

Tujuan Pembelajaran

Setelah mempelajari bab ini anda diharapkan mampu:

- Menampilkan kode PHP ke layar
- Memahami penggunaan komentar pada kode
- Memahami konsep variabel pada PHP
- Memahami beberapa tipe data pada PHP
- Memahami operator pada PHP dan cara menggunakannya

Menuliskan Kode PHP Pertama

Dibagian awal ini kita akan menuliskan kode PHP sederhana dan menampilkannya ke layar. Kode di bawah ini merupakan instruksi untuk menampilkan tulisan “Hello, World!” ke layar.

```
1 <?php
2 echo "Hello, World!";
```

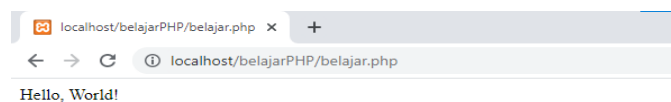
Pada baris pertama merupakan tag untuk memulai kode PHP. Tag tersebut wajib dituliskan ketika memulai kode PHP. Di baris kedua terdapat *syntax* **echo** yang merupakan *syntax* untuk menampilkan tulisan ke layar, text yang berada diantara tanda kutip “” menandakan pesan yang ditampilkan merupakan sebuah tipe data *string* atau teks, setiap baris kode PHP selalu diakhiri dengan tanda titik koma (;). Kita juga dapat menuliskan kode PHP sebagai berikut.

```
1 <?php
2 echo "Hello, World!";
3 ?>
```

Pada baris ketiga kode di atas terdapat penutup tag untuk kode PHP. Hal ini biasa digunakan ketika kode PHP disisipkan diantara kode HTML.

Untuk menjalankan kode tersebut simpan menggunakan ekstensi .php (dot php), dalam hal ini tuliskan nama file **belajar.php**. Simpan kode tersebut ke direktori localhost. Direktori tersebut terletak di <C://xampp/htdocs/> letak direktori tersebut tidak mesti berada di C bisa saja di D atau lainnya tergantung dimana anda menginstallnya. Di dalam folder htdocs buat sebuah folder dengan nama **belajarPHP** lalu simpan file **belajar.php** di folder tersebut.

Setelah itu jalankan XAMPP pada bagian Apache klik tombol start sampai statusnya berwarna hijau. Selanjutnya buka browser ketikkan alamat **localhost** diikuti nama folder **belajarPHP** diikuti nama file **belajar.php**, sebagai contoh anda bisa mengetikkan **localhost/belajarPHP/belajar.php** atau <http://localhost/belajarPHP/belajar.php> dan hasil sebagai berikut:



Dari contoh di atas kode PHP berhasil dijalankan.

Codelab:

Sekarang cobalah untuk menampilkan data diri anda ke layar menggunakan PHP.

Komentar

Instruksi yang ditulis dalam suatu program tetapi tidak dijalankan oleh komputer disebut “comments”. Sebuah komentar akan dilewatkan oleh interpreter atau compiler, sehingga tidak akan memengaruhi alur program yang ditulis. Komentar ini bisa digunakan sebagai dokumentasi atau penjelasan dari kode yang kita tulis.

Terdapat dua metode untuk memberikan komentar. Pertama, untuk memberikan komentar pada satu baris saja, kita bisa gunakan tanda dua garis miring (//) di awal baris.

```
1 <?php
2 //Ini merupakan komentar satu baris, kode di bawah tidak akan dijalankan
3 // echo "Hello, World!";
```

Lalu, untuk memberikan komentar lebih dari satu baris kita bisa menggunakan tanda /* sebagai pembuka komentar dan tanda */ untuk penutup komentar. Teks apa pun yang berada di antara tanda tersebut akan dijadikan komentar dan tidak akan dieksekusi.

```
1 <?php
2 /* Ini merupakan komentar dengan lebih dari satu baris
3 Teks apapun yang berada disini akan dijadikan komentar.
4 Ketika menggunakan ini, jangan lupa untuk menutupnya.
5 */
```

Variabel

Ketika menulis sebuah program, kita memberi tahu komputer cara memproses informasi seperti mencetak teks ke layar atau melakukan operasi perhitungan. Untuk lebih mudah dalam penggunaan dan pemanggilan data, kita bisa memanfaatkan variabel. Variabel umumnya digunakan untuk menyimpan informasi atau nilai yang akan dikelola dalam sebuah program.

Pada PHP penulisan nama variabel harus diawali dengan tanda \$ (dollar sign / tanda dollar) lalu diikuti nama variabel. Nama variabel berupa kombinasi huruf angka atau tanda underscore (_). Nama variabel tidak boleh diawali dengan angka, tidak boleh menggunakan spasi.

Berikut contoh penulisan variabel yang benar pada PHP.

```
1 <?php
2 $nama;
3 $namaLengkap;
4 $_SESSION;
5 $nama1;
```

Ada beberapa karakteristik variabel pada bahasa PHP yaitu:

- Variabel PHP bersifat case sensitive, yang artinya nama variabel akan membedakan penggunaan huruf besar dan huruf kecil, jadi jika anda membuat 2 variabel dengan nama yang sama tapi berbeda penggunaan huruf besar kecilnya maka php akan menganggap 2 variabel tersebut berbeda.

```
1 <?php
2 $nama = "Akbar";
3 $Nama = "Ilham";
4 ?>
```

- Variabel PHP tidak memerlukan deklarasi, berbeda dengan bahasa pemrograman lainnya seperti Pascal, C, C++, C#, Java dan beberapa pemrograman lainnya, yang mengharuskan variabel untuk dideklarasikan sebelum digunakan, namun dalam PHP anda tidak perlu mendeklarasikan variabel sebelum menggunakan, dalam PHP anda bisa membuat variabel PHP dimanapun, bisa diawal ditengah ataupun diakhir dari kode program dan langsung bisa digunakan.

```
1 <?php
2 $nama = "Akbar";
3 echo "Nama saya adalah $nama";
4 // output: Nama saya adalah Akbar
```

- Variabel PHP tidak bertipe, jika pada bahasa pemrograman desktop seperti Java, Pascal, C jika anda membuat variabel, anda diharuskan menentukan tipe data yang digunakan variabel tersebut, semisal nih adalah variabel dengan nama panjang anda mendeklarasikan variabel itu sebagai integer, maka variabel itu hanya bisa diisi oleh angka, tidak bisa diisi oleh huruf (string), karena deklarasi awalnya adalah angka (integer) bukan string (huruf). PHP termasuk dalam Loosely Type Language, yaitu bahasa pemrograman yang variabelnya tidak terikat pada sebuah tipe data tertentu.

```
1 <?php
2 $x = 100; // nilai variabel x berisi angka (integer)
3 echo $x;
4 $x = "Akbar"; // nilai variabel x diubah menjadi huruf (string)
5 echo $x;
```

Untuk memberikan nilai pada variabel PHP, anda bisa menggunakan tanda = (sama dengan) atau Operator = (sama dengan) ini dikenal dengan istilah Assignment Operators.

```
1 <?php
2 $nama = "Akbar";
3 $email = "akbar@hotmail.com";
4 $angka = 100;
```

Pada baris 2 dan 3 kita memberi nilai variabel dengan tipe data string dan pada baris 4 kita memberi nilai dengan tipe data integer.

Untuk menampilkan isi dari variabel tinggal tuliskan menggunakan *syntax* **echo** seperti yang kita pelajari sebelumnya. Kita bisa menuliskan seperti berikut.

```
1 <?php
2 $nama = "Akbar";
3 echo $nama;
4
5 echo "Nama saya adalah $nama";
```

Pada baris 2 kita membuat variabel nama dan memberikannya nilai “Akbar” yang bertipe data string. Pada baris 3 kita menampilkan isi dari variabel nama menggunakan *syntax* echo. Di baris 4 ketika kita ingin menggabungkan variabel dengan string, kita bisa menuliskannya langsung diantara tanda kutip dua (“”).

Codelab:

Buatlah sebuah kode PHP yang memiliki tiga buah variabel yaitu nama, kota, dan sekolah anda. Lalu tampilkan seperti contoh berikut.

Nama saya adalah Akbar, saya tinggal di kota Banjarmasin, dan saya bersekolah di SMK ISFI.

Tipe Data

Tipe data merupakan pengklasifikasian data berdasarkan jenisnya. Dalam php untuk menggunakan tipe data tertentu anda tidak perlu untuk mendeklarasikan dulu di awal, seperti pada bahasa pemrograman lain semisal java, c, c# dan beberapa bahasa lainnya, tetapi setidaknya kita harus mengetahui ada tipe data apa saja di PHP.

Di PHP terdapat beberapa tipe data yaitu, integer, float, string, dan boolean

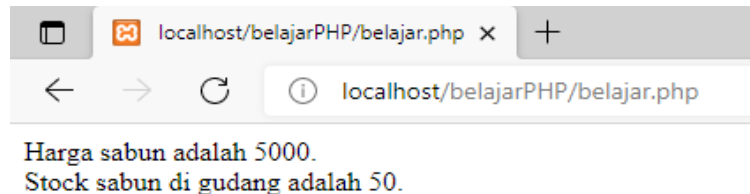
Tipe Data Integer

Tipe data integer atau biasanya lebih dikenal dengan tipe data untuk menyimpan angka bulat (bukan desimal), semisal 1,2,3,4,5,-10,-20 dll. Nilai di tipe data integer bisa bernilai positif (+) dan negatif

(-). Tipe data ini biasanya digunakan untuk menyimpan data seperti harga sebuah barang, jumlah mahasiswa, jumlah penduduk, stok barang, intinya adalah untuk nilai angka yang tidak ada desimalnya. Contoh penggunaan dari tipe data Integer sebagai berikut.

```
1 <?php
2 $harga = 5000;
3 echo "Harga sabun adalah $harga.<br>";
4
5 $jumlahBarang = 50;
6 echo "Stock sabun di gudang adalah $jumlahBarang.<br>";
```

Outputnya sebagai berikut.

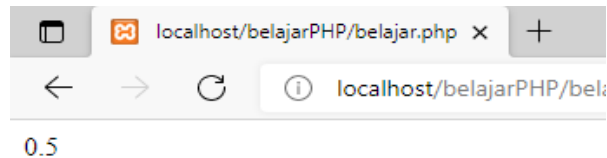


Tipe Data Float

Tipe data float ini digunakan untuk menyimpan data yang memiliki desimal diakhir angka, Tipe data float ini digunakan untuk variabel yang akan berisi angka yang memiliki angka desimal (pecahan), seperti angka hasil pembagian, nilai yang memiliki desimal seperti nilai ulangan, dan nilai angka yang tidak bisa disimpan dengan tipe data Integer. Contoh penggunaan dari tipe data Float sebagai berikut.

```
1 <?php
2 $angkaFloat = 0.50;
3 echo $angkaFloat;
```

Outputnya sebagai berikut.

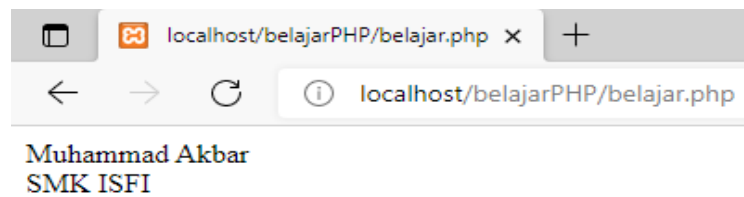


Tipe Data String

Tipe data string adalah tipe data yang digunakan untuk menyimpan data berupa string, sederhananya tipe data string digunakan untuk menyimpan text seperti kalimat atau sebuah kata. Karakter yang mendukung tipe data string adalah 256 karakter ASCII. Penulisan tipe data string diapit diantara tanda kutip dua (" "). Contoh penggunaan dari tipe data String sebagai berikut.

```
1 <?php
2 $nama = "Muhammad Akbar";
3 echo $nama;
4 echo "<br>";
5
6 $sekolah = "SMK ISFI";
7 echo $sekolah;
```

Outputnya sebagai berikut.



Tipe Data Boolean

Tipe data boolean ini memiliki 2 nilai yaitu true, atau false, anda bisa menuliskan **true**, **false** ataupun **TRUE**, **FALSE** (bisa menggunakan huruf kecil ataupun huruf besar). Tipe data boolean ini biasanya digunakan pada statement operasi logika, seperti if, dan juga sebagai pembanding dalam perulangan (looping). Untuk contoh dari tipe data boolean sebagai berikut.

```
1 <?php
2 $nilai_a = TRUE;
3 $nilai_b = FALSE;
```

Operator

Operator dalam bahasa pemrograman sendiri adalah simbol yang memberi tahu interpreter untuk melakukan operasi seperti matematika, relasional, atau logika untuk memberikan hasil tertentu. Pada bahasa PHP terdapat beberapa Operator yaitu assignment, aritmatika, perbandingan dan logika.

Operator Assignment

Operator Assignment atau operator penugasan digunakan untuk memasukkan nilai pada sebuah variabel, kita sebelumnya sudah banyak menggunakan operator Assignment, operator ini kita gunakan untuk mengisi nilai ke dalam variabel. Contoh operator Assignment sebagai berikut.

```
1 <?php
2 $angka = 100;
3 $nama = "Akbar";
```

Pada kode di atas baris 2 kita menugaskan variabel angka dengan nilai 100, dan pada baris 3 menugaskan variabel nama dengan nilai Akbar.

Operator Aritmatika

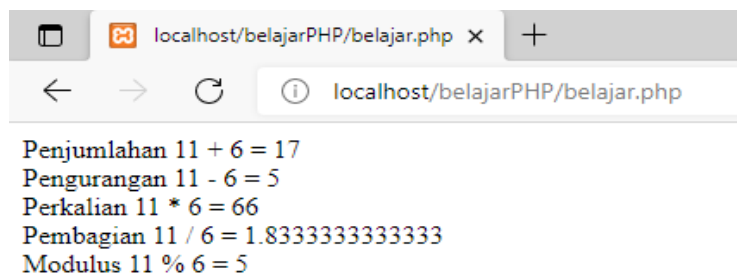
Operator aritmatika berfungsi untuk melakukan perhitungan matematika. Operator aritmatika meliputi perhitungan seperti penjumlahan, pengurangan, pembagian, serta perkalian. Berikut adalah simbol operator aritmatika.

Simbol	Fungsi	Keterangan
+	Penjumlahan	Penjumlahan untuk dua operand (a + b)
-	Pengurangan	Pengurangan untuk dua operand (a - b)
*	Perkalian	Perkalian untuk dua operand (a * b)
/	Pembagian	Pembagian untuk dua operand (a / b)
%	Modulus	Nilai hasil bagi dari dua operand (a % b)

Berikut penerapan operator aritmatika pada PHP.

```
1 <?php
2 $a = 11;
3 $b = 6;
4
5 echo "Penjumlahan $a + $b = ".$a + $b."<br>";
6 echo "Pengurangan $a - $b = ".$a - $b."<br>";
7 echo "Perkalian $a * $b = ".$a * $b."<br>";
8 echo "Pembagian $a / $b = ".$a / $b."<br>";
9 echo "Modulus $a % $b = ".$a % $b."<br>";
```

Keterangan kode di atas pada baris 2 dan 3 kita menugaskan variabel a bernilai 11 dan b bernilai 6. Di baris 5 sampai 9 kita melakukan operasi hitung menggunakan operator aritmatika. Outputnya sebagai berikut.



```
Penjumlahan 11 + 6 = 17
Pengurangan 11 - 6 = 5
Perkalian 11 * 6 = 66
Pembagian 11 / 6 = 1.8333333333333
Modulus 11 % 6 = 5
```

Operator Perbandingan

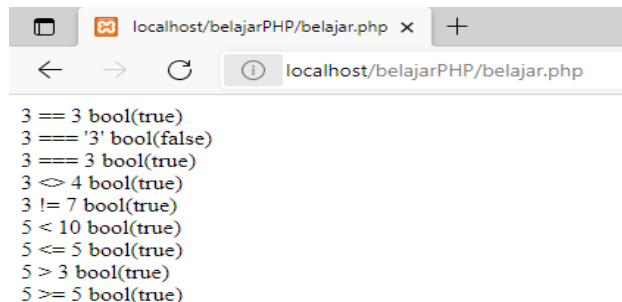
Operator perbandingan atau comparison operator digunakan untuk membandingkan 2 operand atau lebih. Pada umumnya Operator perbandingan digunakan pada kondisi if-else sebagai penentu tingkat kecocokan. Operator ini menghasilkan nilai boolean (**true** atau **false**). Berikut ini adalah jenis-jenis operator perbandingan yang ada di PHP :

Operator	Nama	Contoh	Keterangan
==	Sama dengan	\$a == \$b	Bernilai TRUE jika \$a sama dengan \$b (tanpa melihat tipe data)
===	Identik dengan	\$a === \$b	Bernilai TRUE jika \$a sama dengan \$b dan memiliki tipe data yang sama
<>	Tidak sama dengan	\$a <> \$b	Bernilai TRUE, jika \$a tidak sama dengan \$b (tanpa melihat tipe data)
!=	Tidak sama dengan	\$a != \$b	Bernilai TRUE, jika \$a tidak sama dengan \$b (tanpa melihat tipe data)
!==	Tidak identik dengan	\$a !== \$b	Bernilai TRUE, jika \$a tidak sama dengan \$b, dan memiliki tipe data yang sama
<	Kurang dari	\$a < \$b	Bernilai TRUE, jika \$a kurang dari \$b
>	Lebih dari	\$a > \$b	Bernilai TRUE, jika \$a lebih dari \$b
<=	Kurang dari sama dengan	\$a <= \$b	Bernilai TRUE, jika \$a kurang dari sama dengan \$b
>=	Lebih dari sama dengan	\$a >= \$b	Bernilai TRUE, jika \$a lebih dari sama dengan \$b

Berikut penerapan operator perbandingan pada PHP.

```
1 <?php
2 echo "3 == 3 "; var_dump(3 == 3);
3 echo "<br/>";
4 echo "3 === '3' "; var_dump(3 === '3');
5 echo "<br/>";
6 echo "3 === 3 "; var_dump(3 === 3);
7 echo "<br/>";
8 echo "3 <> 4 "; var_dump(3 <> 4);
9 echo "<br/>";
10 echo "3 != 7 "; var_dump(3 != 7);
11 echo "<br/>";
12 echo "5 < 10 "; var_dump(5 < 10);
13 echo "<br/>";
14 echo "5 <= 5 "; var_dump(5 <= 5);
15 echo "<br/>";
16 echo "5 > 3 "; var_dump(5 > 3);
17 echo "<br/>";
18 echo "5 >= 5 "; var_dump(5 >= 5);
```

Outputnya sebagai berikut.



```
3 == 3 bool(true)
3 === '3' bool(false)
3 === 3 bool(true)
3 <> 4 bool(true)
3 != 7 bool(true)
5 < 10 bool(true)
5 <= 5 bool(true)
5 > 3 bool(true)
5 >= 5 bool(true)
```

- Pada baris 2 menghasilkan nilai TRUE karena $3 == 3$
- Pada baris 4 menghasilkan nilai FALSE karena $3 === '3'$, seperti yang telah dijelaskan diawal untuk perbandingan yang menggunakan $===$ akan juga memperhatikan tipe data yang digunakan, karena dalam hal ini tipe datanya berbeda yang satu integer, dan yang satu adalah string karena itu menghasilkan nilai false
- Pada baris 6 menghasilkan nilai TRUE karena $3 === 3$ keduanya nilainya sama dan tipe datanya juga sama integer
- Pada baris 8 menghasilkan nilai TRUE karena 3 tidak sama dengan 4
- Pada baris 10 menghasilkan nilai TRUE karena 3 tidak sama dengan 7
- Pada baris 12 menghasilkan nilai TRUE karena 5 kurang dari 10
- Pada baris 14 menghasilkan nilai TRUE karena 5 kurang dari sama dengan 5
- Pada baris 16 bernilai TRUE karena 5 lebih besar daripada 3
- Pada baris 18 menghasilkan nilai TRUE karena 5 lebih besar sama dengan 5

Operator Logika

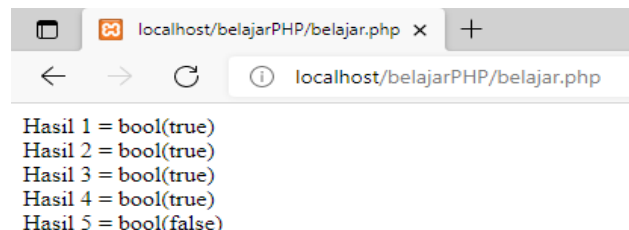
Operator Logika adalah operator yang digunakan untuk membandingkan nilai, dan hasil dari operator logika ini bertipe data boolean, karena nilainya adalah TRUE dan FALSE. Untuk jenis operator logika dalam PHP antara lain :

Nama dan Simbol	Contoh	Keterangan
AND (&&)	\$a AND \$b atau \$a && \$b	Bernilai TRUE jika \$a dan \$b keduanya bernilai TRUE
OR ()	\$a OR \$b atau \$a \$b	Bernilai TRUE jika salah satu dari \$a atau \$b bernilai TRUE
NOT (!)	!\$a	Bernilai TRUE jika \$a bernilai FALSE

Contoh penggunaan operator logika adalah seperti berikut ini :

```
1 <?php
2 $a = 3;
3 $b = 5;
4
5 $hasil_1 = ($a == 3 AND $b == 5);
6 $hasil_2 = ($a == 3 && $b == 5);
7 $hasil_3 = ($a == 3 OR $b == 6);
8 $hasil_4 = ($a == 2 || $b == 5);
9 $hasil_5 = (! $a == 3);
10 echo "Hasil 1 = ";var_dump($hasil_1);
11 echo "<br/>";
12 echo "Hasil 2 = ";var_dump($hasil_2);
13 echo "<br/>";
14 echo "Hasil 3 = ";var_dump($hasil_3);
15 echo "<br/>";
16 echo "Hasil 4 = ";var_dump($hasil_4);
17 echo "<br/>";
18 echo "Hasil 5 = ";var_dump($hasil_5);
```

Outputnya sebagai berikut.



```
Hasil 1 = bool(true)
Hasil 2 = bool(true)
Hasil 3 = bool(true)
Hasil 4 = bool(true)
Hasil 5 = bool(false)
```

- Hasil_1 akan menampilkan TRUE, karena pada Line 5 kita menggunakan operator AND dan nilai dari \$a dan \$b bernilai TRUE
- Hasil_2 akan menampilkan TRUE, karena pada Line 6 kita menggunakan operator && dan nilai dari \$a dan \$b bernilai TRUE
- Hasil_3 akan menampilkan TRUE, karena pada Line 7 kita menggunakan operator OR dan salah satu nilai dari \$a dan \$b bernilai TRUE
- Hasil_4 akan menampilkan TRUE, karena pada Line 8 kita menggunakan operator OR dan salah satu nilai dari \$a dan \$b bernilai TRUE
- Hasil_5 akan menampilkan FALSE, karena pada Line 9 kita menggunakan operator NOT dan nilai dari \$a adalah TRUE.

Rangkuman

Kita telah membahas beberapa hal di bab ini, antara lain:

- Membuat program pertama PHP, menggunakan *syntax* **echo** untuk menampilkan ke layar
- Membuat komentar pada program PHP
- Mengenal beberapa tipe data pada PHP seperti integer, string, boolean, float
- Mengenal operator pada PHP seperti assignment, aritmatika, perbandingan, dan logika