



MODUL PERTEMUAN 11

KKTKJ - K2

Ditulis dengan penuh **cinta**💕

Rahmat Subandi

Table of Contents

Apersepsi	3
Persiapan	3
Apa itu PHP?	5
Komentar dalam File PHP	7
Contoh Komentar Inline	7
Contoh Komentar Multiple-Lines	7
Apa Itu Tipe Data?	9
Macam-Macam Tipe Data Dalam PHP	9
Apa itu Variabel?	10
Peraturan Penamaan Variabel	10
Pembuatan Variabel	11
Tipe Data Integer	11
Tipe Data Float	12
Tipe Data Boolean	13
Tipe Data String	14
Tipe Data Array	15
Tugas	17
Pengumpulan Praktik 11	19

Apersepsi

Dalam pertemuan ini kalian akan Belajar PHP Dasar, kalian akan mempelajari bahasa pemrograman PHP dari bagiannya yang paling dasar. Mulai dari file PHP, variabel, tipe data, logika, fungsi dan hal-hal lainnya.

Pada pertemuan 11 ini kalian akan membahas mulai dari persiapan yang harus dilakukan sebelum memulai pemrograman PHP, kalian juga akan memahami tentang apa itu file PHP dan cara menjalankannya.

Persiapan

1. Silakan kalian buat akun pada website phpsandbox.io, kalian akan menggunakan website ini untuk menjalankan file PHP yang kalian buat serta melihat hasilnya dan pengumpulan tugas yang akan datang.
2. Kalian bisa menggunakan Email Akun Belajar ataupun Akun pribadi. Saran dari saya, buat saja menggunakan akun yang sama emailnya saat kalian membuat Replit, sehingga tidak bingung nantinya jika banyak akun.
3. Pastikan kalian membuat username dengan nama kalian sendiri untuk kemudahan Instruktur dalam penilaian, username tidak boleh menggunakan spasi. Misalnya jika namanya **Budi Siregar** maka username bisa **budisiregar**.
4. Setelah kalian berhasil mendaftar silakan cek email kalian untuk verifikasi akun, jika tidak ada di kotak masuk coba cek di kotak spam.
5. Kembali lagi ke website phpsandbox.io dan login dengan akun yang sudah kalian buat.
6. Jika sudah berhasil login, silakan kalian buat project baru dengan cara klik tombol **Create Notebook** dan beri nama projectnya dengan format **Pertemuan 11 - NAMA - TKJ A/B**.
7. Pada bagian **Template** silakan kalian gulir ke bawah dan pilih **Standart**.
8. Klik tombol **Create Notebook** untuk membuat project baru, proses pembuatan project akan memakan waktu yang lama, jadi silakan kalian tunggu hingga selesai, bagaimanapun ini tergantung dari koneksi internet.
9. Jika pada poin 6 - 8 bingung, silakan kalian tunggu instruktur untuk memberikan panduan lebih lanjut.
10. Untuk tata cara pengumpulan tugas akan dijelaskan pada Instruktur.

CATATAN

Di setiap halaman akan ada potongan kode, silakan kalian tulis ulang untuk penilaian Praktik 11 kalian.

Apa itu PHP?

Kita masuk dari pembahasan yang pertama, yaitu tentang file PHP.

File PHP adalah file yang berekstensi ***.php**, diawali dengan sintaks **<?php** dan diakhiri dengan sintaks **?>**.

Contoh:

```
<?php  
  
echo "Halo dunia!";
```

Kode 1: Hello world dengan php

Atau bisa kalian singkat dengan:

```
<?= "Halo dunia!"; ?>
```

Tanda **<?=** adalah singkatan dari **<?php echo**.

Jika file murni PHP saja, tidak bercampur dengan file yang lain. Kita boleh untuk tidak menutup file tersebut dengan **?>** seperti contoh pada **Kode 1** di atas. Dan ini adalah cara yang disarankan.

Ada pun jika kode program kita bercampur dengan “konten” non-php. Maka kita perlu menggunakan sintaks penutup (**?>**) untuk memperjelas mana yang harus dieksekusi sebagai file php, dan mana yang bukan.

Perhatikan kode program pada **Kode 2** di bawah di mana sintaks PHP bercampur dengan sintaks HTML.

```
<!DOCTYPE html>
<html>
  <body>

    <h1>Halaman HTML</h1>
    <?php
      echo "Halo dunia!";
    ?>

  </body>
</html>
```

Kode 2: *Hello world php yang bercampur dengan konten non php*

Kode program pada contoh **Kode 2** di atas harus disimpan dengan ekstensi **.php**. Jika tidak, maka itu akan dianggap teks HTML biasa.

Komentar dalam File PHP

Kita telah mengetahui dari contoh di atas bahwasanya sintaks php adalah semua yang diapit oleh tag `<?php` dan `?>`. Akan tetapi, ada satu bagian file php yang keberadaannya seperti tidak ada, alias tidak dianggap: yaitu baris komentar.

Baris komentar dalam php ada 2 jenis: *inline* dan *multiple lines*. Anda bisa memanfaatkan komentar dalam PHP untuk menulis catatan tentang fungsi atau alur dari kode program yang anda tulis.

Contoh Komentar Inline

Pada PHP, ada dua cara untuk mendefinisikan komentar *inline*. Yang pertama seperti umumnya komentar pada bahasa pemrograman yang lain, yaitu menggunakan sintaks *double slash* (`//`). Dan cara yang kedua adalah dengan menggunakan tanda pagar atau *hashtag* (`#`).

Perhatikan contoh pada **Kode 3** berikut:

```
<?php

# Ini adalah komentar satu baris dengan tanda pagar (#)
// ini juga komentar satu baris tapi dengan tanda double-slash (//)
echo "Halo dunia!";
```

Kode 3: komentar inline php

Contoh Komentar Multiple-Lines

Jenis komentar yang kedua adalah komentar *multiple lines*. Ia diawali tanda `/**` dan diakhiri dengan tanda `*/`. Agar lebih bagus, biasanya setiap baris komentar akan ditambahkan tanda bintang `*`.

```
<?php

/**
 * Terkadang anda ingin menggunakan komentar dalam bentuk beberapa baris
 * Anda bisa menulisnya seperti ini
 *
 * NB: Ini biasa digunakan untuk menjelaskan logika/alur program yang
 cukup panjang
 */

echo "Halo dunia!";
```

Kode 4: komentar multiple-lines php

Seperti yang anda saksikan pada **Kode 3** dan **Kode 4**, kita bisa menambahkan komentar pada file php kita tanpa mempengaruhi program kita sama sekali. Untuk komentar *inline*, kita bisa menggunakan tanda `#` mau pun tanda `//`. Ada pun untuk *multiple lines*, anda bisa menggunakan tanda `/**` dan ditutup dengan tanda `*/`.

Apa Itu Tipe Data?

Tipe data adalah klasifikasi jenis data atau bentukan dari suatu data. Ia menjelaskan suatu data: dari jenis apakah ia tersusun? Apakah bilangan riil? Atau kah bilangan pecahan? Atau kah ia data yang tersusun dari bentukan karakter?

Intinya tipe data adalah klasifikasi jenis dari data yang kita ingin simpan dalam sebuah variabel. Hampir seluruh bahasa pemrograman yang ada mendukung berbagai macam jenis tipe data, seperti: integer untuk bilangan riil, boolean untuk `true` dan `false`, string untuk kumpulan karakter, dan sebagainya 1.

Macam-Macam Tipe Data Dalam PHP

Ada berbagai macam tipe data: mulai dari tipe data asli dan tipe data buatan. Untuk PHP sendiri, ia mendukung setidaknya 8 tipe data skalar. Akan tetapi dalam pertemuan ini, sementara kita akan fokuskan dahulu pada pembahasn 5 tipe data saja, karena 5 tipe data ini adalah tipe data yang paling dasar. 5 tipe data tersebut adalah:

Tipe Data	Keterangan
Integer	Tipe data yang berupa bilangan bulat. Contoh: 1, 2, 3, 4, 5, dan seterusnya.
Float	Tipe data yang berupa bilangan riil. Contoh: 1.1, 2.2, 3.3, 4.4, 5.5, dan seterusnya.
String	Tipe data yang berupa kumpulan karakter/teks yang diapit oleh tanda <code>' '</code> atau <code>" "</code> . Contoh: "Hello World", "Halo Dunia", "Aku Superman", dan seterusnya.
Boolean	Tipe data yang berupa <code>true</code> atau <code>false</code> .
Array	Tipe data yang berupa kumpulan / himpunan data.

PHP adalah bahasa pemrograman yang bersifat *dynamic typing*, yang artinya ia tidak memiliki aturan ketat terhadap pendefinisian tipe data pada setiap variabel. PHP akan otomatis menentukan tipe data dari suatu variabel tertentu ketika program dijalankan.

Apa itu Variabel?

Variabel adalah suatu "wadah" yang digunakan untuk menyimpan suatu data atau nilai. Kita bisa menyimpan berbagai macam data dari berbagai macam tipe, misalkan kita menyimpan data teks untuk nama mahasiswa, atau data desimal untuk nilai ipk mahasiswa, dan seterusnya.

Peraturan Penamaan Variabel

Untuk menyimpan suatu data pada variabel, kita perlu memberi nama terhadap variabel tersebut.

Dalam PHP, terdapat beberapa peraturan dalam pemberian nama variabel sebagaimana pada tabel berikut:

Peraturan	Keterangan
diawali tanda \$	Setiap nama variabel dalam bahasa pemrograman PHP didefinisikan dengan tanda \$ lalu diikuti oleh nama variabel itu sendiri
nama diawali huruf atau underscore (_)	Nama variabel PHP harus diawali huruf, atau tanda underscore (_). Kita bisa membuat variabel dengan nama <code>\$_nilai</code> atau <code>\$nilai123</code> akan tetapi tidak bisa membuat variabel dengan nama \$1nilai
case sensitive	PHP membedakan huruf besar dan kecil dalam penamaan variabel. Maka variabel <code>\$nilai</code> , <code>\$niLai</code> dan juga <code>\$nILAi</code> dianggap 3 variabel yang berbeda
hanya boleh huruf dan angka	nama variabel hanya boleh tersusun dari huruf <code>[a-z]</code> atau <code>[A-Z]</code> , dan juga angka <code>[0-9]</code> . Kita tidak bisa memberi nama variabel misalkan dengan tanda <code>^</code> atau <code>&</code> dan lain sebagainya. Kita juga tidak bisa menggunakan spasi dalam penamaan variabel di PHP
inisiasi dengan tanda =	Kita bisa memberi nilai terhadap suatu variabel dengan menggunakan operator sama dengan (<code>=</code>). Contoh: <code>\$nama = "Nurul Huda";</code>

NB

Penamaan suatu variabel disarankan agar menggunakan nama yang sesuai dengan tugas variabel tersebut. Misalkan menggunakan nama `$nilaiMatematika` untuk menyimpan nilai matematika, dan tidak menggunakan nama `$a` untuk menyimpan nama hari misalnya.

Pembuatan Variabel

Untuk pembuatan variabel dalam PHP, kombinasi sintaksnya adalah sebagai berikut:

```
<?php  
  
$namaVariabel = [nilai variabel];
```

Bagian `$namaVariabel` adalah nama yang anda berikan untuk variabel tersebut, sedangkan `[nilai variabel]` adalah nilai yang akan anda masukkan ke dalam variabel tersebut.

Nilai variabel ini lah yang menentukan tipe data dari variabel itu sendiri. Jika anda memasukkan angka, maka tipe datanya menjadi tipe data numeric (*integer/float*), jika anda masukkan `true/false` maka jadinya adalah `boolean`, dan seterusnya.

Tipe Data Integer

Tipe data yang pertama adalah `integer`. Ia adalah tipe data yang digunakan untuk menyimpan bilangan bulat.

Kita akan membuat variabel dengan tipe data `integer` lalu menampilkannya dengan perintah `echo`. Kita juga bisa menampilkan tipe data dari variabel tersebut dengan perintah `var_dump`.

Perhatikan kode program berikut:

```
<?php
# inisiasi dan inisialisasi variabel
$a = 10;
$b = 5;
$c = $a + 5;
$d = $b + (10 * 5);
$e = $d - $c;

# Tampilkan data dengan perintah echo
echo "Variabel a: {$a} <br>";
echo "Variabel b: {$b} <br>";
echo "Variabel c: {$c} <br>";
echo "Variabel d: {$d} <br>";
echo "Variabel e: {$e} <br>";

# mengetahui tipe data dari variabel
var_dump($e);
```

Kode program di atas akan menghasilkan output seperti berikut:

```
Variable a: 10
Variable b: 5
Variable c: 15
Variable d: 55
Variable e: 40
int(40)
```

Tipe Data Float

Untuk tipe data **float**, caranya sama saja dengan **integer**. Hanya saja, ia menerima data desimal dengan angka **.** sebagai pembaginya.

Perhatikan dan praktikkan contoh berikut:

```

<?php

$nilaiMatematika = 5.1;
$nilaiIPA = 6.7;
$nilaiBahasaIndonesia = 9.3;

# hitung nilai rata-rata
$rataRata = ($nilaiMatematika + $nilaiIPA + $nilaiBahasaIndonesia) / 3;

# Tampilkan data

echo "Matematika: {$nilaiMatematika} <br>";
echo "IPA: {$nilaiIPA} <br>";
echo "Bahasa Indonesia: {$nilaiBahasaIndonesia} <br>";
echo "Rata-rata: {$rataRata} <br>";

# lihat tipe data dari variabel $rataRata
var_dump($rataRata);

```

Kode program di atas akan menghasilkan output seperti berikut:

```

Matematika: 5.1
IPA: 6.7
Bahasa Indonesia: 9.3
Rata-rata: 7.0333333333333
float(7.0333333333333)

```

Tipe Data Boolean

Tipe data **boolean** hanya bisa menampung nilai **true** atau **false**.

Tipe data ini adalah tipe data yang paling simpel, akan tetapi butuh logika yang kuat untuk bisa memanfaatkannya dengan benar.

Berikut ini contoh penggunaan tipe data **boolean** pada PHP:

```
<?php

$apakahSiswaLulus = true;
$apakahSiswaSudahUjian = false;

var_dump($apakahSiswaLulus);
echo "<br>";
var_dump($apakahSiswaSudahUjian);
```

Output:

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

Tipe Data String

Tipe data **string** adalah tipe data yang digunakan untuk menyimpan teks. Semua teks tersebut diapit oleh tanda petik satu (' ') mau pun tanda petik dua (" ").

Pada contoh-contoh di atas, sebenarnya kita telah menggunakan tipe data ini ketika kita memanggil fungsi **echo** yang diikuti setelahnya dengan teks.

Akan tetapi data **string** tersebut tidak kita simpan ke dalam sebuah variabel. Akan tetapi langsung kita tampilkan dengan perintah **echo**.

Sekarang, saya akan beri contoh bagaimana membuat variabel, lalu mengisinya dengan data **string**.

Perhatikan kode program berikut:

```

<?php

$namaDepan = "Ibnu"; # pakai tanda petik dua
$namaBelakang = 'Jakaria'; # pakai tanda petik satu

# menggabungkan dua variabel dengan tanda
# petik dua
$namaLengkap = "{$namaDepan} {$namaBelakang}";

# anda juga bisa menggabungkan string dengan menggunakan tanda titik (.)
$namaLengkap2 = $namaDepan . ' ' . $namaBelakang;

# [Tampilkan Data]
# kita bisa memasukkan variabel lain jika menggunakan tanda petik dua
echo "Nama Depan: {$namaDepan} <br>";
# ada pun jika pakai tanda petik satu, kita tidak bisa memasukkan
variabel
# di dalam string akan tetapi menggabungkannya dengan operator titik (.)
echo 'Nama Belakang: ' . $namaBelakang . '<br>';

echo $namaLengkap;

```

Output:

```

Nama Depan: Ibnu
Nama Belakang: Jakaria
Ibnu Jakaria

```

Terdapat beberapa hal yang perlu dibahas terkait tipe data string. Seperti misalnya memotong teks string, me-replace suatu kata dalam string, mengubah string menjadi UPPERCASE atau menjadi lowercase, dan sebagainya.

Hal tersebut akan kita pelajari pada akhir-akhir pembelajaran PHP dasar ini pada pembahasan: **Belajar Memanipulasi String Dalam PHP**.

Tipe Data Array

Tipe data **array** berfungsi untuk menyimpan himpunan data. Himpunan data tersebut diapit oleh tanda kurung siku (**[]**).

Sebagai contoh, saya memiliki 3 mahasiswa, dan saya ingin menyimpan ketiga nama mahasiswa dalam variabel.

Maka saya bisa melakukan hal tersebut dengan menggunakan tipe data **array** sebagai berikut:

```
<?php  
  
$listMahasiswa = ["Wahid Abdullah", "Elmo Bachtiar", "Lendis Fabri"];
```

Untuk mengakses isi dari variabel **array**, kita bisa menggunakan indeks. Indeks dimulai dari **0**. Sehingga jika saya akan menampilkan nama pertama dari variabel **\$listMahasiswa**, saya akan menggunakan indeks **0** seperti di bawah:

```
<?php  
  
echo $listMahasiswa[0]; // "Wahid Abdullah"
```

Contoh di atas masih sangat sederhana sekali. Array merupakan tipe data yang cukup kompleks. Kalian tidak akan mempelajarinya sampai detil pada tahap ini agar tidak terlalu membingungkan.

Tugas

CATATAN

Pastikan hasil nya di cetak dalam format HTML.

1. Membuat Variabel Integer

- **Soal:** Buatlah variabel bernama `$umur` dan isi dengan nilai `25`. Tampilkan nilai variabel tersebut dalam format HTML.

2. Membuat Variabel Float

- **Soal:** Buatlah variabel bernama `$berat` dan isi dengan nilai `65.5`. Tampilkan nilai variabel tersebut dalam format HTML.

3. Membuat Variabel String

- **Soal:** Buatlah variabel bernama `$nama` dan isi dengan nama Anda. Tampilkan nilai variabel tersebut dalam format HTML.

4. Membuat Variabel Boolean

- **Soal:** Buatlah variabel bernama `$isStudent` dan isi dengan nilai `true`. Tampilkan nilai variabel tersebut dalam format HTML.

5. Membuat Array Sederhana

- **Soal:** Buatlah array bernama `$hewan` yang berisi tiga hewan: "Kucing", "Anjing", dan "Burung". Tampilkan hewan kedua dalam format HTML.

6. Mengakses Elemen Array

- **Soal:** Buatlah array bernama `$angka` yang berisi tiga angka: 10, 20, dan 30. Tampilkan angka pertama dalam format HTML.

7. Membuat Variabel Gabungan

- **Soal:** Buatlah dua variabel string, `$kata1` dan `$kata2`, dan isi dengan kata

"Belajar" dan "PHP". Gabungkan dan tampilkan hasilnya dalam format HTML.

8. Operasi Penjumlahan

- **Soal:** Buatlah dua variabel integer `$a` dan `$b`, isi dengan angka 5 dan 10, lalu hitung dan tampilkan hasil penjumlahannya dalam format HTML.

9. Menghitung Rata-rata

- **Soal:** Buatlah tiga variabel float bernama `$nilai1`, `$nilai2`, dan `$nilai3`, isi dengan nilai 8.5, 7.0, dan 9.0. Hitung rata-rata dan tampilkan dalam format HTML.

10. Menampilkan Tipe Data

- **Soal:** Buatlah variabel bernama `$status` dengan nilai `false`. Gunakan `var_dump` untuk menampilkan tipe data dan nilainya dalam format HTML.

Pengumpulan Praktik 11

Silakan kumpulkan di link berikut:

<https://forms.gle/ZtEw3jjueqCZCPT29>

CATATAN

Pengumpulan tugas paling lambat **10 Nopember 2024 Pukul 21:00 WIB**. Tidak ada toleransi dengan alasan yang kurang jelas.