



EKSPLORASI PBW

# Dasar-Dasar Pemrograman PHP

Oleh : Rizki Septiana

# Tipe Data Number

## TIPE DATA NUMBER (INT & FLOAT)

- Di php terdapat dua tipe data untuk number
- Pertama, itu untuk integer(bilangan bulat)
- Kedua, itu float(bilangan desimal)

# Kode

```
echo "Tipe data number int".PHP_EOL;  
var_dump(123455);  
var_dump(123_234);  
var_dump(122222222);
```

```
echo "Tipe data number float".PHP_EOL;  
var_dump(1.23455);  
var_dump(1.111);  
var_dump(1_244.3);
```

# Tipe Data Boolean

- Tipe data boolean adalah tipe data paling sederhana di php
- Tipe data boolean adalah tipe data dengan nilai kebenaran (true/false)

# Kode

```
echo "Tipe data number Boolean".PHP_EOL;  
var_dump(TRUE);  
var_dump(true);  
var_dump(truE);
```

```
echo "Tipe data number Boolean".PHP_EOL;  
var_dump(FALSE);  
var_dump(false);  
var_dump(false);
```

# Tipe Data String

- Tipe data string adalah tipe data representasi dari teks
- String bisa mengandung kosong atau banyak karakter
- Penggunaannya bisa menggunakan kutip satu atau kutip dua
- Salah satu kelebihan menggunakan double quote adalah kita menggunakan escape sequence untuk beberapa hal, seperti `\n` untuk ENTER dan `\t` untuk TAB

# Kode

```
echo "Tipe data number String".PHP_EOL;  
echo 'Rizki'.PHP_EOL;  
echo 'Pemrograman Web';  
  
echo "\n";  
echo "Nama\t: Rizki\n";  
echo "Matkul\t: Pemrograman Web";
```

# Tipe Data String

- Namun ketika kita perlu membuat ENTER yang berulang atau TAB yang berulang itu akan menyulitkan dan tidak efisien ketika menggunakan `\n \t`
- Maka di PHP ada yang dinamakan heredoc
- Heredoc adalah fitur yang membuat String yang panjang, sehingga kita tidak perlu manual melakukan enter, tab atau yang lainnya secara manual



# Kode

```
echo<<<Tag
```

Ini adalah contoh String yang panjang  
dan juga perlu mengetik ENTER secara manual  
jika tidak menggunakan heredoc, "bisa menambahkan quote"  
Tag;

# Variabel

- Variabel adalah tempat untuk menyimpan data
- Di PHP variabel bisa menampung berbagai jenis tipe data dan bisa berubah=ubah tipe data
- Untuk membuat variabel kita bisa menggunakan tanda \$(dolar) diikuti dengan nama variabelnya
- Penamaan variabel tidak boleh mengandung spasi

# Kode

```
8
9  $name = "Arip";
0  $age = 58;
1  // $nama depan = "arif"; akan error
2
3  echo "Name : ";
4  echo $name;
5  echo "\n";
6  echo "Age : ";
7  echo $age;
8
```

# Variable Variables

- PHP memiliki kemampuan variable variables, yaitu membuat variabel dari string value variable
- Walaupun fitur ini ada, tapi fitur ini sangat membingungkan jika digunakan secara luas, jadi disarankan untuk tidak menggunakan fitur ini kecuali memang diperlukan
- Untuk membuat variable dari value variable kita bisa menggunakan \$\$ diikuti dengan nama variabelnya

# Kode

```
49  
50 $contoh = "Asep";  
51 $$contoh = "Ahmad Sunaedi";  
52  
53 echo "name : ";  
54 echo $contoh;  
55 echo "\n";  
56 echo "\$asep : ";  
57 echo $Asep;  
58
```

# Constant

- Variabel di PHP bersifat mutable, artinya bisa dirubah
- Jika kita ingin membuat variabel yang immutable maka tidak bisa dilakukan di PHP
- Kecuali sebagai gantinya, terdapat fitur yang namanya constant
- Constant adalah tempat untuk menyimpan data yang tidak bisa dirubah lagi setelah di deklarasikan
- Untuk membuat constant kita bisa menggunakan function `define()`
- Terdapat 2 paramater yang pertama nama variabel yang kedua valuenya
- Menggunakan `UPPER_CASE` agar kita tahu bahwa itu constant

# Kode

```
9  define("NAMA", "Indra Gunawan");
10 // define("NAMA", "Asep"); akan warning karena udah di define
11 define("UMUR", 45);
12 define("ASAL", "Sumatera");
13
14 echo "Nama : ";
15 echo NAMA;
16 echo "\n";
17 echo "Umur : ";
18 echo UMUR;
19 echo "\n";
20 echo "Asal : ";
21 echo ASAL;
22
23 // ASAL = "Bandung"; akan error
24
```

# Data NULL

- Nilai NULL merepresentasikan sebuah variabel tanpa nilai
- Saat membuat variabel, lalu ingin menghapus data yang terdapat di variabel tersebut, kita bisa menggunakan NULL untuk mengosongkan variabel tersebut
- Untuk membuat data NULL, kita bisa menggunakan kata kunci NULL



# Kode

```
74
75     $name = "Rizki";
76     $name = NULL;
77
78     $age = null;
79
80     echo "Name : ";
81     echo $name;
82     echo "\n";
83
84     echo "Age : ";
85     echo $age;
86     echo "\n";
```

# Mengecek Apakah Data NULL

- Kadang kita ingin tahu apakah sebuah data berniali null atau tidak
- Untuk mengecek apakah sebuah data bernilai null, kita bisa menggunakan function `is_null($variable)`

# Kode

```
37  
38     $kosong = NULL;  
39     $isi = 12;  
40     var_dump(is_null($kosong));  
41     var_dump(is_null($isi));  
42  
43
```

# Menghapus dan Mengecek Variabel

- Selain mengubah menjadi NULL, di PHP juga kita bisa menghapus sebuah variabel, caranya dengan menggunakan function `unset($variabel)`
- Namun hati-hati, ketika kita menghapus variabel, kita tidak bisa lagi mengakses variabel tersebut, bahkan function `is_null($variabel)` pun akan menjadi error jika mengakses variabel tersebut
- Lalu agar lebih aman, kita bisa menggunakan function `isset($variabel)` untuk mengecek apakah sebuah variabel ada dan nilainya tidak NULL

# Kode

```
$contoh = "Amin";  
unset($contoh);  
  
echo $contoh; //undefined
```

```
08  
09 $age = 15;  
10 $name = Null;  
11 $gender = "Male";  
12 unset($gender);  
13  
14 var_dump(isset($age));  
15 var_dump(isset($name));  
16 var_dump(isset($gender));  
17
```