19. Setelah itu kita menuju ke file javascript dan memasukkan queryselector. Berikut merupakan contohnya.



```
1 const papanSkor = document.querySelector('.papan-skor');
```

Gambar 2.100 Membuat query pada class papan-skor

20. Selanjutnya, kita membuat fungsi pukul yang berfungsi sebagai ketika tikus dipukul akan menambah skor. Berikut merupakan contohnya.



```
1 function mulai() {
2   selesai = false;
3   skor = 0;
4   papanSkor.textContent = 0;
5   munculkanTikus();
6   setTimeout(() => {
7     selesai = true;
8   }, 10000);
9 }
10
11 function pukul() {
12   skor++;
13   this.parentNode.classList.remove('muncul');
14   papanSkor.textContent = skor;
15 }
```

Gambar 2.101 membuat script function pukul

21. Selanjutnya pastikan semua variabel terdeklarasi.
Berikut merupakan contohnya.



```
1 let tanahSebelumnya;  
2 let selesai;  
3 let skor;
```

Gambar 2.102 Memastikan semua variabel terdeklarasi

22. Selanjutnya kita membuat ketika masuk ke game si cursor akan merubah jadi palu disini kita siapkan gambar palu terlebih dahulu. Setelah kita siapkan gambar palu. Kita membuka file css dan berikut merupakan contoh scriptnya.

A screenshot of a code editor with a dark background and light-colored text. The code is CSS for a class named '.container'. It is numbered from 1 to 5. The code sets the width to 600px, the margin to auto, and the cursor to a URL pointing to 'palu1.png', with a fallback to 'auto'.

```
1  .container {  
2    width: 600px;  
3    margin: auto;  
4    cursor: url(palu1.png), auto;  
5  }
```

Gambar 2.103 Script untuk membuat cursor ketika masuk ke game menjadi palu

23. Setelah kita mengganti cursor menjadi gambar palu. Selanjutnya kita ingin memberi efek kepada palu tersebut ketika ingin memukul palu tersebut. Berikut merupakan contohnya.



```
1 .container:active {  
2   cursor: url(palu2.png), auto;  
3 }
```

Gambar 2.104 Script untuk memberikan efek palu ketika pukul tikus

Dan hasil akhirnya akan menjadi seperti ini.



Gambar 2.105 Hasil akhir game pemrograman javascript



PHP(Hypertext Preprocessor)

Pengertian PHP

PHP merupakan sebuah bahasa pemrograman yang berjalan pada Server Side Scripting maksudnya adalah bahasa ini tentu tidak dapat ditampilkan ke halaman website browser dan juga bisa untuk membuat game berbasis website berbeda dengan bahasa pemrograman Javascript yang berjalan pada Client Side Scripting.

Contohnya kita membuat program menghitung pada website dan ingin mengalikan tiga bilangan dengan menggunakan bahasa pemrograman PHP. Sebelumnya kalian membuat dulu form untuk menghitung perkalian dengan tiga bilangan tersebut. Setelah itu anda membuat script pemrograman PHP untuk mengelola tiga bilangan tersebut. Dan sekarang hasil akhir dari form yang kita buat menggunakan Client Side Scripting dan bahasa pemrograman PHP yang kita buat akan sama sama akan diupload ke dalam server.

Selanjutnya, jika user memakai website kita, user akan menuju ke halaman server kita dan posisi si user merupakan client. Selanjutnya si user akan memasukkan angka pada form yang kita buat dan fungsi dari bahasa PHP adalah mengelola atau menghitung angka angka tersebut. Dan menunjukkan hasil akhir dari perhitungan angka kepada si client atau user.

Kelebihan Bahasa PHP

1. Bahasa Pemrograman PHP Mudah untuk Dipelajari
Bahasa pemrograman PHP bukan bahasa yang memiliki sifat eksklusif dan juga bahasa ini mudah untuk dipelajari

dimana saja. Bahasa pemrograman ini termasuk ke dalam level yaitu entry level, yang dimaksud entry level ini adalah orang awam tentang pemrograman bisa untuk mempelajari bahasa pemrograman tersebut. Apalagi pada jaman sekarang dunia internet semakin berkembang, semakin mudah untuk mempelajari bahasa pemrograman PHP.

2. Bahasa Pemrograman PHP Bersifat Open Source
Bahasa Pemrograman PHP adalah bahasa yang berjenis open source. Arti dari open source sendiri adalah bahasa pemrograman php bisa digunakan untuk membuat aplikasi maupun website. Developer dari bahasa PHP ini tidak akan meminta biaya jika ingin menggunakan bahasa PHP ini.
3. Bahasa pemrograman PHP memiliki fitur upgrade yang sangat mudah
Dalam bahasa PHP jika PHP memiliki versi terbaru maka bahasa tersebut akan mengalami pembaruan secara otomatis dan akan melakukan proses kompilasi.
4. Bahasa Pemrograman PHP memiliki komunitas yang sangat besar
Bahasa PHP memiliki komunitas belajar yang sangat besar, komunitas bahasa ini tidak hanya di dalam negeri saja. Tetapi di luar negeri juga banyak komunitas PHP lainnya. Dari banyaknya komunitas tersebut anda bisa belajar bahasa tersebut mulai dasar sampai ahli.
5. Bahasa Pemrograman PHP memiliki Banyak Peluang untuk Mencari Pekerjaan
Bahasa pemrograman PHP bersifat open source yaitu bisa membuat aplikasi dan website. Dengan begitu banyak perusahaan yang membutuhkan programmer dengan bahasa ini.

Kekurangan Bahasa PHP

Kelebihan yang dimiliki bahasa pemrograman PHP memiliki banyak kelebihan. Meskipun banyak kelebihan, bahasa pemrograman ini memiliki kekurangan juga. Berikut merupakan kekurangan dari bahasa pemrograman PHP.

1. Memiliki Saingan yang Sangat Banyak

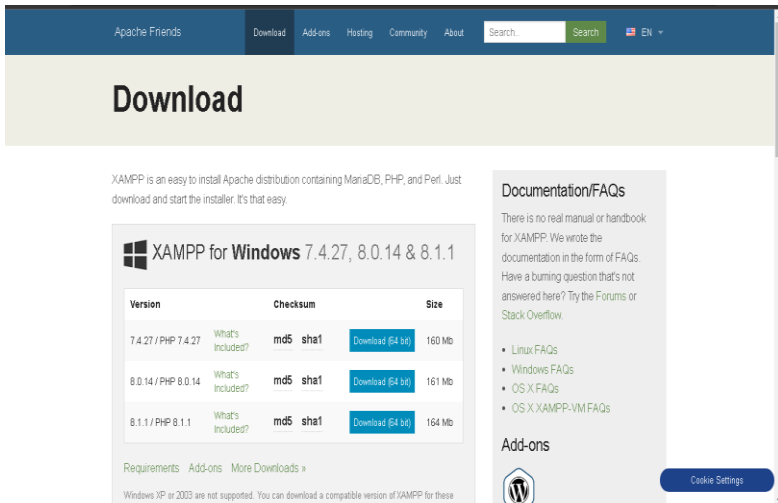
Pada bahasa pemrograman PHP ini sangat mudah untuk dipahami bahkan orang awam pun tentu bisa mempelajari bahasa ini. Sehingga bahasa pemrograman ini memiliki banyak pengguna. Dan jika anda mencari pekerjaan akan mendapatkan saingan yang sangat banyak dan juga ketat.

2. Program PHP Mudah Ditiru oleh Orang Lain

Pada bahasa pemrograman PHP ini, jika kita membuat sebuah program dengan bahasa ini sangat mudah untuk ditiru oleh orang lain dan dimodifikasi isi dari website tersebut. Supaya program anda yang buat memiliki keamanan yaitu menambahkan fitur add on. Sayangnya, fitur ini berbayar.

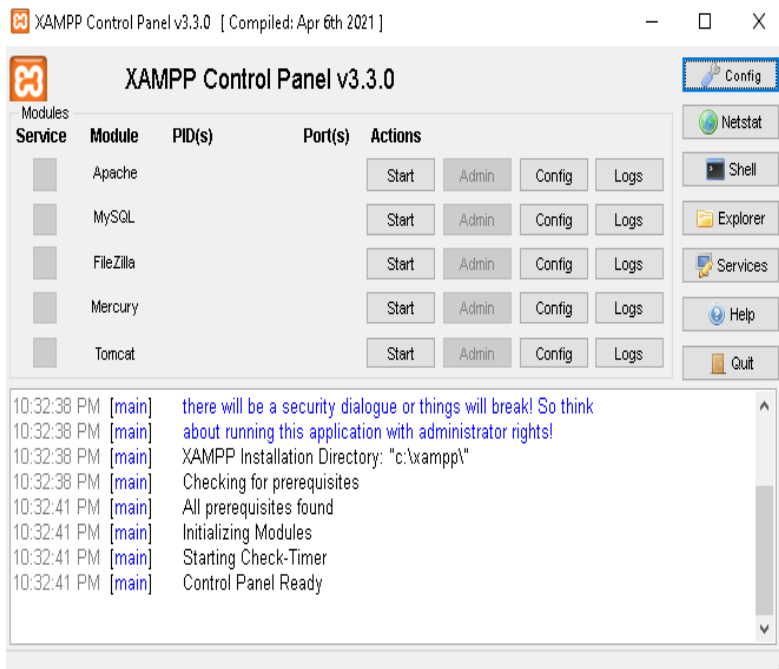
Cara Download dan Instalasi Xampp

Sebelum anda memulai untuk membuat program PHP, terlebih dahulu anda harus meng-install software xampp, xampp sendiri memiliki kelebihan yaitu bisa di install di semua sistem operasi baik itu windows, linux, dan mac os. Jika anda ingin mendownload software tersebut salin link ini <https://www.apachefriends.org/download.html>



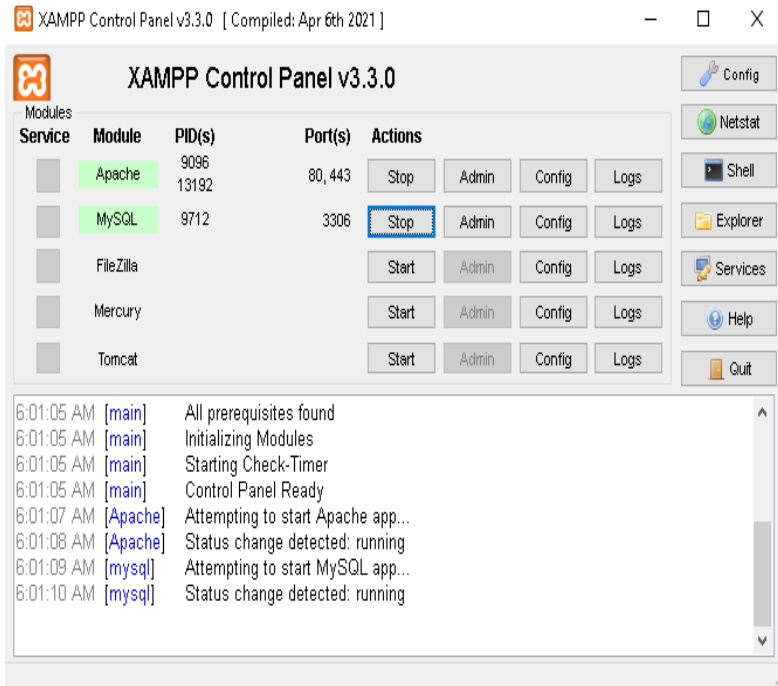
Gambar 3.1 Halaman download xampp

Kali ini kita akan mendownload PHP versi terbaru. Sebelum kalian mendownload anda harus tahu perangkat 64 bit atau 32 bit. Setelah itu kalian download dan jika selesai kalian buka aplikasi tersebut.



Gambar 3.2 Halaman control panel xampp

Pada tampilan ini kita meng klik tombol start pada Apache dan Mysql. Tunggu sampai apache dan mysql bewarna hijau. Berikut merupakan contohnya.



Gambar 3.3 Halaman control panel xampp start apache dan mysql

Setelah itu anda membuka browser kalian search localhost dan akan menuju ke halaman website seperti ini.



Welcome to XAMPP for Windows 8.0.8

You have successfully installed XAMPP on this system! Now you can start using Apache, MariaDB, PHP and other components. You can find more info in the FAQs section or check the HOW-TO Guides for getting started with PHP applications.

XAMPP is meant only for development purposes. It has certain configuration settings that make it easy to develop locally but that are insecure if you want to have your installation accessible to others. If you want have your XAMPP accessible from the internet, make sure you understand the implications and you checked the FAQs to learn how to protect your site. Alternatively you can use XIAMP, MAMP or LAMP which are similar packages which are more suitable for production.


Start the XAMPP Control Panel to check the server status.

Community

XAMPP has been around for more than 10 years – there is a huge community behind it. You can get involved by joining our Forums, adding

Gambar 3.4 halaman xampp pada browser

Disini bisa anda lihat bahwa php anda sudah ter install dan siap untuk membuat program dengan bahasa PHP. Jika kalian ingin melihat detail dari PHP anda bisa melalui PHPinfo. Berikut merupakan contohnya.

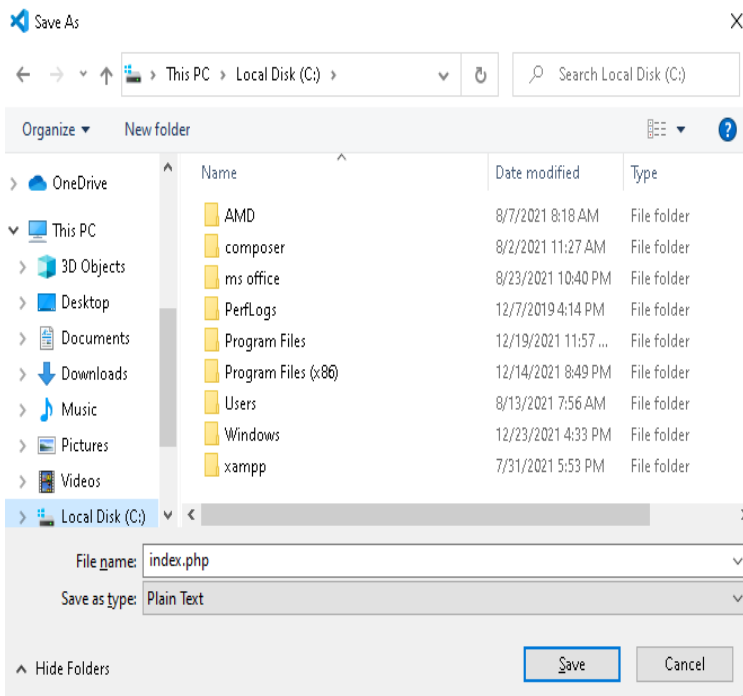
PHP Version 8.0.8 	
System	Windows NT DESKTOP-L59AKV3 10.0 build 19042 (Windows 10)AMD64
Build Date	Jun 29 2021 15:59:22
Build System	Microsoft Windows Server 2019 Datacenter [10.0.17763]
Compiler	Visual C++ 2019
Architecture	x64
Configure Command	script\phpize --script configure --enable-snmp --enable-sockets --enable-debug-pack --with-pdo-oci=1.11.1\instantclient_ods_19c --with-oci8=19c.1.1.1\instantclient_ods_19c --enable-object-out-dir=jobp --enable-com-dotnet-shared --without-analyzer --with-pgsql
Server API	Apache 2.0 Handler
Virtual Directory Support	enabled
Configuration File (php.ini) Path	no value
Loaded Configuration File	C:\xampp\php\php.ini
Scan this dir for additional .ini files	(none)
Additional .ini files parsed	(none)
PHP API	20200930
PHP Extension	20200930
Zend Extension	420200930
Zend Extension Build	API4.20200930,TS,VS16
PHP Extension Build	API20200930,TS,VS16
Debug Build	no
Thread Safety	enabled
Thread API	Windows Threads
Zend Signal Handling	disabled
Zend Memory Manager	enabled

Gambar 3.5 Halaman versi PHP dalam xampp

Struktur Dasar PHP

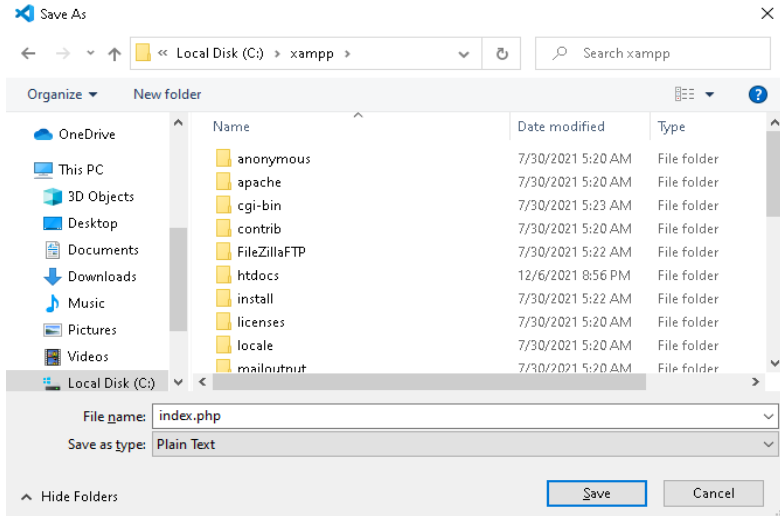
Pada sub bab ini kita akan mulai belajar dasar dari bahasa pemrograman PHP terlebih dahulu sebelum membuat sebuah game pada PHP. Yang pertama-tama kalian harus terlebih dahulu membuat file php terlebih dahulu. Untuk membuat file ini berbeda dengan html, css dan javascript yang menaruh file 3 program tersebut disembarang tempat, berbeda dengan php pembuatan file php dan harus ditempatkan di folder xampp yang kita buat. Berikut merupakan langkah-langkahnya.

1. Langkah pertama ialah membuat file php di dalam text editor yang kalian gunakan lalu klik save as atau tekan ctrl + s. Lalu kalian menuju ke Local Disk (C:). Setelah itu masuk ke dalam folder bernama xampp.



Gambar 3.6 Proses meletakkan dalam file melalui Local Disk(C:)

- Setelah itu anda cari folder bernama htdocs dan anda klik folder tersebut. Setelah itu anda save project anda pada folder tersebut.



Gambar 3.7 Proses meletakkan file php dalam folder xampp

Setelah kalian tahu pembuatan dan tata letak untuk menaruh file php, sekarang kita membuat program. Sekarang kita coba menuliskan hello word dalam bahasa pemrograman PHP. Dalam bahasa pemrograman PHP untuk menampilkan tulisan ada banyak cara. Berikut merupakan contohnya.

1. Cara yang pertama

A screenshot of a code editor with a dark background and light text. At the top left, there are three colored circles: red, yellow, and green. Below them, the code is as follows:

```
1 <?php
2     echo("Hello world");
3 ?>
```

Gambar 3.8 Cara menampilkan pada browser cara pertama

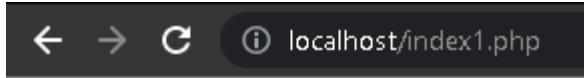
2. Cara yang kedua

A screenshot of a code editor with a dark background and light text. At the top left, there are three colored circles: red, yellow, and green. Below them, the code is as follows:

```
1 <?php
2     echo "Hello world";
3 ?>
```

Gambar 3.9 Cara menampilkan pada browser cara kedua

Jika anda ingin melihat hasil dari program php anda berbeda dengan cara melihat program html anda. Langkah-langaknya adalah anda masuk ke browser anda lalu ketik localhost/nama file project anda.php. Berikut merupakan contohnya.



Hello world

Gambar 3.10 Hasil dari script echo

Jika anda memasukkan nama file anda dan anda kekurangan memasukkan .php anda akan menyebabkan seperti ini.

Not Found

The requested URL was not found on this server.

Apache/2.4.48 (Ubuntu) OpenSSL/1.1.1k PHP/8.0.8 Server at localhost Port 80

Gambar 3.11 Jika salah atau typo memasukan nama file pada pencarian browser

Konstanta dan Variabel dalam PHP

Arti dari variabel adalah berfungsi sebagai menampung data baik berupa angka dan juga huruf. Dan data ini juga bisa berubah ubah. Berbeda dengan konstanta memiliki fungsi yang sama dengan variabel tetapi data di dalam konstanta tidak dapat dirubah.

Berikut adalah contoh cara penulisan variabel dalam bahasa PHP.

Tabel 3.1 Aturan penulisan variabel dalam PHP

Benar	Salah
\$alamat	\$2alamat
\$alamat_rumah	\$alamat rumah
\$alamatswa	\$namakamu?
\$alamatrmhbudi	\$nama.anda

Berikut merupakan contoh program dari penjelasan variabel dan konstanta dalam bahasa PHP.

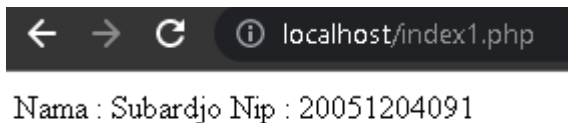
- Konstanta

A screenshot of a code editor with a dark background and light text. The code is as follows:

```
1 <?php
2 define ("Nama", "Subardjo");
3 define ("Nip", "20051204091");
4
5 // Nama Achmad akan menyebabkan eror pada program
6 echo "Nama : " . Nama;
7 echo " Nip : " .Nip;
8 ?>
```

Gambar 3.12 Script penggunaan konstanta

Berikut merupakan hasil dari script tersebut jika ditampilkan pada browser.



Gambar 3.13 Hasil script konstanta dalam PHP

- Variabel

A screenshot of a code editor window with a dark background and three colored window control buttons (red, yellow, green) at the top left. The code is written in a light green monospace font and is numbered from 1 to 7 on the left side. The code defines two variables, \$nama and \$nip, and then uses the echo function to output their values.

```
1 <?php
2 $nama = "Subardjo";
3 $nip = "20051204091";
4
5 echo "Nama : " . $nama . "<br>";
6 echo "Nip : " . $nip;
7 ?>
```

Gambar 3.14 Script penggunaan variabel

Jika anda ingin menampilkan pada browser maka hasilnya akan menjadi seperti ini.

Nama : Subardjo
Nip : 20051204091

Gambar 3.15 Hasil script penggunaan variabel

Tipe Data dalam Bahasa PHP

Tipe data merupakan hal yang sangat penting dalam pembuatan program. Tipe data merupakan sebuah interuksi agar si program dapat berjalan sesuai dengan interuksi dari

tipe data. Contohnya kita mempunyai 2 bilangan yaitu 4 dan 10 jika kita membagi kedua angka tersebut dengan tipe data integer akan menghasilkan nilai 2. Dan jika anda memasukkan tipe data float hasil niali akhirnya adalah 2,5.

Pada bahasa pemrograman php, jarang sekali developer/programmer menggunakan tipe data variabel. Karena bahasa PHP akan otomatis menginterpretir dengan sendirinya. Namun, dalam proses pengembangan sebuah program, developer menggunakan serta mendefinisikan tipe data variabel. Dalam bahasa PHP mensuport tipe data primitif. Berikut merupakan tipe data primitif pada bahasa PHP.

1. Integer
2. Float
3. Boolean
4. String
5. Object
6. Array
7. NULL

Berikut merupakan penjelasan dari tipe data pada diatas.

Tipe Data Integer

Pada Tipe data ini berfungsi sebagai tempat penyimpanan data berupa angka bilangan bulat. Meskipun kita membuat 2 variabel dengan isi data masing masing angka bilangan koma tetap hasilnya akan bernilai angka bulat.

Berikut merupakan contoh script tipe data integer pada bahasa pemrograman PHP. Disini kita memakai echo dan var dump sebagai menampilkan hasil tipe data.


```
1 <?php
2 # inisiasi dan inisialisasi variabel
3 $a = 20;
4 $b = 10;
5 $c = $a + 5;
6 $d = $b + (10 * 5);
7 $e = $b + $d;
8
9 # Tampilkan data dengan perintah echo
10 echo "Variabel a: {$a} <br>";
11 echo "Variabel b: {$b} <br>";
12 echo "Variabel c: {$c} <br>";
13 echo "Variabel d: {$d} <br>";
14 echo "Variabel e: {$e} <br>";
15
16 # mengetahui tipe data dari variabel
17 var_dump($e);
18 ?>
```

Gambar 3.16 Script menggunakan tipe data Integer

Berikut merupakan hasil dari script diatas jika kita tampilkan pada browser.

Variabel a: 20
Variabel b: 10
Variabel c: 25
Variabel d: 60
Variabel e: 70
int(70)

Gambar 3.17 Hasil script menggunakan tipe data integer

Tipe Data Float

Pada tipe data ini, hampir sama dengan tipe data integer. Integer menghasilkan angka bernilai bulat kalo tipe data float

digunakan untuk angka bernilai decimal berikut merupakan contoh scriptnya. Disini kita memakai echo dan var_dump untuk menampilkan data.



```
1 <?php
2 $basisdata = 11.5;
3 $pemrogramanwebsite = 15.5;
4 $sistemoperasi = 10.5;
5
6 # hitung nilai rata-rata
7 $Ratarata = ($basisdata + $pemrogramanwebsite + $sistemoperasi) / 3;
8
9 # Tampilkan data
10
11 echo "Nilai Basis Data: {$basisdata} <br>";
12 echo "Nilai Pemrograman Website: {$pemrogramanwebsite} <br>";
13 echo "Nilai Sistem Operasi: {$sistemoperasi} <br>";
14 echo "Rata-rata Nilai: {$Ratarata} <br>";
15
16 # Lihat tipe data dari variabel $rataRata
17 var_dump($Ratarata);
18 ?>
19
```

Gambar 3.18 Script PHP menggunakan tipe data float

Jika kita tampilkan pada browser maka hasilnya akan menjadi seperti ini.

```
Nilai Basis Data: 11.5  
Nilai Pemrograman Website: 15.5  
Nilai Sistem Operasi: 10.5  
Rata-rata Nilai: 12.5  
float(12.5)
```

Gambar 3.19 Hasil script menggunakan tipe data float

Tipe Data Boolean

Pada tipe data ini, terdapat 2 value atau nilai yaitu true dan false. Tipe data ini sangat simple, tetapi tipe data ini membutuhkan logika yang sangat kuat jika menggunakan metode ini. Berikut merupakan contoh script Boolean.



```
1 <?php
2
3 $apakahMahasiswaLulus = false;
4 $apakahMahasiswaSudahUjian = true;
5
6 var_dump($apakahMahasiswaLulus);
7 echo "<br>";
8 var_dump($apakahMahasiswaSudahUjian);
9 ?>
```

Gambar 3.20 Script menggunakan tipe data boolean

Berikut merupakan hasil dari script diatas jika anda tampilkan pada browser.

```
bool(false)
bool(true)
```

Gambar 3.21 Hasil script menggunakan tipe data boolean

Tipe Data String

Tipe data ini merupakan sebuah penyimpanan data berbentuk teks atau kata. Tipe data string ditandai dengan tanda “ atau tanda ”. Berikut merupakan contoh dari script tipe data string.

```
1 <?php
2
3 $namaDepan = "Achmad"; # pakai tanda petik dua
4 $namaBelakang = 'Subardjo'; # pakai tanda petik satu
5
6 # menggabungkan dua variabel dengan tanda
7 # petik dua
8 $namaLengkap = "{$namaDepan} {$namaBelakang}";
9
10 # anda juga bisa menggabungkan string dengan menggunakan tanda titik (.)
11 $namaLengkap2 = $namaDepan . ' ' . $namaBelakang;
12
13 # [Tampilkan Data]
14 # kita bisa memasukkan variabel lain jika menggunakan tanda petik dua
15 echo "Nama Depan: {$namaDepan} <br>";
16 # ada pun jika pakai tanda petik satu, kita tidak bisa memasukkan variabel
17 # di dalam string akan tetapi menggabungkannya dengan operator titik (.)
18 echo 'Nama Belakang: ' . $namaBelakang . ' <br>';
19
20 echo "Nama lengkap : {$namaLengkap}";
```

Gambar 3.22 Script menggunakan tipe data string

Jika kita tampilkan pada browser maka akan menjadi seperti ini.

```
Nama Depan: Achmad
Nama Belakang: Subardjo
Nama lengkap : Achmad Subardjo
```

Gambar 3.23 Hasil script menggunakan tipe data string

Tipe Data Object

Pada tipe data ini berfungsi tidak hanya menyimpan data saja tetapi juga bisa memberi informasi bagaimana megelolah data tersebut. Pada tipe data ini berisikan tipe tipe data lain

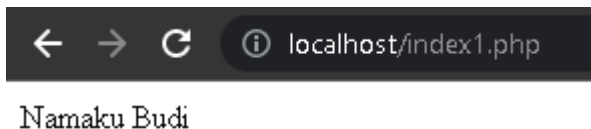
contohnya float, integer, string, dll. Berikut merupakan contoh tipe data object.

A screenshot of a code editor with a dark background and light-colored text. The code is PHP and defines a class named 'Nama' with a property '\$str'. It then creates an instance of the class, sets the property to 'Namaku Budi', and echoes the value. The code is as follows:

```
1 <?php
2 class Nama
3 {
4     var $str;
5 }
6
7 // membuat object pesan. ini disebut instansiasi
8 $Diri = new Nama();
9
10 //memberikan nilai utk properti str pada objek pesan
11 $Diri->str="Namaku Budi";
12
13 // mencetak nilainya (pengolahan data)
14 echo $Diri->str;
15 ?>
```

Gambar 3.24 Script menggunakan tipe data object

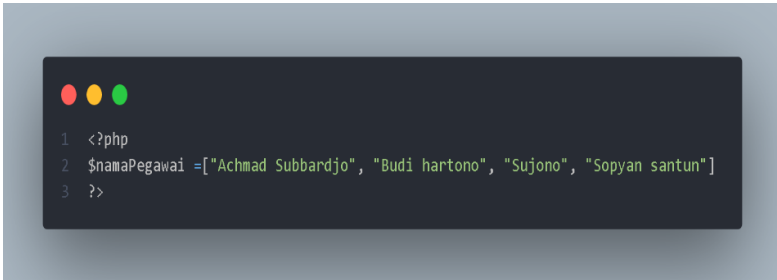
Jika kita tampilkan pada browser maka hasilnya akan menjadi seperti ini.



Gambar 3.25 Hasil script menggunakan tipe data object

Tipe Data Array

Pada tipe data ini berfungsi sebagai menyimpan kumpulan data. Pada tipe data array diawali dengan tanda [dan diakhiri dengan tanda]. Sekarang kita akan buat kumpulan nama pegawai. Berikut merupakan contoh tipe data array.



```
1 <?php
2 $namaPegawai =["Achmad Subbardjo", "Budi hartono", "Sujono", "Sopyan santun"]
3 ?>
```

Gambar 3.26 Script menggunakan tipe data array

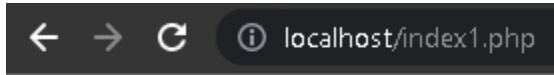
Disini kita akan memanggil nama budi hartono. Dalam array untuk menghitung sebuah data dimulai dari angka 0. Berikut merupakan contohnya.



```
1 <?php
2 $namaPegawai =["Achmad Subbardjo", "Budi hartono", "Sujono", "Sopyan santun"];
3 echo $namaPegawai[1]; // Budi hartono
4 ?>
```

Gambar 3.27 Script php tipe data array pemanggilan data

Berikut merupakan hasil jika kita tampilkan pada browser.




Budi hartono

Gambar 3.28 Hasil dari script menggunakan tipe data array

Tipe Data NULL

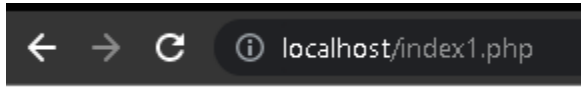
Pada tipe data ini berfungsi untuk menunjukkan bahwa variabel tidak memiliki sebuah data ataupun nilai. Tipe data null pada bahasa PHP hampir mirip dengan tipe data null pada bahasa python. Berikut merupakan contoh scriptnya.

A screenshot of a code editor with a dark background and light text. The code is PHP and demonstrates how to compare a variable with 0. It includes comments in Indonesian. The code is as follows:

```
1 <?php
2 // x pertama kali diberi nilai 1
3 $x = 1;
4 if($x >0){
5     echo "Bisa dibandingkan karena tidak null";
6 }
7
8 // x diberi NULL
9 $x = null;
10 if($x > 0){
11     echo "Tidak bisa dibandingkan karena null";
12 }
13 ?>
```

Gambar 3.29 Script menggunakan tipe data null

Jika kita tampilkan pada browser maka hasilnya akan menjadi seperti ini.



Bisa dibandingkan karena tidak null

Gambar 3.30 Hasil script tipe data null

Kondisi

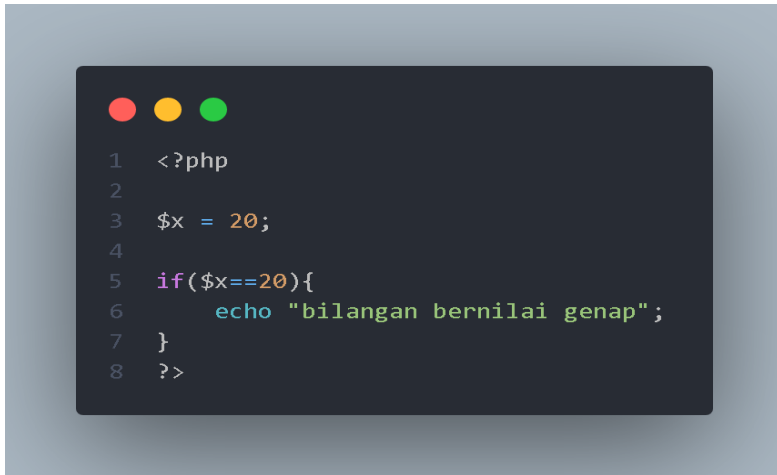
Pada bahasa PHP kondisi merupakan sebuah pilihan, dalam php mempunyai 4 macam kondisi yaitu

- If
- If ... else if ... else
- If ... else
- Switch-case

Sekarang kita akan membahas satu persatu dari 4 macam kondisi.

If

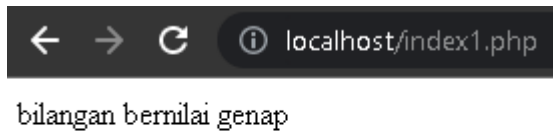
Kondisi if berfungsi sebagai menyatakan satu pilihan kondisi. Berikut merupakan contohnya.

A screenshot of a terminal window with a dark background and light text. The code is as follows:

```
1 <?php
2
3 $x = 20;
4
5 if($x==20){
6     echo "bilangan bernilai genap";
7 }
8 ?>
```

Gambar 3.31 Script php kondisi if

Jika kita tampilkan pada browser maka hasilnya akan menjadi seperti ini.



Gambar 3.32 Hasil script php kondisi if

If.. else if.... else....

Pada kondisi ini dimana user dapat memilih beberapa pilihan. Berikut merupakan contoh scriptnya.

```
1 <?php
2
3 $x = 20;
4
5 if($x==10){
6     echo "bilangan bernilai genap";
7 }
8 else if($x==11){
9     echo "bilangan bernilai ganjil";
10 }
11 else{
12     echo "bilangan tidak ada dalam pilihan";
13 }
14 ?>
```

Gambar 3.33 Script php kondisi if else if else

Pada script diatas jika kita tampilkan pada browser maka akan menjadi seperti ini.



bilangan tidak ada dalam pilihan

Gambar 3.34 Hasil script php kondisi if elseif else

If.. else...

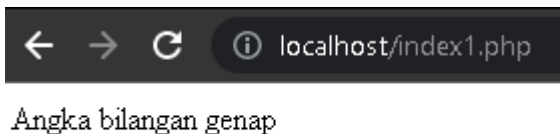
Pada kondisi ini menyatakan 2 pilihan. Pada program sederhana biasanya sering dipakai kondisi ini. Berikut merupakan contoh kondisi if else pada PHP.

A screenshot of a code editor with a dark background and light text. The code is a PHP script that checks if a variable \$x is equal to 35. If true, it echoes "Angka bilangan ganjil"; otherwise, it echoes "Angka bilangan genap". The code is numbered from 1 to 10.

```
1 <?php
2
3 $x = 20;
4
5 if($x==35){
6 echo "Angka bilangan ganjil";
7 }else{
8     echo "Angka bilangan genap";
9 }
10 ?>
```

Gambar 3.35 Script php kondisi if else

Jika kita tampilkan pada browser maka hasilnya akan menjadi seperti ini.



Gambar 3.36 Hasil script kondisi if else

Swith-Case

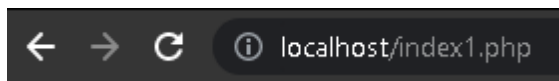
Switch-case merupakan sebuah kondisi dimana berfungsi sebagai mengecek kebenaran suatu data atau nilai dan memiliki banyak pilihan. Berikut merupakan contoh script switch-case Bahasa pemrograman PHP.



```
1 <?php
2 $angka = 4;
3
4 switch ($angka) {
5     case 1:
6         echo " data angka adalah satu";
7         break;
8
9     case 2:
10        echo "data angka adalah dua";
11        break;
12
13    case 3:
14        echo "data angka adalah tiga";
15        break;
16
17    case 4:
18        echo "data angka adalah empat";
19        break;
20
21    case 5:
22        echo "data angka adalah lima";
23        break;
24
25    case 6:
26        echo "Isi variabel angka adalah enam";
27        break;
28
29    default:
30        echo "Isi variabel tidak di temukan";
31        break;
32    }
33 ?>
```

Gambar 3.37 Script menggunakan kondisi switch-case

Jika kita tampilkan pada browser maka hasilnya akan menjadi seperti ini



data angka adalah empat

Gambar 3.38 Hasil script menggunakan kondisi switch case

Fungsi dalam PHP

Dalam bahasa pemrograman PHP, memiliki fungsi lebih dari 1000 dari bawaan bahasa php. Selain itu, anda bisa membuat fungsi sendiri dalam script php.

Fungsi merupakan sebuah pernyataan yang dapat bisa dapat anda gunakan beberapa kali dalam script anda. Sebuah fungsi tidak secara otomatis akan dieksekusi dalam sebuah halaman website. Tetapi sebuah fungsi akan dieksekusi ketika fungsi tersebut dipanggil.

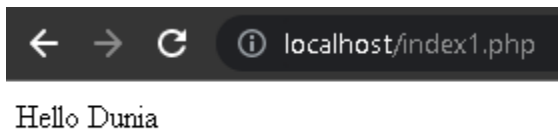
Sekarang kita akan membuat fungsi dalam bahasa php. Kita akan menuliskan fungsi `myFunction()` dalam script php dan diawali dengan `{` dan diakhir dengan `}`. Berikut merupakan contohnya.



```
1 <?php
2     function myFunction() {
3         echo "Hello Dunia";
4     }
5
6     myFunction(); // panggil fungsinya
7 ?>
```

Gambar 3.39 Script fungsi PHP

Jika kita tampilkan pada browser maka hasilnya akan menjadi seperti ini.



Gambar 3.40 Hasil script fungsi PHP

Argumen pada Fungsi PHP

Pada fungsi PHP ini, data/nilai pada php akan dilanjutkan ke dalam fungsi melalui argument. Argument dapat anda tentukan setelah nama fungsi. Anda bisa memasukkan argument sebanyak banyaknya dalam fungsi tersebut dengan cara menambahkan tanda koma setiap argument.

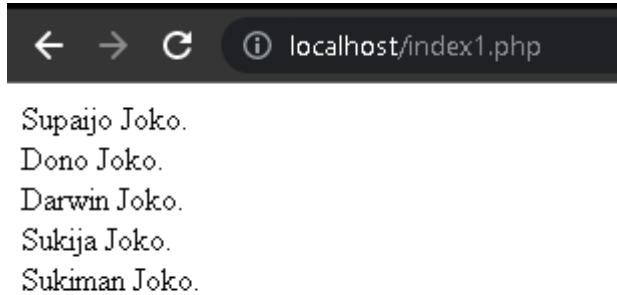
Berikut merupakan contoh script dari argument pada fungsi php. Disini kita akan memakai dengan sebuah argument yaitu \$fmahasiswa. Setelah itu kita membuat sebuah fungsi yaitu teknikinformatikaName(). Disini anda menambahkan nama dalam fungsi.



```
1 <?php
2     function teknikinformatikaName($fmahasiswa) {
3         echo "$fmahasiswa Joko.<br>";
4     }
5
6     teknikinformatikaName("Supaijo");
7     teknikinformatikaName("Dono");
8     teknikinformatikaName("Darwin");
9     teknikinformatikaName("Sukija");
10    teknikinformatikaName("Sukiman");
11    ?>
```

Gambar 3.41 Script php menggunakan argumen

Berikut merupakan hasil dari script di atas jika kita tampilkan pada browser.



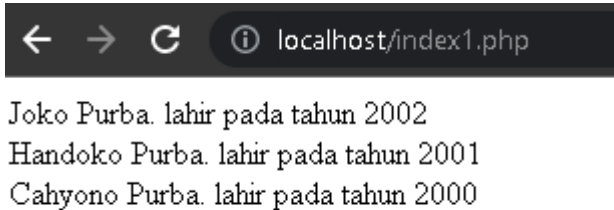
Gambar 3.42 Hasil script php menggunakan argumen

Sekarang kita akan menggunakan 2 argumen yaitu \$fname dan \$fyear. Berikut merupakan contoh scriptnya.



Gambar 3.43 Script PHP menggunakan 2 argumen

Berikut merupakan hasil script jika kita ditampilkan pada browser.



Gambar 3.44 Hasil script PHP menggunakan 2 argumen

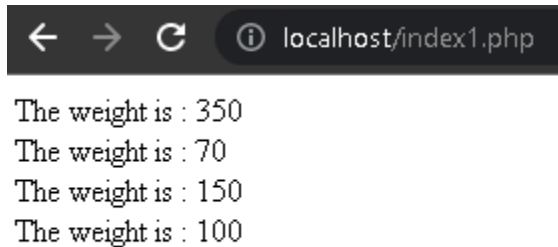
Data Default Argument pada PHP

Berikut merupakan contoh penggunaan default argument pada bahasa PHP. Disini kita akan memanggil sebuah fungsi yaitu `setWeight()` dan tanpa menggunakan sebuah argument.



Gambar 3.45 Script PHP menggunakan data default

Jika kita tampilkan pada browser maka hasilnya akan menjadi seperti ini.



Gambar 3.46 Hasil script PHP menggunakan data default

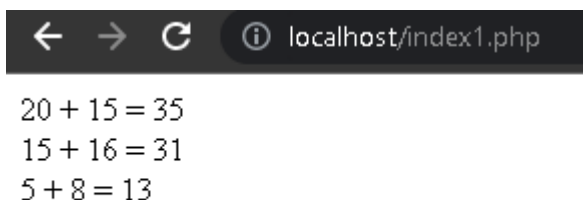
Fungsi PHP mengembalikan nilai/return

Dalam bahasa pemrograman PHP, ada suatu fungsi untuk mengembalikan nilai. Untuk fungsi pengembalian nilai pada bahasa php kita menggunakan fungsi return. Berikut merupakan contoh script pengembalian nilai.

```
1 <?php declare(strict_types=1); // strict requirement
2     function sum(int $a, int $b) {
3         $c = $a + $b;
4         return $c;
5     }
6
7     echo "20 + 15 = " . sum(20, 15) . "<br>";
8     echo "15 + 16 = " . sum(15, 16) . "<br>";
9     echo "5 + 8 = " . sum(5, 8);
10 ?>
```

Gambar 3.47 Script PHP menggunakan return

Jika kita tampilkan pada browser maka hasilnya akan menjadi seperti ini.



Gambar 3.48 Hasil script PHP menggunakan return

Looping/Perulangan pada PHP

Pada sub bab ini kita akan mempelajari tentang looping pada bahasa pemrograman PHP. Looping artinya perulangan jadi biasanya kita menulis kode yang sama secara manual dan akan memakan waktu. Tetapi adanya looping ini kode untuk perulangan tersebut akan menjadi ringkas dan anda tidak

perlu memakan banyak waktu untuk menulis scriptnya. Berikut merupakan macam macam looping pada bahasa pemrograman PHP.

Tabel 3.2 Definisi dari macam macam perulangan

Macam macam looping	Definisi
while	While merupakan sebuah looping melalui sebuah barisan kode, selama kondisi yang anda tentukan benar.
Do while	Do while merupakan sebuah looping melalui barisan kode minimal sekali, selama kondisi yang anda tentukan benar.
For	For merupakan sebuah looping dengan melalui blok kode dalam beberapa kali.
Foreach	Foreach merupakan sebuah looping dengan dengan melalui blok kode dengan mneggunakan tipe data array.

Setelah kita mempelajari dari definisi dari masing masing looping. Sekarang kita akan menerapkan looping pada php melalui script. Berikut merupakan macam macam contoh looping pada PHP.

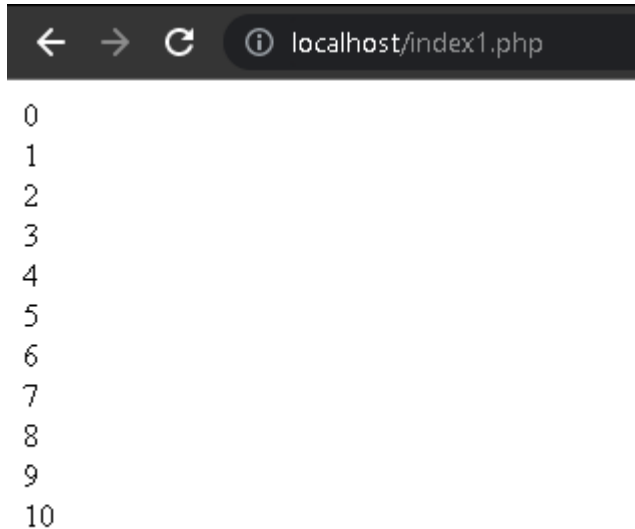
While

A screenshot of a code editor window with a dark background and light text. The code is a PHP script that initializes a variable \$a to 0 and then enters a while loop that prints the value of \$a and increments it by 1 until it reaches 10. The code is as follows:

```
1 <?php
2     $a=0;
3
4     while($a <= 10){
5         echo $a."<br>";
6         $a++;
7     }
8 ?>
```

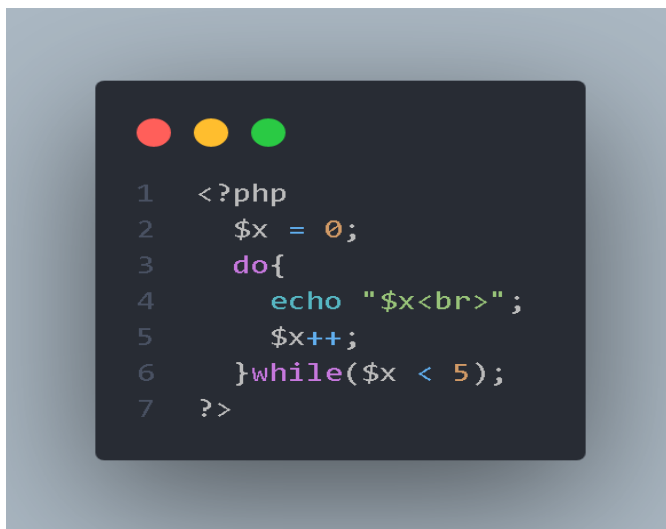
Gambar 3.49 Script PHP menggunakan while

Pada script diatas kita menggunakan variabel a yaitu 0. Setelah itu kita membuat perulangan menggunakan while disini kita menggunakan tanda `<=` artinya kurang dari sama dengan. Berbeda dengan tanda `<` kalo anda memasukkan tanda `<` hasil akhirnya akan kurang dari 10. Disini kita juga memakai `$a++` yaitu logika `a=a+1` berbeda dengan `++a` artinya adalah `a=1+a`. Berikut jika kita tampilkan pada browser maka hasilnya akan menjadi seperti ini.



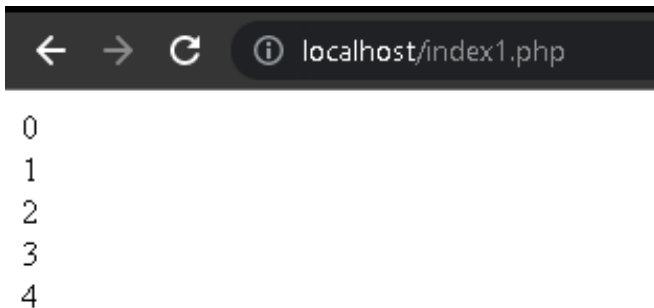
Gambar 3.50 Hasil script php menggunakan while

Do-while



Gambar 3.51 Script PHP menggunakan do while

Pada script perulangan do-while kita menggunakan variabel x dengan nilai 0 pada do, kita memasukkan echo sebagai menampilkan data dan
 sebagai enter. Setelah itu kalian membuat post increment pada variabel x. Setelah mengisi do kita akan mengisi while yaitu sebuah kondisi untuk variabel x dan kita akan menentukan batasan looping. Berikut merupakan hasil script do while jika kita tampilkan pada browser.



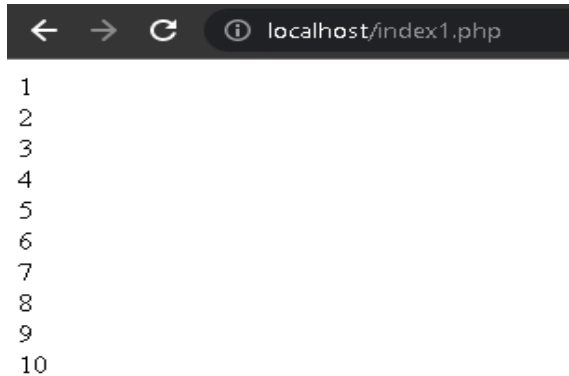
Gambar 3.52 hasil script php menggunakan do while

For



Gambar 3.53 Script PHP menggunakan for

Pada Script perulangan for kita bisa mendeklarasikan sebuah variabel ke dalam perulangan dan langsung mengeksekusinya. Berikut hasil dari script perulangan for.



```
localhost/index1.php  
1  
2  
3  
4  
5  
6  
7  
8  
9  
10
```

Gambar 3.54 Hasil script php menggunakan do while

Foreach

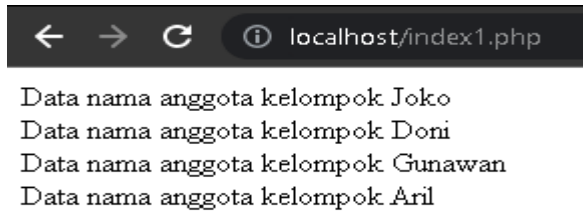


```
1 <?php  
2 $array = array("Joko", "Doni", "Gunawan", "Ari1");  
3  
4 foreach($array as $nama){  
5     echo "Data nama anggota kelompok $nama <br />";  
6 }  
7 ?>
```

Gambar 3.55 Script php menggunakan foreach

Pada looping menggunakan foreach, kita menggunakan tipe data array sebagai isi dari perulangan tersebut. Berikut hasil

dari script foreach jika kita tampilkan pada browser maka hasilnya akan menjadi seperti ini.



Gambar 3.56 Hasil script PHP menggunakan foreach

Membuat Game pada Bahasa Pemrograman PHP

Setelah kita mempelajari dasar dasar pada bahasa pemrograman PHP sekarang kita akan melakukan pembuatan game pada PHP disini kita membutuhkan sebuah file php dan kita menggunakan internal css supaya menghemat file. Disini kita akan membuat sebuah game sederhana yaitu game suit dan musuh kita melawan computer berikut merupakan langkah langkah pembuatan game.

1. Kita membuat struktur html terlebih dahulu. Disini kita menggunakan html 5. Berikut merupakan contohnya.

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>GAME SUIT</title>
8 </head>
9 <body>
10
11 </body>
12 </html>
```

Gambar 3.57 Struktur html untuk pembuatan game PHP

2. Setelah itu, kalian membuat struktur untuk membuat pikiran dari komputer melalui bahasa pemrograman PHP berikut merupakan contohnya.

```
1 <!------- LOGIC KOMPUTER ----->
2 <?php
3   $com = rand();
4   if ($com > 100000000 && $com < 800000000) {
5     $com = 'GAJAH';
6   }elseif ($com > 800000000 && $com < 1250000000){
7     $com = 'SEMUT';
8   }else {
9     $com = 'ORANG';
10  }
11
12  ?>
```

Gambar 3.58 Script membuat logika komputer

3. Setelah itu tahap selanjutnya adalah membuat sebuah input untuk pemain antara user dan computer berikut merupakan contohnya.

A screenshot of a code editor with a dark background and light-colored text. The code is HTML and is numbered from 1 to 9 on the left side. The code defines a form with a title, a paragraph of instructions, a text input field, and a submit button. The input field has a JavaScript event listener that converts the input to uppercase. The submit button has a unique name.

```
1 <!------- INPUTAN PEMAIAN ----->
2 <form action="" method="POST">
3 <h1>GAMES SUIT</h1>
4 <p>MASUKAN : GAJAH / SEMUT / ORANG ( Catatan : Jangan Ada Sepasi )</p>
5 <input onkeyup="this.value = this.value.toUpperCase()" type="text"
6 name="nama" placeholder="Masukan Sesuatu. . ." required autofocus>
7 <button type="submit" name="sumbit"> SUUUIIITTT ! </button>
8 </form>
9 <br>
```

Gambar 3.59 Membuat input pada user

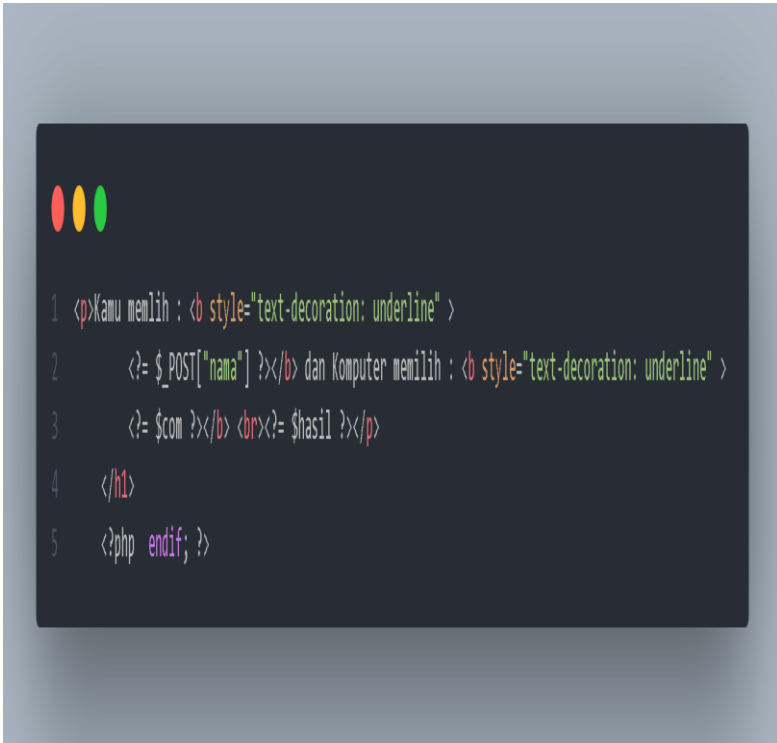
4. Tahap selanjutnya adalah kita membuat logika pada game tersebut menggunakan bahasa pemrograman PHP, berikut merupakan contohnya.

A screenshot of a code editor with a dark background and light-colored text. The code is PHP logic for a game, enclosed in HTML tags. It checks for a submit button and then uses a series of if-elseif-else statements to determine the outcome based on the player's name and the computer's move. The outcomes are 'HASILNYA SERI !', 'KAMU MENANG !', or 'KAMU KALAH !'. There is also a catch-all else statement for invalid input.

```
1 <!------- LOGIC GAMES ----->
2
3 <?php if ( isset($_POST["submit"]) ) : ?>
4   <h1><?php
5     $hasil = '';
6     if ( $_POST["nama"] == $com){
7       $hasil = 'HASILNYA SERI !';
8     }elseif ( $_POST["nama"] == 'GAJAH'){
9       if ( $com == 'ORANG'){
10        $hasil ='KAMU MENANG !';
11      }else{
12        $hasil ='KAMU KALAH !';
13      }
14    }elseif ( $_POST["nama"] == 'ORANG'){
15      if ( $com == 'SEPUT'){
16        $hasil ='KAMU MENANG !';
17      }else{
18        $hasil ='KAMU KALAH !';
19      }
20    }elseif ( $_POST["nama"] == 'SEPUT'){
21      if ( $com == 'GAJAH'){
22        $hasil ='KAMU MENANG !';
23      }else{
24        $hasil ='KAMU KALAH !';
25      }
26    }else {
27      $hasil = 'YANG ANDA MASUKAN SALAH ATAU ADA SEPASI !';
28    }
29  ?>
```

Gambar 3.60 Membuat logika game

5. Tahap selanjutnya adalah kalian membuat sebuah pernyataan untuk mengetahui siapa yang menang dan siapa yang kalah. Berikut merupakan contohnya.

A screenshot of a code editor with a dark background and three colored window control buttons (red, yellow, green) in the top left corner. The code is written in a light-colored font and is numbered 1 through 5 on the left side. The code uses PHP to output HTML for a game result, including bold text and underlined text.

```
1 <p>Kamu memilih : <b style="text-decoration: underline" >  
2     <?=$_POST["nama"] ?></b> dan Komputer memilih : <b style="text-decoration: underline" >  
3     <?=$_com ?></b> <br><?=$_hasil ?></p>  
4 </h1>  
5 <?php endif; ?>
```

Gambar 3.61 Membuat hasil menang atau kalah terhadap user dan komputer

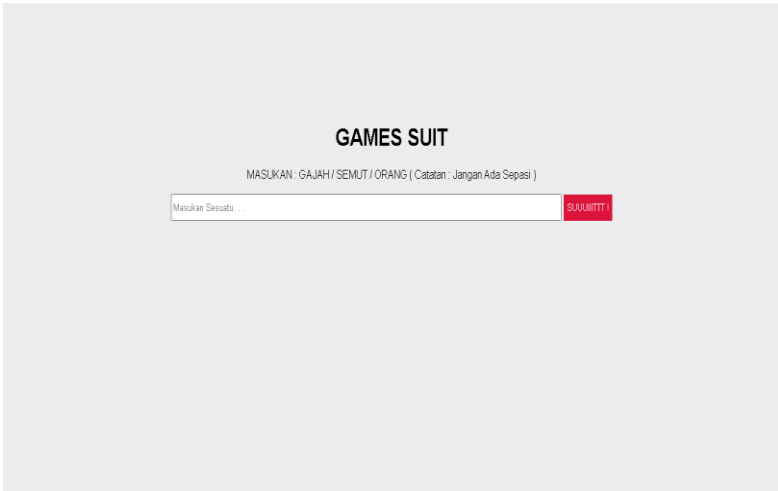
6. Setelah itu kalian membuat style css untuk mempercantik halaman game anda. Berikut merupakan contohnya.

A screenshot of a code editor with a dark background and light text. The code is CSS styling for a game page. It includes a font import from Google Fonts, a body style with padding, background color, and font family. It also defines styles for h1, p, form, input, and button elements, including a hover effect for the button. The code is numbered from 1 to 30.

```
1 <style>
2   @import url('https://fonts.googleapis.com/css2?family=Nunito+Sans:wght@300;400;600&display=swap');
3   body {
4     padding-top: 10%;
5     background-color: #ececfc;
6     font-family: 'Nunito+Sans', sans-serif;
7   }
8   h1, p, form {
9     text-align: center;
10  }
11
12  input{
13    width: 50%;
14    height: 30px;
15  }
16
17  button {
18    height: 35px;
19    background-color: crimson;
20    color: white;
21    border: none;
22    cursor: pointer;
23    margin: auto;
24  }
25
26  button:hover{
27    background-color: #688F88;
28  }
29
30 </style>
```

Gambar 3.62 Styling game pada PHP

7. Setelah itu kalian masuk ke halaman game anda.



Gambar 3.63 Halaman game suit PHP

8. Setelah kita melakukan kalian coba game anda. Berikut merupakan contohnya.



Gambar 3.64 Contoh hasil antara user dan komputer



DAFTAR PUSTAKA

- Enterprise Jubilee.2015.Kumpulan Aplikasi PHP untuk Pemula.Jakarta : PT Elex Media Komputindo.
- Gregory Jason.2017.*Game Engine Architecture*. CRC Press.
- Gregory Jason.2018.*Game Engine Architecture, Third Edition*. CRC Press.
- Haverbeke Marijin.2018.*Eloquent Javascript, 3rd Edition : a modern introduction to programming*. No starch Press.
- Lengstorf Jason.2009. *PHP for Absolute Beginners*. Apress.
- McNave Terry.2011. *JavaScript for Absolute Beginners*. Apress.
- Rex van der Spuy.2013.*Foundation Game Design with HTML5 and JavaScript*.Apress.
- Wiley John & Sons.2005. *Professional Javascript for Web Developer*. Indianapolis : Wiley Publishing Inc.
- Williams Lamar James.2012.Learning HTML5 Game Programming. Addison-Wesley.



GLOSARIUM

A

Argumen = merupakan sebuah fungsi inputan jika fungsi tersebut dipanggil.

Assignment = merupakan sebuah cara untuk memberi nilai pada sebuah variabel dengan memiliki 2 cara yaitu mengcopy nilai atau data dari data variabel lain.

Array = merupakan sebuah tip data untuk menampung banyak data. Data tersebut bisa bernilai angka maupun kata.

B

Boolean = merupakan sebuah tipe data yang memiliki 2 nilai yaitu true dan false.

Browser = merupakan sebuah tempat pencarian dan juga biasa menjadi tempat untuk melihat hasil dari kodingan.

C

Client side programming = merupakan scripting yang bertuju kepada tampilan website.

Const = Variabel yang bernilai konstanta

D

Data = merupakan sebuah kumpulan fakta yang menggambarkan suatu keadaan. Dalam pemrograman data adalah bisa merupakan angka dan kata.

DOM (*Document Object Model*) = merupakan sebuah cara pada bahasa pemrograman javascript untuk melihat document document pada HTML.

Do-while = merupakan salah satu perulangan hasil dari modifikasi dari perulangan while. Perulangan pada do-while adalah pemindahan posisi pemeriksaan suatu kondisi ke akhir perulangan.

E

Embedded script = Merupakan sebuah sisipan bahasa pemrograman ke dalam dokumen html.

External script = merupakan penulisan bahasa pemrograman di luar file html tetapi file html dan file bahasa pemrograman tersebut tersambungkan oleh syntax.

F

Function = merupakan sebuah kode dalam program untuk menyelesaikan tugas tugas tertentu.

For = merupakan salah satu jenis perulangan atau looping yang dapat mengerjakan sebuah pernyataan berulang-ulang.

Foreach = merupakan salah satu jenis perulangan atau looping yang di khususkan untuk bertipe data array.

A

Argumen, viii, 112, 131
Array, 15, 18, 98, 104, 131
Assignment, vii, 20, 131

B

Browser, vii, 44, 131

C

Client side programming, 131
Const, 131

D

Data, vii, viii, 15, 98, 100, 101, 102, 103, 104, 105, 114, 131
DOM, vii, xi, 52, 53, 54, 55, 58, 59, 62, 63, 132
Do-while, 118, 132

E

Embedded script, 132
External script, 132

F

For, 116, 119, 132
Foreach, 116, 120, 132
Function, vii, 27, 28, 132



BIOGRAFI PENULIS



Dodik Arwin Dermawan, S.ST., S.T., M.T., lahir di Tanggul – Jember 8 Januari 1978. Pendidikan Sekolah Dasar sampai dengan Sekolah Menengah Atas di Kota Pahlawan Surabaya. Pendidikan di perguruan tinggi ditempuh di Diploma 3 Politeknik Negeri Surabaya dengan konsentrasi Bidang Telekomunikasi. Selanjutnya meneruskan pendidikan di S1 Teknik Elektro ITS dengan konsentrasi Bidang Telekomunikasi, dan D4 Politeknik Negeri Surabaya dengan konsentrasi Bidang Komputer. Setelah lulus S1, penulis mendapatkan beasiswa unggulan dari pemerintah untuk melanjutkan kuliah di S2 ITS Jaringan Cerdas Multimedia konsentrasi Bidang Game Tech. Pengalaman kerja penulis diawali di Kota Pahlawan Surabaya, yakni di PT. Ketabang Kali Surabaya sebagai Kepala Quality Control. Penulis juga pernah bekerja sebagai Technical Support VSAT PT. Pasifik Satalit Nusantara, serta terlibat di pengembangan sistem informasi di beberapa instansi diantaranya ; Dinas Perindustrian dan Perdagangan Provinsi Jawa Timur, Dinas Lingkungan Hidup Provinsi Jawa Timur, PT. Axo Green Laboratory, PT. Mitra Hijau Indonesia, dan Kemkominfo. Pengalaman kerja di bidang pendidikan juga pernah ditempuh, khususnya mengajar di rumpun Teknologi Informasi dan Komunikasi, diantaranya ; Politeknik Madiun, Universitas Islam Negeri Maulana Malik Ibrahim Malang, Politeknik NSC Surabaya, Universitas Hasyim Asy’ari (UNHASY) Jombang, Prisma Profesional Surabaya, dan Universitas Negeri Surabaya sampai dengan sekarang. Untuk di Universitas Negeri

Surabaya, penulis pernah menjabat sebagai Kepala Divisi Pengembangan Sistem Informasi PPTI dan Kepala Program Studi D4 Manajemen Informatika. Penulis juga aktif dalam penulisan artikel penelitian dan beberapa pengembangan sistem informasi telah di daftarkan di HKI.



Chamdan Mashuri, S.Kom., M.Kom., Lahir di Jombang pada tanggal 4 Juni 1988. Sejak kecil tinggal dan besar di kota santri, kota Jombang, Jawa Timur. Sekolah Dasar hingga Sekolah Menengah Atas berada di lingkungan Pondok Pesantren Darul Ulum, kemudian pendidikan

Sarjana dari tahun 2007 hingga 2011 di Universitas Trunojoyo Madura (UTM) dengan mengambil program studi Teknik Informatika. Setelah lulus S1 kemudian bekerja selama 2 tahun sebagai IT dan GA Supervisor di PT. Indospring Tbk Gresik, Jawa Timur. Kemudian pada tahun 2013 memutuskan untuk mengundurkan diri dan memulai usaha sendiri serta bekerja di bidang pendidikan yaitu sebagai Dosen di Universitas Hasyim Asy'ari (UNHASY) Tebuireng Jombang di Fakultas Teknologi Informasi dari tahun 2014 sampai sekarang. dengan mengajar beberapa mata pelajaran termasuk Algoritma Pemrograman, dan desain SI, Manajemen Proyek, Perencanaan Bisnis dan Perilaku Organisasi. Tahun 2017 telah lulus Magister Sistem Informasi (MSI) Universitas Diponegoro (Undip) Semarang, Jawa Tengah, mulai tahun 2015 dipercaya menjadi pembina organisasi kemahasiswaan Fakultas Teknologi Informasi dan tahun 2018 - sekarang menjadi Ketua Program Studi D3 Manajemen Informatika. Aktif menulis dan meneliti sejak 2018 hingga saat ini telah menghasilkan beberapa buku ber-ISBN dan HKI serta menerbitkan 6 artikel internasional terindeks Scopus, dan

beberapa artikel di jurnal nasional terakreditasi. Selain itu juga Sebagai Reviewer Jurnal Nasional Terakreditasi Sinta 2 yaitu Jurnal JSINBIS UNDIP dan Reviewer pada "International Symposium of Eart, Energy, Environmental Science and Sustainable Development" (International Symposium of JESSD) College of Environmental Sciences Universitas Indonesia (UI).



Ginanjar Setyo Permadi, S.Kom., M.Kom.,

Lulus S1 Program Studi Sistem Informasi Universitas Amikom tahun 2015 dan Lulus S2 program Magister Sistem Informasi Universitas Diponegoro 2017, Lahir di Ponorogo 14 April 1993, Saat ini penulis merupakan dosen salah satu Universitas di Jawa Timur, yakni dosen Fakultas Teknologi Informasi, Universitas Hasyim Asy'ari Tebuireng Jombang dan juga sebagai Ketua Unit PMB dan Wisuda, penulis juga mengajar sebagai Tutor di Universitas Terbuka, aktif menulis jurnal nasional maupun international salah satu judulnya "Sistem Evaluasi Bahan Pembelajaran Menggunakan Metode DEMATEL dan ANP" dimuat di JSINBIS, pernah menulis buku dengan judul Buku Ajar Datawarehouse. Penulis dapat dihubungi melalui email ginanjar.s.permadi@gmail.com



Duta Alif Gunawan, Lahir di Surabaya pada tanggal 26 Mei 2002. Sejak kecil tinggal di kota Surabaya Jawa Timur. Pendidikan SDN Sawunggaling 1 Surabaya pada tahun 2008 sampai 2014, SMP Negeri 16 Surabaya pada tahun 2014 sampai 2017, dan SMA WACHID HASYIM 2 TAMAN pada tahun 2017 sampai

2020. Pada tahun 2020 menempuh pendidikan kuliah di UNIVERSITAS NEGERI SURABAYA dengan mengambil studi teknik informatika. Pada tahun 2021, saya membuat buku berjudul Pemrograman Website Dasar. Penulis dapat dihubungi melalui email : dutaalif.20091@mhs.unesa.ac.id



Dini Widiasih S.T., lahir di Semarang 22 Juli 1978. Pendidikan Sekolah Dasar ditempuh di Semarang. Pendidikan Sekolah Menengah Pertama dan Menengah Atas ditempuh di kota Jambi. Pendidikan di perguruan tinggi yang pernah ditempuh adalah Diploma 3 Politeknik Negeri Sriwijaya, Palembang dengan konsentrasi Bidang Telekomunikasi. Selanjutnya meneruskan pendidikan di S1 Teknik Elektro ITS dengan konsentrasi Bidang Telekomunikasi. Saat ini sedang mengajar di Sekolah Menengah Atas Wachid Hasyim 2 Taman, Sidoarjo yang berada di bawah naungan Yayasan Pendidikan & Sosial Ma'arif dengan mata pelajaran yang diampu adalah Informatika.