|  |  |
| --- | --- |
| **Materi** | **Nilai** |
| Materi PHP Untuk Siswa SMK V 1-9 | 95 |

**Saya Sudah Belajar dan Mengerti dan Saya Bisa**

1. Saya sudah bisa menginstall Xampp sebagai web service untuk melihat output coding PHP, yang dimana PHP itu adalah server side programming artinya Program ini harus diletakkan di dalam web server (htdocs). Dan pastikan start apache

2. Saya bisa menambahkan code dari user snippet html.json ke php.json untuk mempercepat coding

3. Untuk menampilkan tulisan menggunakan echo “Tulisan”;

4. Untuk dapat memberi jarak antar tulisan atau enter ke bawah menggunakan echo ‘<br>’;

5. Semua program PHP harus masuk ke dalam antara tag open dan close PHP, yang diluar itu sudah bukan PHP

6. Jika didalam elemen HTML terdapat 2 tanda petik dua (“ ”) maka echo yang ada pada PHP diawali dengan tanda petik 1, begitu juga sebaliknya

7. Untuk menyambung dapat menggunakan titik (.) baik itu angka, kalimat, kata, dll.

8. Jika ingin menampilkan angka langsug saja echo 2019; (tidak perlu tanda petik), jika diberi tanda petik maka akan dianggap sebagai string

9. Jika kita ingin menampilkan string dan angka tidak bisa disambung menggunakan titik, jadi dimasukkan ke dalam string saja

10. echo 2019; merupakan angka karena itu bisa langsung diberi operasi matematika, misal echo 2019 + 2; maka hasilnya menjadi 2021

11. PHP tidak seperti bahasa pemrograman lain yang harus mendeklarasikan variabel nya, di PHP variabelnya adalah $

12. Tempat variabel yaitu Kiri = Wadah / Variabel. Kanan = Isi nya

13. Untuk menampilkan isinya kita hanya perlu memanggil variabelnya seperti echo $nama;

14. Di dalam pemanggilan variabel kita bisa menggabungkan/menyambung string dengan variabel dengan titik(.)

15. Untuk melihat tipe data kita menggunakan var\_dump($nama); . bisa nanti hasilnya string, int, float, dll.

16. Tipe data dalam PHP akan mengikut nilai data yang dimasukkan, misal jika diberi koma antara angka (20.19) akan dianggap sebagai float

17. Untuk memberi komentar dalam PHP dapat dilakukan dengan // komentar, bisa juga misal untuk komentar yang panjang dengan menggunakan /\* Komentar \*/ . dengan menggunakan komentar kita akan mematikan coding nya

18. Dengan mendeklarasikan variabelnya didepan dalam var\_dump maka tipe data nya dapat berubah misal var\_dump ((int)$angka); maka isi dalam variabel angka yang awalnya string akan menjadi int

19. Function digunakan untuk menyimpan sementara echo yang nanti nya akan dipanggil

20. Cara memanggil function hanya dengan memanggil namanya saja contoh belajar(); . Function ini juga bisa dipanggil berulang-ulang

21. Function juga bisa diisi kurungnya dengan variabel untuk membuat web menjadi lebih dinamis, bisa juga mengganti ukuran jika akan dipanggil dalam tanda kurung, contoh luasPesegi(10.20)

22. echo di dalam function bisa diganti dengan return, arti return yaitu mengeluarkan. Tetapi memanggilnya harus menggunakan echo namafunction();

23. pemanggilan function bisa ditambahi dengan operasi matematika lagi dapat diubah angkanya didalam kurung dan tambahan operasi matematika diisi diluar kurung

24. Operator matematika dalam PHP ada kali(\*), bagi(/), tambah(+), kurang(-), dan untuk mencari sisa pembagian (%)

25. Jika misal ada pembagian lalu hasilnya koma, kita bisa membulatkan. Jika ingin bulat ke atas menggunakan echo round($nama);, jika ingin bulat ke bawah menggunakan echo floor($nama)

26. Ada juga operator logika yaitu hasilya hanya benar atau salah, jika benar bernilai 1, jika salah bernilai 0 atau tidak ada

27. Operator logika ada > (lebih besar), < (lebih kecil), == (sama dengan), != (tidak sama dengan

28. Ada juga increment atau naik contoh $a++; atau decrement atau turun contoh $a--;

29. Untuk menambahkan operator string bisa digabung dengan titik(.)

30. Kondisi benar atau salah yang tadinya hanya berisi 1(benar) dan 0(salah) saat ingin menampilkan variabel dapat diubah dengan menggunakan if

31. Pemanggilan variabel dapat diganti dengan dimasukkan kedalam, misal if ($nama < 30){}

32. Saya mengerti untuk dapat memberikan kondisi yang lain atau kondisi salah dapat ditambah dengan else {echo ‘Nama’;}

33. Saya mengerti bahwa kita juga bisa menambahkan if lagi didalam if untuk pengujian yang lain

34. Pengujian IF selalu dimulai berurutan dari atas, jika yang atas tidak cocok / kondisi salah maka akan turun lagi ke bawahnya hingga menemukan yang cocok, jika tidak ada satupun yang cocok maka hasilya akan false (0)

35. Pengujian IF juga bisa didefiniikan dahulu dengan angka minus

36. Untuk pengujian If kita bisa menambahkan 2 pengujian dengan &&, misal  if ($nilai >= 0 && $nilai <= 100)

37. kondisi if ($nilai >= 0 && $nilai <= 100), dalam kondisi &&(and) kedua pengujiannya nya harus benar, jika salah satu salah maka semuanya akan salah

38. kondisi if ($nilai >= 0 || $nilai <= 100), dalam kondisi || (or) kedua pengujiannya nya tidak selalu harus benar semua, jika salah satu salahdan satunya benar maka hasilnya akan tetap benar

38. Saya mengerti bahwa tanda <=100 itu artinya kurang dari 100 tetapi 100 nya ikut masuk juga dalam hitungannya. Berbeda dengan < 100 itu menghitungya mulai 99, 100 nya tidak ikut

39. Saya mengerti bahwa switch case dalam PHP digunakan sebagai selector dari case-case yang akan dipanggil oleh variabel

40. Kondisi default pada switch case digunakan jika tidak ada kondisi/selector yang cocok dengan variabel

41. Selector juga hampir sama dengan IF yaitu menguji semua kondisi dari atas dan mencocokannya, jika tidak ada maka akan masuk ke kondisi default(Switch case), dan else(IF)

42. case ‘value’; didalam kondisi switch itu artinya berfungsi seperti select case jika ingin menampilkan cocok atau tidak tinggal memanggil value / dapat juga diganti dengan yang lain sesuai dengan kebutuhan

43. Perulangan (loop) merupakan sesuatu yang dikerjakan berulang – ulang artinya juga dalam perulangan ada awal dan ada akhirnya

44. Dalam PHP untuk membuat loop menggunakan for, contoh for ($i=1; $i <=12 ; $i++) {}

45. Dalam kondisi for ($i=1; $i <=12 ; $i++) {  
 echo $i;}   
artinya kondisi i dimulai dari satu dan sampai 12, serta arti $i++ adalah setiap kali berjalan dia akan naik satu-satu, seterusnya

46. Setiapo kondisi for dia akan selalu mengecek dengan dicocokkan dengan kondisi variabelnya misal yang tadi berarti akan menampilkan 1,2,3,4,5,6,7,8,9,10,11,12 (naik satu-satu)

47. Dalam pemanggilan for kita juga bisa menambahkan string lain bisa kalimat, angka tapi harus disambung terlebih dahulu dengan titik(.)

48. kondisi for ($i=1; $i <=12 ; $i++) $i nya dapat diubah menjadi for ($i=1; $i <=12 ; $i=$i+2) dan hasilnya akan ditambah masing masing 2/ naik sebanyak 2 selama itu masih diantara 1-12

49. Intinya, contoh for ($i=1; $i <=12 ; $i++) artinya $i=1; (mulainya), $i<=12(berakhirnya), $i++(langkahnya).

50. Ada juga perulangan menurun yaitu kebalikannya, yang diisikan terlebih dahulu dari besar ke kecil, contohnya for ($i=12; $i >=1 ; $i--), artinya yaitu dari 12,11,10,9,8,7,6,5,4,3,2,1 dan harus menggunakan $i—agar ke bawah atau turun. Bisa juga $i—diganti dengan $i-2 untuk langkahnya turun masing-masing sebanyak 2

51. Perulangan while dan for sebenarnya memiliki fungsi yang sama, hanya saja cara penulisanya yang beda, jika for adalah for ($i=12; $i >=1 ; $i--){}, while adalah while ($a <= 10) {  
  $a++; }

52. Ada juga perulangan do while fungsinya sama sepert for dan while, cara penulisannya yaitu do {  
 # code...  
 $a++;  
 } while ($a <= 10);

53. Untuk penulisan perulangan while dan do while harus menuliskan sendiri langkahnya ($a++)

54. Array adalah seperti sebuah tabel dan ada isi-isinya serta kita bisa juga menambahkan elemennya. Dengan menggunakan array kita bisa menyimpan banyak data dengan variabel dan isinya sebagai data

55. Cara menampilkan array dapat menggunakan var\_dump($nama);, dan juga var\_dump dapat melihat tipe data nya

56. Saya mengerti bahwa dalam menulis array dapat menulis tipe data yang berbeda, contohnya     $nama = array("Joni", "Tejo", "Budi", "Siti",100,2.5);

57. Array selalu dimulai dari 0, jadi cara memanggilnya harus dimulai dari 0, contoh cara memanggilnya yaitu ec ho $nama[0];

58. Untuk dapat menampilkan semua array kita dapat menggunakan loop, contohnya for ($i=0; $i <6 ; $i++) {   
  echo $nama[$i]. '<br>';}

59. Untuk dapat menampilkan semua array juga kita bisa menggunakan foreach, contohnya foreach ($variabel as $key ) {  
 echo $key. "<br>”}

60. Array asosiatif adalah array yang isinya memiliki pasangan lagi dan untuk menunjukkan pasangannya diberi tanda =>. Contoh array asosiatif yaitu  $nama = array (  
 "Joni" => "Surabaya",  
         "Budi" => "Malang") . untuk memisahkan antar array digunakan tanda koma(,) dan juga jika sudah ke terakhir tidak perlu diberi koma lagi

61. Dalam array asosiatif jika di var\_dump akan ada kurung staples[] itu berarti menunjukkan setiap array atau key nya

62. Saya bisa memanggil array asosiatif yaitu denga memangil nama key nya, contoh echo $nama [‘Joni’];

63. Saya bisa menampilkan isi dari array asosiatif dengan menggunakan foreach, contohnya foreach ($nama as $key => $value) {  
 echo $key."=>".$value;  
       } . yang dipanggil bisa hanya key atau bisa hanya value nya (disesuaikan saja)

64. Penulisan array asosiatif biasa dapat diganti dengan, contoh :  
 $nama["Joni"]="Surabaya";  
     $nama["Budi"]="Malang Raya";  
 $nama["Tejo"]="Jakarta";  
     $nama["SIti"]="Sidoajo";

65. Saya mengerti bahwa foreach dapat digunakan untuk menampilkan array

66. Cara memanggil array dimensi menggunakan foreach yaitu yang di echo harus key nya, contoh echo $key; . Yang dimaksud kunci yaitu jika sudah memegang kuncinya maka kita akan dapat isinya

67. Untuk memanggil array asosiatif menggunakan foreach harus disertakan $key dan $value nya juga, contoh foreach ($variabel as $key => $value) {  
 echo $key. '='.$value;}

68. Saya mengerti bahwa foreach ($variabel as $key => $value) $key dan $value tidak harus selalu begitu, kita juga bisa menggantinya seperti foreach ($variabel as $k => $v), dsb. Variabel nya itu bebas asal dipanggil nya dengan variabel yang sama juga

69. Saya mengerti bahwa require\_once itu digunakan untuk mengambil / memasukkan semua isi dari file lain ke dalam file sekarang, contoh pemanggilannya yaitu require\_once "namafile.php";

70. Jika misal kita memanggil file require\_once dua kali maka yang ditampilkan hanya yang pertama karena sudah pernah dipanggil

**Saya Belum Mengerti**

1.