



# Variabel Dasar , Tipe Data Membuat DataBase

---

Oleh: Taryana Suryana M.Kom  
Ilmu Pemerintahan Unikom 2022

[taryanarx@email.unikom.ac.id](mailto:taryanarx@email.unikom.ac.id)

[taryanarx@gmail.com](mailto:taryanarx@gmail.com)

[Line/Telegram: 081221480577](https://t.me/081221480577)

## 2.1 Tipe data dan Variabel

---

Pemrograman PHP mendukung beberapa tipe data, sebagai berikut:

- integer
- double
- string
- array
- object

Tipe variabel ini tidak perlu ditentukan oleh programmer, melainkan ditentukan pada saat runtime oleh PHP dengan kebergantungan pada konteks pemakaian variabel tersebut.

### 2.1.1 Menginisialisasi Variabel

---

Untuk menginisialisasi suatu variabel dalam PHP, secara sederhana berikan saja suatu nilai padanya. Tetapi untuk tipe seperti array dan objek membutuhkan mekanisme yang berbeda.

```
$nama = "Fahra Ragita";
```

```
$umur = 6;
```

```
$berat = 20.5;
```

## 2.1.2. Menginisialisasi Array

---

Array dapat diinisialisasi dengan dua cara: dengan pemberian nilai langsung, dan menggunakan konstruksi `array()`.

Untuk memberikan nilai secara langsung pada suatu array, secara sederhana adalah memberikan nilai ke variabel array dengan subscript kosong. Nilai tersebut akan ditambahkan sebagai elemen terakhir dari array.

```
$namasiswa[] = "Ilham";           // $namasiswa[0] = "Ilham"
```

```
$namasiswa[] = "Melati";         // $namasiswa[1] = "Melati"
```

Sesuatu yang perlu diingat adalah elemen array dimulai dari index 0 (bukan 1)

## 2.1.3. Menginisialisasi Objek

---

Untuk menginisialisasi suatu object, anda dapat menggunakan perintah `new`. Perintah ini digunakan untuk menginisialisasi suatu object kepada suatu variabel.

```
<?php
// buat class laptop
class laptop {

    // buat property untuk class laptop
    var $pemilik;
    var $merk;
    var $ukuran_layar;

    // buat method untuk class laptop
    function hidupkan_laptop() {
        return "Hidupkan Laptop";
    }
    function matikan_laptop() {
        return "Matikan Laptop";
    }
}

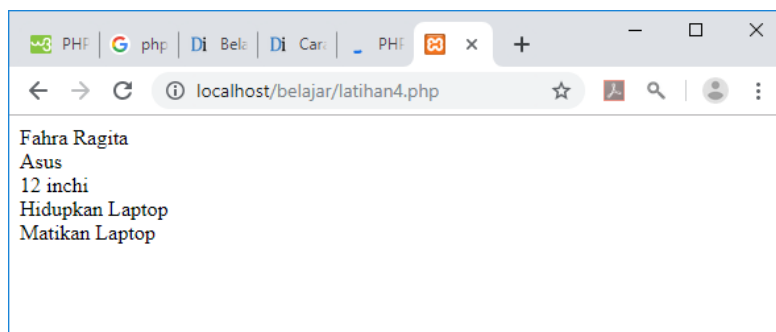
// buat objek dari class laptop (instansiasi)
$laptop_baru = new laptop();

// set property
$laptop_baru->pemilik="Fahra Ragita";
$laptop_baru->merk="Asus";
$laptop_baru->ukuran_layar="12 inchi";

// tampilkan property
echo $laptop_baru->pemilik;
```

```
echo "<br />";  
echo $laptop_baru->merk;  
echo "<br />";  
echo $laptop_baru->ukuran_layar;  
echo "<br />";  
  
// tampilkan method  
echo $laptop_baru->hidupkan_laptop();  
echo "<br />";  
echo $laptop_baru->matikan_laptop();  
?>
```

Kode Program: latihan4.php



## 2.1.4. Jangkauan dari variabel

Jangkauan dari suatu variabel adalah tergantung pada dimana variabel tersebut didefinisikan. Pada umumnya semua variabel PHP hanya memiliki suatu jangkauan. Setiap variabel yang digunakan dalam suatu fungsi, maka secara default jangkauannya adalah lokal. Sebagai contoh:

```
$awal = 1; /* jangkauan global */  
function Coba () {  
    echo $awal; /* mengacu pada variabel jangkauan lokal */  
}  
Coba ();
```

Script diatas tidak mengeluarkan apapun, karena \$awal dalam fungsi Coba mengacu pada \$awal lokal yang tidak memiliki nilai apapun. Hal ini berbeda dengan bahasa C dimana setiap variabel global akan berlaku bagi semua function, kecuali didefinisikan sebagai local. Dalam PHP, variabel global harus dideklarasikan sebagai global didalam suatu fungsi jika mereka akan digunakan dalam fungsi tersebut, contoh :

```
$panjang = 10;  
$lebar = 5;  
function Hitungluas () {  
    global $panjang, $lebar, $luas;  
    $luas = $panjang * $lebar;
```

```
}  
Hitungluas ();  
echo $luas;
```

Script diatas akan menghasilkan keluaran "50". Dengan mendeklarasikan \$panjang dan \$lebar sebagai global didalam fungsi.

Cara kedua untuk mengakses variabel sebagai acuan global adalah dengan menggunakan definisi \$GLOBAL array pada PHP. Contoh sebelumnya dapat juga ditulis menjadi:

```
echo "<hr>Cara Kedua<br>";  
$panjang= 5;  
$lebar= 3;  
function Hitungluas2() {  
    $GLOBALS["luas"]= $GLOBALS["panjang"] * $GLOBALS["lebar"];  
}  
Hitungluas2();  
echo $luas;
```

\$GLOBALS array adalah suatu asosiasi array dengan key adalah nama dari variabel global.

Hal lain yang penting dari jangkauan adalah variabel static. Suatu variabel hanya ada pada jangkauan lokal fungsi, dan nilainya akan tetap dipertahankan:

Contoh berikut menggunakan variabel static.

```
<?php  
function Coba () {  
    static $a = 0;  
    echo $a;  
    $a++;  
}  
Coba();  
?>
```

Sekarang, setiap kali fungsi Coba(), dipanggil maka nilai \$a adalah nilai \$a pada pemanggilan sebelumnya.

## 2.1.5. Variabel variabel

---

Kadang-kadang adalah lebih nyaman menggunakan variable-variabel; Yang mana adalah nama variabel yang dapat digunakan secara dinamis. Secara normalnya variabel dibuat dengan :

```
$a = "hello";
```

Suatu variabel variabel akan menggunakan nilai dari suatu variabel menjadi nama variabel, Contoh:



```
$a = "hello";
```

```
$$a = "world";
```

Pada contoh diatas akan terbentuk dua variabel yaitu \$a dan \$hello. Dimana \$a berisi "hello" dan \$hello berisi "world". Selanjutnya perintah berikut ini:

```
echo "$a ${$a}";
```

atau

```
echo "$a $hello"
```

Akan menghasilkan keluaran : hello world

## 2.1.6. Penentuan tipe variabel

---

PHP tidak membutuhkan deklarasi variabel secara eksplisit; tipe variabel ditentukan berdasarkan konteks pemakaiannya pada saat runtime. Dengan kata lain; jika anda memberi nilai string kesuatu variabel var, var menjadi suatu variabel tipe string. Jika anda memberi nilai integer ke var, maka otomatis berubah menjadi tipe integer.

Suatu contoh dari otomatisasi konversi tipe pada PHP adalah operator penjumlahan '+'. Jika salah satu operandnya adalah tipe double, maka semua operand lainnya dievaluasi sebagai double dan hasilnya adalah double.

```
$coba = "0";           // $coba adalah string (ASCII 48)
$coba++;             // $coba adalah string "1" (ASCII 49)
$coba += 1;         // $coba sekarang adalah integer (2)
$coba = $coba + 1.3; // $coba sekarang adalah double (3.3)
$coba = 5 + "10 kotak"; // $coba adalah integer (15)
$coba = 5 + "10 kotak kecil"; // $coba adalah integer (15)
```

## 2.1.7. Mengetahui tipe suatu variabel

---

Karena PHP menentukan tipe variabel sesuai dengan konteks pemakaiannya, maka anda dapat menggunakan fungsi berikut untuk memeriksa tipe pada suatu variabel `gettype()`, `is_long()`, `is_double()`, `is_string()`, `is_array()`, dan `is_object()`.

## 2.1.8. Tipe casting

---

Tipe casting dalam PHP bekerja seperti dalam C: nama dari tipe yang diinginkan ditulis didalam kurung sebelum variabel yang akan di cast.

```
$coba = 10;           // $coba adalah suatu integer
```

```
$bar = (double) $coba; // $bar adalah suatu double
```

## 2.1.9. Cast yang diperbolehkan:

---

(int), (integer) - cast ke integer  
(real), (double), (float) - cast ke double  
(string) - cast ke string  
(array) - cast ke array  
(object) - cast ke object

## 2.1.10. Konversi String

---

Ketika suatu string dievaluasikan sebagai suatu nilai numerik, nilai hasil dan tipe ditentukan sebagai berikut.

String akan dievaluasikan sebagai suatu double jika mengandung salah satu karakter '.', 'e' atau 'E'. Jika tidak akan dievaluasikan sebagai suatu string.

Jika string dimulai dengan data numerik yang sah, maka nilai tersebut akan digunakan, jika tidak akan memiliki nilai 0 (nol).

```
$coba = 1 + "10.5";           // $coba adalah double (11.5)
$coba = 1 + "-1.3e3";        // $coba adalah double (-1299)
$coba = 1 + "bob-1.3e3";    // $coba adalah integer (1)
$coba = 1 + "bob3";         // $coba adalah integer (1)
$coba = 1 + "10 Small Pigs"; // $coba adalah integer (11)
$coba = 1 + "10 Little Piggies"; // $coba adalah integer (11);
$coba = "10.0 pigs " + 1;    // $coba adalah int (11)
$coba = "10.0 pigs " + 1.0;  // $coba adalah double (11)
```

Tipe dari variabel tergantung pada ekspresi kedua, jika ekspresi pertamanya adalah string

## 2.1.11. Menangani variabel Form, Cookies dan Environment

---

Pada pemrograman CGI, program kita akan berinteraksi dengan variabel-variabel dari luar yang dikirim melalui form baik dengan metode GET maupun metode POST.

Ketika suatu form dikirim ke suatu PHP script, semua variabel dari form secara otomatis dapat diproses oleh script PHP sebagaimana variabel biasanya. Sebagai contoh, perhatikan form berikut ini:

```
<form action="proses.php" method="post">
  Nama: <input type="text" name="nama"><br>
  <input type="submit" value="Kirim">
</form>
```



Nama:

Ketika form disubmit, maka PHP akan membuat variabel \$nama, yang mana mengandung apa yang diketikkan pada field Nama: di form tersebut.

PHP juga mendukung variabel array dalam konteks form, tetapi dibatasi hanya 1 dimensi, Contoh:

```
<form action="array.php" method="post">
  Nama: <input name="personal[nama]"><br>
  Email: <input name="personal[email]"><br>
  Jurusan: <br>
  <select name="jurusan[]">
    <option value="Informatika">Informatika
    <option value="Akuntansi">Akuntansi
  </select>
  <input type="submit">
</form>
```

Top of Form

Nama :

Email :

Beer:

Bottom of Form

## 2.1.12. Variabel pada IMAGE SUBMIT

Ketika mengirim suatu form, juga dimungkinkan untuk menggunakan suatu gambar sebagai pengganti tombol submit dengan tag HTML berikut ini:

```
<input type="image" src="image.gif" name="sub">
```

Ketika pemakai melakukan klik pada gambar tersebut, maka form akan dikirim ke CGI dengan dua variabel tambahan, yaitu sub\_x dan sub\_y. Kedua variabel ini adalah koordinat dimana klik dilakukan pada gambar.

## 2.1.13. HTTP Cookies

---

PHP secara transparan mendukung HTTP cookies. Cookies adalah suatu mekanisme penyimpanan data secara remote pada browser klien. Hal ini dapat digunakan untuk mengidentifikasi pemakai pada kunjungan berikutnya.

Anda dapat menggunakan fungsi `SetCookie()`. Cookie adalah bagian dari HTTP header, jadi fungsi `SetCookie` harus dipanggil sebelum output lainnya dikirim ke browser. Hal ini adalah sama batasannya untuk fungsi `Header()`. Semua cookies yang dikirim pada anda dari klien akan secara otomatis di ubah kedalam suatu variabel PHP seperti method GET dan POST data.

Jika anda ingin memberikan banyak nilai pada suatu cookie tunggal, tambahkan saja `[]` pada nama cookie.

Sebagai contoh:

```
SetCookie ("MyCookie[]", "Testing", time()+3600);
```

Catatan bahwa suatu cookie akan menimpa cookie sebelumnya yang memiliki nama yang sama dalam browser anda, kecuali path atau domainnya berbeda. Jadi untuk suatu aplikasi shopping cart anda perlu menyimpan suatu counter dan mengirimnya bersamaan.

Contoh.

```
$Count++;
```

```
SetCookie ("Count", $Count, time()+3600);
```

```
SetCookie ("Cart[$Count]", $item, time()+3600);
```

## 2.1.14. Variabel Environment

---

PHP secara otomatis membuat variabel lingkungan normalnya seperti variabel PHP.

```
echo $HOME; /* Shows the HOME environment variable, if set. */
```

Sejak informasi datang dengan GET, POST dan mekanisme Cookies secara otomatis menjadi variabel PHP, adalah lebih baik membaca variabel langsung dari lingkungan untuk mendapatkan versi yang sebenarnya. Fungsi `getenv()` dapat digunakan untuk melakukan hal ini. Anda dapat juga dapat membentuk suatu variabel lingkungan dengan fungsi `putenv()`.

## 2.1.15. Konstruksi Bahasa PHP

---

Suatu PHP script terdiri dari sejumlah perintah yang berurutan. Suatu perintah dapat berupa pemberi nilai, pemanggilan terhadap fungsi, perulangan, perintah kondisi dan baris kosong. Setiap perintah diakhiri dengan sebuah titik koma (;). Dan beberapa perintah dapat dikelompokkan menjadi satu dalam kurung kurawal { dan }.

Catatan : Setiap perintah diakhir dengan sebuah titik koma (;).

Konstanta

PHP memiliki sejumlah konstanta yang telah didefinisikan, dan anda diberi kesempatan untuk membuat konstanta sesuai dengan kebutuhan anda.



Konstanta yang telah didefinisikan adalah `__FILE__` (nama file yang sedang diproses) `__LINE__` (nomor baris dari file yang sedang diproses)

Contoh :

```
<?php
function report_error($file, $line, $message) {
    echo "Suatu kesalahan terjadi pada file $file baris $line: $message.";
}
report_error(__FILE__, __LINE__, "Telah terjadi suatu kesalahan!");
?>
```

Anda dapat mendefinisikan konstanta ciptaan dengan menggunakan fungsi `define()`.

Contoh :

```
<?php
define("CONSTANT", "Hello world.");
echo CONSTANT; // akan mencetak "Hello world."
?>
```

## 2.1.16. Ekspresi

---

Hampir semua yang anda tulis dalam PHP script adalah ekspresi. Definisi yang paling mudah dari ekspresi adalah "segala sesuatu yang memiliki nilai".

Contohnya yang paling sederhana dari ekspresi adalah konstanta dan variabel. Ketika anda mengetik `"$a = 5"`, anda telah memberikan nilai '5' ke `$a`.

Contoh yang lebih komplek untuk suatu ekspresi adalah fungsi. Untuk jelasnya perhatikan contoh berikut:

```
function hello () {
    return 5;
}
```

maka kalau anda menulis `$c = hello()` adalah sama dengan memberikan nilai 5 ke variabel `$c`, karena fungsi `hello` mengembalikan 5. Diatas adalah contoh fungsi yang sederhana.

PHP mendukung tiga tipe nilai skalar yaitu : nilai integer, floating point dan string. (nilai skalar adalah nilai yang tidak dapat dibagi menjadi bagian yang lebih kecil, seperti array). PHP juga mendukung dua tipe nilai bukan skalar yaitu array dan objek.

## 2.2. Database

---

Basis data atau Database dapat didefinisikan atau diartikan sebagai kumpulan data yang disimpan secara sistematis di dalam komputer yang dapat diolah atau dimanipulasi menggunakan perangkat lunak (software) program atau aplikasi untuk menghasilkan

informasi. Pendefinisian basis data meliputi spesifikasi berupa tipe data, struktur data dan juga batasan-batasan pada data yang kemudian disimpan.

Basis data merupakan aspek yang sangat penting dalam sistem informasi karena berfungsi sebagai gudang penyimpanan data untuk diolah lebih lanjut. Basis data menjadi penting karena dapat mengorganisasi data, menghindari duplikasi data, menghindari hubungan antar data yang tidak jelas dan juga update yang rumit.

## 2.2.1. Membuat Database

Untuk membuat database dapat dilakukan dengan menggunakan phpMyAdmin, seperti yang pernah dibahas di Bab 1 pada awal perkuliahan.



Gambar 2.1. Membuat Database HI pada phpMyAdmin










Adapun langkahnya adalah sebagai berikut”

1. Click New Database, pada kotak Create database, tuliskan nama database yang ingin dibuat, pada contoh dimateri ini nama database yang akan dibuat adalah: **HI** kemudian click tombol Create
2. Jika database HI sudah ada, lanjutkan dengan membuat tabel

Setelah database berhasil dibuat, langkah selanjutnya adalah membuat tabel, misalnya anda memiliki data seperti berikut:

**Tabel:BARANG**

Extra options

				KODE	NAMABARANG	HARGA	JUMLAH
<input type="checkbox"/>	 Edit	 Copy	 Delete	B001	HP	3000000	5
<input type="checkbox"/>	 Edit	 Copy	 Delete	B002	LAPTOP	7000000	10
<input type="checkbox"/>	 Edit	 Copy	 Delete	B003	KOMPUTER	5000000	2

Pada tabel ini, ada dua tipe data yang akan digunakan, yaitu data yang bertipe Varchar untuk Data Text atau string dan Data bertipe Int untuk data angka atau numerik

## 2.2.2. Menampilkan Data

Pada bagian ini akan dibahas mengenai bagaimana membuat program dengan menggunakan bahasa pemrograman PHP untuk menampilkan data yang ada dalam database MySQL

### 2.2.2.1. Membuat File Koneksi dari PHP ke MYSQL

Pada bagian ini akan dibuat bagaimana cara menghubungkan antara program yang dibuat dengan PHP agar dapat terkoneksi dengan database dan tabel yang ada dalam MYSQL

NamaFile: **Koneksi.php**

```
<?php
$conn=mysqli_connect("localhost","root","","AK");
?>
```

\*Jangan lupa selama belajar database ini semua file simpan dalam folder yang sama yaitu:  
di folder **xampp/htdocs/AK**



### 2.2.2.2. Kode Program Menampilkan Data

Kemudian buat file berikut untuk menampilkan data yang ada dalam tabel, misalnya akan ditampilkan data yang ada dalam tabel:BARANG

NamaFile:tampilbarang.php

```
<html>
<center>
<h1>MENAMPILKAN DATA BARANG</h1>
<hr>
<table border=1>
<tr><th>NO<th>KODE<th>NAMABARANG<th>HARGA<th>JUMLAH
<?php
require ("koneksi.php");
$sql="select * from barang";
$hasil=mysqli_query($conn,$sql);
$row=mysqli_fetch_row($hasil);
$n=1;
do
{
list($kode,$namabarang,$harga,$jumlah)=$row;
echo "<tr><td>$n<td>$kode<td>$namabarang<td>$harga<td>$jumlah";
$n++;
}
while($row=mysqli_fetch_row($hasil));
?>
</table>
<hr>
Komputer Aplikasi E-Commerce HI UNIKOM
```

## 2.3. Menampilkan Hasil Via Browser

---

Setelah kedua program tadi dibuat dan disimpan dalam folder yang sama, kemudian adalah bagaimana cara menampilkannya ke layar komputer:

Untuk menampilkan hasilnya dapat dilakukan dengan cara seperti biasa menampilkan program php:

1. Jalankan Browser
2. Pada Address Bar, ketikkan:localhost/AK
3. Click File tampilbarang.php

Selanjutnya jika berhasil maka akan ditampilkan jendela berikut:



Gambar 2.5. Menampilkan data Barang

## TUGAS 2

Dengan mengikuti Contoh seperti sebelumnya buatlah Tabel dan Program Komputer Untuk menampilkan Data Pegawai Berikut:

SELECT \* FROM pegawai

Profiling [ Edit Inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

	NOPEG	NAMA	BAGIAN	MASUK	ANAK	PENDIDIKAN
<input type="checkbox"/>	P0001	SANTI NURYATI	KEUANGAN	2010	2	SARJANA
<input type="checkbox"/>	P0002	HELGA SALAM	HRD	2005	1	MAGISTER
<input type="checkbox"/>	P0003	REED JULIAN	PRODUKSI	2020	0	SARJANA
<input type="checkbox"/>	P0004	RIDWAN NULLOH	IT	2010	2	DIPLOMA
<input type="checkbox"/>	P0005	DONNY REZA	IT	2002	2	SARJANA
<input type="checkbox"/>	P0006	TIARA	KEUANGAN	2020	1	SMA

Program Menampilkan Data Pegawai



Instalasi xampp Untuk Belajar | localhost / 127.0.0.1 / ip / pegawai | localhost/ip/tampilpegawai.php

localhost/ip/tampilpegawai.php

### MENAMPILKAN DATA PEGAWAI

NO	NOPEG	NAMA	BAGIAN	MASUK	ANAK	PENDIDIKAN
1	P0001	SANTI NURYATI	KEUANGAN	2010	2	SARJANA
2	P0002	HELGA SALAM	HRD	2005	1	MAGISTER
3	P0003	REED JULIAN	PRODUKSI	2020	0	SARJANA
4	P0004	RIDWAN NULLOH	IT	2010	2	DIPLOMA
5	P0005	DONNY REZA	IT	2002	2	SARJANA
6	P0006	TIARA	KEUANGAN	2020	1	SMA

Komputer Aplikasi IT-5 IP UNIKOM

Kerjakan Tugas2, kirim ke <https://kuliahonline.unikom.ac.id> dalam format PDF Coding Program dan Screenshot hasil akhir, paling Lambat hari Minggu Sore

ooOOoo

## PUSTAKA

1. Suryana, Taryana (2022) [Materi 7 Hapus Data Barang](#). [Teaching Resource]
2. Suryana, Taryana (2022) [6.edit Data Barang Pada Database Barang](#). [Teaching Resource]
3. Suryana, Taryana (2022) [6.edit Data Ktp](#). [Teaching Resource]
- 4.
5. Suryana, Taryana (2022) [Materi 5 Pencarian Data Barang Dan Data Pegawai](#). [Teaching Resource]
6. Suryana, Taryana (2022) [Membuat Form Memasukan Data Pegawai](#). [Teaching Resource]
7. Suryana, Taryana (2022) [Materi 3 Input Dan Output Pada Database Barang](#). [Teaching Resource]
8. Suryana, Taryana (2022) [Membuat Database](#). [Teaching Resource]
9. Suryana, Taryana (2022) [Instalasi Xampp Untuk Pemrograman E-commerce 2022](#). [Teaching Resource]
10. Suryana, Taryana (2022) [Pengenalan E-commerce](#). [Teaching Resource]
11. Suryana, Taryana (2022) [Membuat Program Untuk Edit Data Akta Perkawinan](#). [Teaching Resource]
12. Suryana, Taryana (2022) [Membuat Program Untuk Perubahan Atau Edit Data](#). [Teaching Resource]
13. Suryana, Taryana (2022) [Menampilkan Relasi Penjualan Barang](#). [Teaching Resource]
14. Suryana, Taryana (2021) [Membuat Program Untuk Menangani Penjualan Barang](#). [Teaching Resource]
15. Suryana, Taryana (2021) [Membuat Toko Online](#). [Teaching Resource]
16. Suryana, Taryana (2021) [Edit Data Pada Aplikasi Ijin Gangguan Ho](#). [Teaching Resource]
17. Suryana, Taryana (2021) [Pembuatan Aplikasi Ijin Gangguan Ho Dengan Php Dan Mysql](#). [Teaching Resource]
18. Suryana, Taryana (2021) [Fungsi Logika Dalam Pemrograman Php Dan Mysql](#). [Teaching Resource]
19. Suryana, Taryana (2021) [Input Dan Output Data Barang Dan Data Member](#). [Teaching Resource]
20. Suryana, Taryana (2021) [Instalasi Xampp Untuk Menjalankan Web Server Apache Database Mysql Dan Program Php](#). [Teaching Resource]
21. Suryana, Taryana (2021) [Membuat Formulir Pemasukan Data Dengan Html Dan Html5 Kelas If-wa1-ul2-2021](#). [Teaching Resource]
22. Suryana, Taryana (2021) [Berbagai Macam Operator Dalam Php Dan Mysql](#). [Teaching Resource]
23. Suryana, Taryana (2021) [Instalasi Xampp Untuk Pemrograman E-commerce](#). [Teaching Resource]
24. Suryana, Taryana (2021) [Pengenalan E-commerce](#). [Teaching Resource]
25. Suryana, Taryana (2021) [Membuat Database](#). [Teaching Resource]
26. Suryana, Taryana; Koesheryatin (2014), [Aplikasi Internet Menggunakan HTML, CSS & Java Script](#), Elexmedia Komputindo, Jakarta
27. Suryana, Taryana (2021) [Pelatihan Pembuatan Website Dengan Menggunakan Html Dan Dreamweaver](#). [Teaching Resource]
28. Suryana, Taryana (2010) [Membuat Web Pribadi dan Bisnis dengan HTML](#)., Gava Media, Yogyakarta
29. <https://www.w3schools.com/> September, 2021
30. Suryana, Taryana; Sarwono, Jonathan. [E-Commece Menggunakan PHP & MySQL](#), Graha Ilmu, Yogyakarta
31. Suryana, Taryana (2021) [Materi 1. Pengenalan Html Web Browser Dan Text Editor](#). [Teaching Resource]
32. Suryana, Taryana, [e-COMMERCE MENGGUNAKAN PHP DAN MYSQL](#), Graha Ilmu, Yogyakarta
33. <http://www.grahailmu.co.id/previewpdf/978-979-756-182-6-231.pdf>