

**LAPORAN PRAKTIKUM
PERANCANGAN DAN PEMROGRAMAN WEB**

**MODUL 9
(PHP)**



**Universitas
Telkom**

Oleh:

(Rengganis Tantri Pramudita) (2311104065)

**PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK
DIREKTORAT KAMPUS PURWOKERTO
UNIVERSITAS TELKOM
2025**

BAB I

PENDAHULUAN

1.1 Dasar Teori

Pada pemrograman web, PHP (PHP: Hypertext Preprocessor) merupakan bahasa pemrograman *server-side scripting* yang digunakan untuk membangun halaman web dinamis. PHP berjalan di sisi server dan menghasilkan output berupa halaman HTML yang kemudian dikirimkan ke browser. Untuk menjalankan PHP diperlukan web server seperti Apache dan lingkungan server lokal seperti XAMPP yang sudah menyediakan Apache, PHP, dan MySQL sekaligus. PHP memiliki berbagai fitur seperti variabel, konstanta, operator, struktur kontrol (if, switch), perulangan (for, while, do-while, foreach), fungsi, serta dukungan manipulasi array, yang semuanya digunakan untuk membangun aplikasi web secara dinamis.

1.2 Tujuan

- Mahasiswa mampu memahami konsep dan implementasi PHP pada web.
- Mahasiswa mampu memahami sintaks, elemen, dan fungsi pada PHP.

1.3 Manfaat

Pembelajaran mengenai PHP pada modul ini memberikan manfaat bagi mahasiswa dalam memahami cara kerja pemrograman web sisi server secara mendasar. Dengan mempelajari PHP, mahasiswa mampu membuat halaman web yang dinamis melalui penggunaan variabel, operator, kondisi, perulangan, fungsi, dan array. Selain itu, pemahaman terhadap instalasi dan penggunaan XAMPP mempermudah mahasiswa dalam menyiapkan lingkungan pengembangan web lengkap dengan Apache, PHP, dan MySQL. Pengetahuan ini menjadi dasar penting untuk mengembangkan aplikasi web yang lebih kompleks di masa depan, termasuk integrasi dengan database dan pembuatan sistem berbasis web yang fungsional.

BAB II

HASIL PRAKTIKUM

2.1 GUIDED

- Pengenalan PHP



A screenshot of a terminal window. At the top, there are three colored dots (red, yellow, green). Below them, the following PHP code is displayed:

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

Penjelasan:

Kode yang kamu buat menggunakan tag `<?php ... ?>` yang menandakan bahwa bagian tersebut adalah script PHP. Di dalamnya terdapat berbagai contoh fitur dasar PHP seperti menampilkan teks dengan echo, komentar satu baris maupun multi-line, serta struktur kode lain yang menunjukkan dasar cara kerja PHP sebagai bahasa pemrograman server-side.

Output:



- Variabel

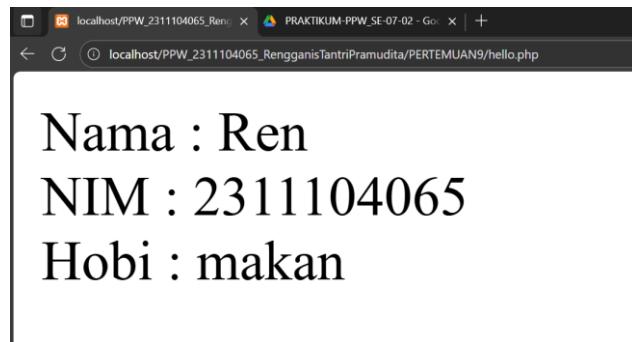


```
1 $nama = "Ren";
2 $nim = "2311104065";
3 $hobi = "makan";
4
5 echo "Nama : " . $nama . "<br>";
6 echo "NIM : " . $nim . "<br>";
7 echo "Hobi : " . $hobi . "<br>";
```

Penjelasan:

Pada bagian variabel, kode mendefinisikan beberapa variabel seperti \$nama, \$nim, dan \$hobi, lalu menampilkannya menggunakan echo. Variabel di PHP selalu diawali dengan tanda \$ dan nilainya dapat berupa teks ataupun angka tanpa perlu deklarasi tipe data terlebih dahulu.

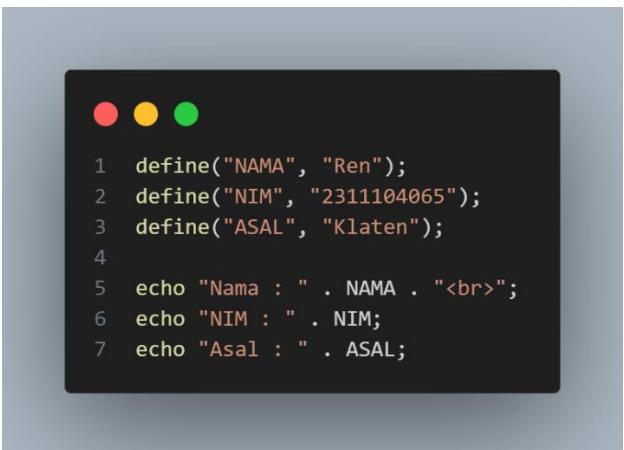
Output:



localhost/PPW_2311104065_Ren
PRAKTIKUM-PPW_SE-07-02 - Google Chrome

Nama : Ren
NIM : 2311104065
Hobi : makan

- Konstanta

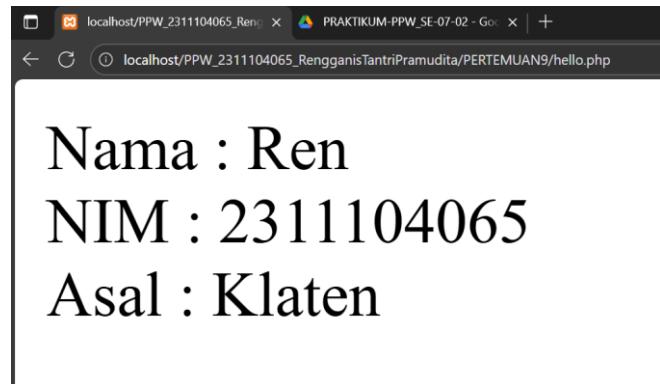


```
1 define("NAMA", "Ren");
2 define("NIM", "2311104065");
3 define("ASAL", "Klaten");
4
5 echo "Nama : " . NAMA . "<br>";
6 echo "NIM : " . NIM;
7 echo "Asal : " . ASAL;
```

Penjelasan:

Pada bagian konstanta, kode menggunakan define() untuk membuat nilai tetap seperti NAMA, NIM, dan ASAL. Konstanta berbeda dari variabel karena nilainya tidak dapat diubah setelah didefinisikan, dan saat dipanggil tidak menggunakan tanda \$.

Output:



```
Nama : Ren  
NIM : 2311104065  
Asal : Klaten
```

- Struktur Kondisi

If - then

```
● ● ●  
1 $nilai = 45;  
2  
3 if ($nilai > 50) {  
4     echo "Nilai Anda adalah " . $nilai . ". Selamat, Anda lulus";  
5 } else {  
6     echo "Nilai Anda adalah " . $nilai . ". Maaf, Anda tidak lulus";  
7 }
```

Output:



```
Nilai Anda adalah 45. Maaf, Anda tidak lulus
```

Switch case

```
● ● ●
1 $nilai = 60;
2
3 switch (true) {
4     case ($nilai > 50 && $nilai <= 60):
5         echo "Nilai Anda adalah $nilai. Indeks nilai Anda C";
6         break;
7
8     case ($nilai > 60 && $nilai <= 70):
9         echo "Nilai Anda adalah $nilai. Indeks nilai Anda BC";
10        break;
11
12    case ($nilai > 70 && $nilai <= 75):
13        echo "Nilai Anda adalah $nilai. Indeks nilai Anda B";
14        break;
15
16    case ($nilai > 75 && $nilai <= 80):
17        echo "Nilai Anda adalah $nilai. Indeks nilai Anda AB";
18        break;
19
20    case ($nilai > 80 && $nilai <= 100):
21        echo "Nilai Anda adalah $nilai. Indeks nilai Anda A";
22        break;
23
24    default:
25        echo "Nilai Anda adalah $nilai. Maaf, Anda tidak lulus";
26        break;
27 }
28
```

Output:

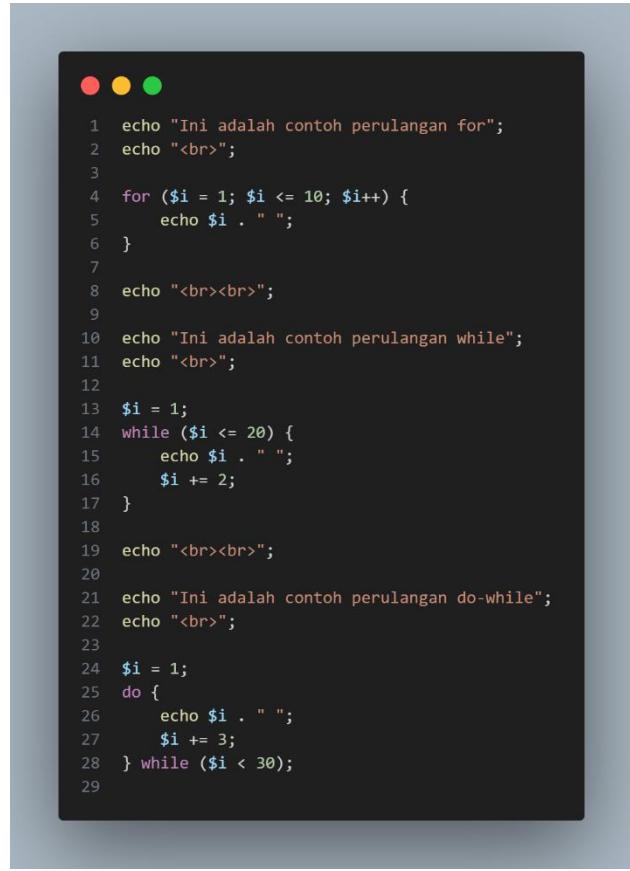


A screenshot of a web browser window. The address bar shows the URL `localhost/PPW_2311104065_RengganisTantriPramudita/PERTEMUAN9/hello.php`. The main content area of the browser displays the text "Nilai Anda adalah 60. Indeks nilai Anda C".

Penjelasan:

Struktur kondisi pada kode ditunjukkan melalui penggunaan if-else dan switch-case. Keduanya berfungsi untuk memeriksa suatu kondisi dan menjalankan blok kode tertentu. Pada contoh if-else, program mengecek apakah nilai lebih dari 50, sedangkan switch-case mengelompokkan nilai menjadi beberapa kategori indeks berdasarkan rentang nilai.

- Perulangan (Looping)

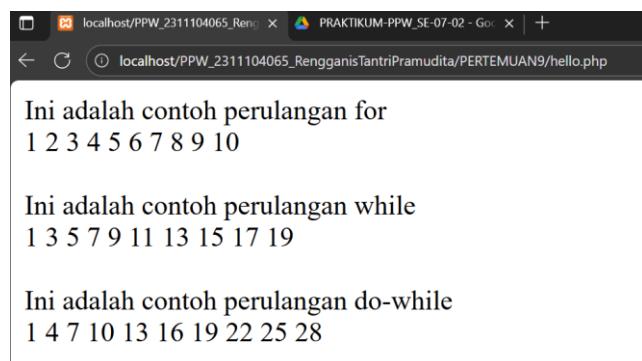


```
1 echo "Ini adalah contoh perulangan for";
2 echo "<br>";
3
4 for ($i = 1; $i <= 10; $i++) {
5     echo $i . " ";
6 }
7
8 echo "<br><br>";
9
10 echo "Ini adalah contoh perulangan while";
11 echo "<br>";
12
13 $i = 1;
14 while ($i <= 20) {
15     echo $i . " ";
16     $i += 2;
17 }
18
19 echo "<br><br>";
20
21 echo "Ini adalah contoh perulangan do-while";
22 echo "<br>";
23
24 $i = 1;
25 do {
26     echo $i . " ";
27     $i += 3;
28 } while ($i < 30);
29
```

Penjelasan:

Contoh looping dalam kode mencakup for, while, dan do-while. Perulangan digunakan untuk mengeksekusi blok kode secara berulang, seperti menampilkan angka secara berurutan. for digunakan saat jumlah perulangan jelas, while berjalan selama kondisi masih benar, dan do-while minimal menjalankan perulangan sekali sebelum mengecek kondisi.

Output:



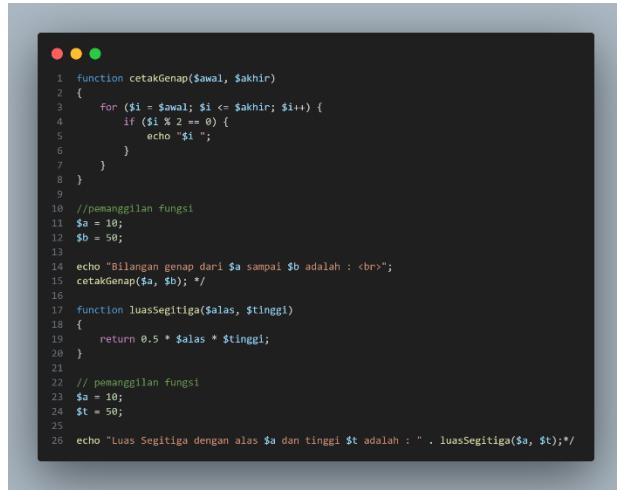
localhost/PPW_2311104065_RengganisTantriPramudita/PERTEMUAN9/hello.php

Ini adalah contoh perulangan for
1 2 3 4 5 6 7 8 9 10

Ini adalah contoh perulangan while
1 3 5 7 9 11 13 15 17 19

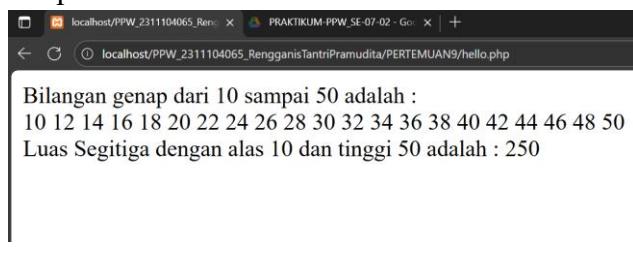
Ini adalah contoh perulangan do-while
1 4 7 10 13 16 19 22 25 28

- Function



```
1 function cetakGenap($awal, $akhir)
2 {
3     for ($i = $awal; $i <= $akhir; $i++) {
4         if ($i % 2 == 0) {
5             echo "$i ";
6         }
7     }
8 }
9
10 //pemanggilan fungsi
11 $a = 10;
12 $b = 50;
13
14 echo "Bilangan genap dari $a sampai $b adalah : <br>";
15 cetakGenap($a, $b); */
16
17 function luasSegitiga($ alas, $tinggi)
18 {
19     return 0.5 * $alas * $tinggi;
20 }
21
22 // pemanggilan fungsi
23 $a = 10;
24 $t = 50;
25
26 echo "Luas Segitiga dengan alas $a dan tinggi $t adalah : " . luasSegitiga($a, $t);*/
```

Output:



```
Bilangan genap dari 10 sampai 50 adalah :
10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50
Luas Segitiga dengan alas 10 dan tinggi 50 adalah : 250
```

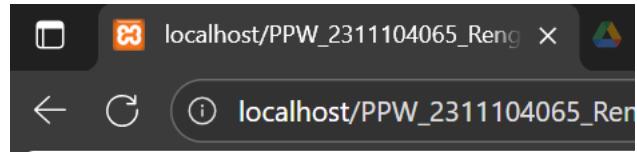
Penjelasan:

Bagian function menunjukkan cara membuat fungsi seperti cetakGenap() dan luasSegitiga(). Fungsi digunakan untuk mengeksekusi tugas tertentu agar kode lebih terstruktur dan dapat digunakan kembali. Ada fungsi yang hanya mencetak hasil dan ada fungsi yang mengembalikan nilai dengan return.

- Array

```
1 $arrKendaraan = ["Mobil", "Pesawat", "Kereta Api", "Kapal Laut", "Sepeda"];
2
3 echo $arrKendaraan[0] . "<br>"; // Mobil
4 echo $arrKendaraan[2] . "<br>"; // Kereta Api
5 echo $arrKendaraan[4] . "<br>"; // Kereta Api
6
7 $arrKota = [];
8 $arrKota[] = "Jakarta";
9 $arrKota[] = "Medan";
10 $arrKota[] = "Bandung";
11 $arrKota[] = "Malang";
12 $arrKota[] = "Sulawesi";
13
14 echo $arrKota[1] . "<br>"; // Medan
15 echo $arrKota[2] . "<br>"; // Bandung
16 echo $arrKota[4] . "<br>"; // Sulawesi */
17
18
19 $arrAlamat = [
20     "Rona" => "Banjarmasin",
21     "Dhiva" => "Bandung",
22     "Ilham" => "Medan",
23     "Oku" => "Hongkong",
24 ];
25
26 echo $arrAlamat["Dhiva"] . "<br>"; // Bandung
27 echo $arrAlamat["Oku"] . "<br>"; // Hongkong
28
29 $arrNim = [];
30 $arrNim["Rona"]    = "11011112";
31 $arrNim["Dhiva"]   = "11011101";
32 $arrNim["Ilham"]   = "11011309";
33 $arrNim["Oku"]     = "11014765";
34 $arrNim["Fadlan"]  = "11011113";
35
36 echo $arrNim["Ilham"] . "<br>";
```

Output:



Mobil
Kereta Api
Sepeda
Medan
Bandung
Sulawesi
Bandung
Hongkong
11011309
11011113

Penjelasan:

Struktur array pada kode berfungsi menyimpan banyak nilai dalam satu variabel. Kamu menggunakan array numerik seperti \$arrKendaraan dan \$arrKota, serta array asosiatif seperti \$arrAlamat dan \$arrNim yang menggunakan key berupa string. Semua array tersebut bisa diakses menggunakan indeks atau key yang sesuai.

2.2 UNGUIDED

1. Buat program konversi suhu yang dapat mengkonversi:
 - Celcius ke Fahrenheit
 - Fahrenheit ke Celcius
 - Celcius ke Kelvin

Tampilkan menggunakan 2 desimal di belakang koma

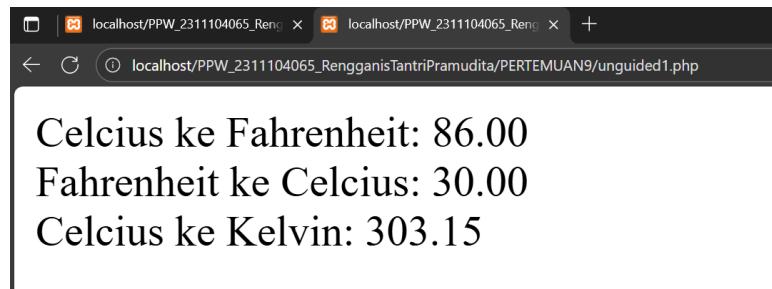
Jawab:

```
● ● ●
1 <?php
2 $celcius = 30;
3 $fahrenheit = 86;
4
5 // Celcius ke Fahrenheit
6 $cf = ($celcius * 9/5) + 32;
7
8 // Fahrenheit ke Celcius
9 $fc = ($fahrenheit - 32) * 5/9;
10
11 // Celcius ke Kelvin
12 $ck = $celcius + 273.15;
13
14 echo "Celcius ke Fahrenheit: " . number_format($cf, 2) . "<br>";
15 echo "Fahrenheit ke Celcius: " . number_format($fc, 2) . "<br>";
16 echo "Celcius ke Kelvin: " . number_format($ck, 2) . "<br>";
17 ?>
18
```

Penjelasan:

Program konversi suhu ini menggunakan variabel untuk menyimpan nilai Celcius dan Fahrenheit, kemudian melakukan perhitungan menggunakan operator aritmatika sesuai rumus konversi suhu. Konversi yang dilakukan meliputi Celcius ke Fahrenheit, Fahrenheit ke Celcius, dan Celcius ke Kelvin. Setiap hasil perhitungan ditampilkan menggunakan fungsi number_format() agar output memiliki dua angka di belakang koma, sesuai ketentuan. Program ini memanfaatkan konsep dasar PHP seperti variabel, operator, dan perintah echo untuk menampilkan hasil konversi ke layar.

Output:



```
Celcius ke Fahrenheit: 86.00
Fahrenheit ke Celcius: 30.00
Celcius ke Kelvin: 303.15
```

2. Kalkulator Diskon:

Input: Total belanja

- Diskon 10% jika belanja \geq Rp 100.000
- Diskon 20% jika belanja \geq Rp 500.000
- Diskon 30% jika belanja \geq Rp 1.000.000

Tampilkan: total belanja, diskon, dan total bayar

Jawab:



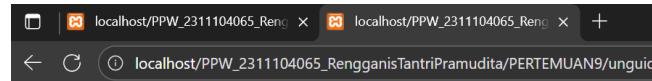
```
<?php
$total = 750000;
$diskon = 0;
if ($total >= 1000000) {
    $diskon = 0.30;
} elseif ($total >= 500000) {
    $diskon = 0.20;
} elseif ($total >= 100000) {
    $diskon = 0.10;
}
$jumlahDiskon = $total * $diskon;
$totalBayar = $total - $jumlahDiskon;
echo "Total Belanja: Rp " . number_format($total, 0, ',', '.') . "<br>";
echo "Diskon: Rp " . number_format($jumlahDiskon, 0, ',', '.') . "<br>";
echo "Total Bayar: Rp " . number_format($totalBayar, 0, ',', '.') . "<br>";
?>
```

Penjelasan:

Program kalkulator diskon ini menggunakan variabel untuk menyimpan total belanja, lalu menentukan besar diskon menggunakan struktur kondisi *if-elseif-else* sesuai ketentuan nominal belanja. Setelah diskon ditentukan, program menghitung jumlah potongan harga dan total bayar menggunakan operator aritmatika. Hasil akhirnya ditampilkan menggunakan echo, sehingga pengguna dapat melihat total belanja, nilai diskon, dan total bayar setelah potongan.

Program ini memanfaatkan konsep dasar PHP seperti variabel, operator, dan struktur kondisi.

Output:



Total Belanja: Rp 750.000

Diskon: Rp 150.000

Total Bayar: Rp 600.000

3. Manipulasi Array:

Buat array nilai mahasiswa: [75, 89, 65, 90, 85, 70, 98, 65, 69, 70, 12]

Tampilkan:

- Nilai tertinggi
- Nilai terendah
- Rata-rata nilai
- Jumlah mahasiswa yang lulus (≥ 70)

Urutkan nilai dari tertinggi ke terendah

Jawab:



```
1 <?php
2 $nilai = [75, 89, 65, 90, 85, 70, 98, 65, 69, 70, 12];
3
4 // Nilai tertinggi & terendah
5 $max = max($nilai);
6 $min = min($nilai);
7
8 // Rata-rata
9 $rata = array_sum($nilai) / count($nilai);
10
11 // Jumlah yang lulus
12 $lulus = 0;
13 foreach ($nilai as $n) {
14     if ($n >= 70) {
15         $lulus++;
16     }
17 }
18
19 // Urutkan nilai dari terbesar ke terkecil
20 rsort($nilai);
21
22 echo "Nilai Tertinggi: $max <br>";
23 echo "Nilai Terendah: $min <br>";
24 echo "Rata-rata Nilai: " . number_format($rata, 2) . "<br>";
25 echo "Jumlah Mahasiswa Lulus: $lulus <br>";
26
27 echo "Nilai Urut Tertinggi ke Terendah: <br>";
28 foreach ($nilai as $n) {
29     echo $n . " ";
30 }
31 ?>
32
```

Penjelasan:

Program manipulasi array ini menggunakan sebuah array berisi daftar nilai mahasiswa, kemudian memanfaatkan fungsi PHP untuk mengambil nilai tertinggi, nilai terendah, serta menghitung rata-rata menggunakan `array_sum()` dibagi jumlah elemen. Untuk menghitung jumlah mahasiswa yang lulus, digunakan perulangan `foreach` dan kondisi `if` untuk mengecek apakah nilai ≥ 70 . Nilai kemudian diurutkan dari yang tertinggi ke terendah menggunakan `rsort()`. Program ini menunjukkan penggunaan array, perulangan, kondisi, serta fungsi bawaan PHP dalam mengolah data.

Output:

```
Nilai Tertinggi: 98
Nilai Terendah: 12
Rata-rata Nilai: 71.64
Jumlah Mahasiswa Lulus: 7
Nilai Urut Tertinggi ke Terendah:
98 90 89 85 75 70 70 69 65 65 12
```

BAB III

KESIMPULAN & SARAN

3.1 Kesimpulan

Berdasarkan materi dan praktik yang dilakukan, dapat disimpulkan bahwa PHP merupakan bahasa pemrograman server-side yang mudah dipahami dan sangat membantu dalam pembuatan halaman web dinamis. Melalui latihan seperti konversi suhu, kalkulator diskon, serta manipulasi array, mahasiswa dapat memahami penggunaan variabel, operator, struktur kondisi, perulangan, fungsi, dan array secara langsung. Implementasi konsep dasar tersebut memungkinkan mahasiswa untuk mengolah data, mengambil keputusan berdasarkan kondisi tertentu, dan menghasilkan output yang sesuai kebutuhan. Praktikum ini memberikan dasar yang kuat untuk mempelajari pengembangan web lebih lanjut, termasuk integrasi dengan database dan pembuatan aplikasi web yang lebih kompleks.

3.2 Saran

Untuk meningkatkan pemahaman, mahasiswa disarankan untuk terus berlatih membuat berbagai variasi program menggunakan PHP agar lebih terbiasa dengan logika pemrograman. Selain itu, penting untuk mulai mencoba menggabungkan konsep-konsep dasar seperti fungsi, array, dan kondisi dalam satu program agar kemampuan penyusunan kode menjadi lebih terstruktur. Mahasiswa juga dapat mulai mengeksplorasi materi lanjutan seperti pengelolaan form dan koneksi database agar siap menghadapi pengembangan web tingkat berikutnya. Dengan konsistensi latihan, keterampilan pemrograman web akan berkembang secara signifikan.