

LAPORAN PRAKTIKUM PEMROGRAMAN WEB
“PHP (KONTROL DAN PERULANGAN)”



disusun oleh :

Vian Azis Tio Riwanto

E31191848

GOLONGAN C
PROGRAM STUDI MANAJEMEN INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI JEMBER
2020

BAB I

PENDAHULUAN

1.1 Latar Belakang

Ada banyak jenis bahasa pemrograman yang bisa Kita gunakan dalam pembuatan website. Salah satunya di Indonesia saat ini yang paling populer bahasa pemrograman web adalah menggunakan PHP. PHP sekarang ini menjadi dasar bagi para web developer sebelum belajar bahasa pemrograman tingkat lanjut, termasuk menggunakan framework PHP. Dalam pengembangan website, PHP menjadi salah satu bahasa pemrograman yang wajib dipelajari. Alasannya, bahasa pemrograman ini mampu untuk membuat website menjadi dinamis. Pada praktikum kali ini akan membahas pengertian dan cara mengimplementasikan kontrol/kondisional dan perulangan dalam PHP.

1.2 Rumusan masalah

- 1) Apa Pengertian PHP?
- 2) Apa Pengertian Pernyataan Kontrol Atau Kondisional Dalam PHP?
- 3) Apa Pengertian Pernyataan Looping Dalam PHP?

1.3 Tujuan

Tujuan dari praktikum ini di bentuk :

- 1) Mahasiswa memahami dan mampu mengimplementasikan tag php di html pada halaman web.
- 2) Mahasiswa mampu menggunakan proses kontrol dan perulangan pada PHP.

BAB II

TEORI

2.1 Pengertian PHP

PHP (PHP: Hypertext Preprocessor) adalah sebuah bahasa pemrograman server side scripting yang bersifat open source. Bahasa pemrograman ini banyak digunakan untuk pengembangan website.

Sebagai sebuah scripting language, PHP menjalankan instruksi pemrograman saat proses runtime. Hasil dari instruksi tentu akan berbeda tergantung data yang diproses. Inilah yang membuat PHP sering digunakan untuk membangun website yang dinamis seperti toko online.

PHP merupakan bahasa pemrograman server-side, maka script dari PHP nantinya akan diproses di server. Jenis server yang sering digunakan bersama dengan PHP antara lain Apache, Nginx, dan LiteSpeed.

Selain itu, PHP juga merupakan bahasa pemrograman yang bersifat open source. Pengguna bebas memodifikasi dan mengembangkan sesuai dengan kebutuhan mereka.

Meskipun awalnya hanya untuk membangun situs pribadi, perkembangan PHP cukup pesat. Saat ini, tak kurang dari 78% website di seluruh dunia menggunakan bahasa pemrograman yang diciptakan Rasmus Lerdorf di tahun 1995 ini. Bahkan platform besar seperti Facebook juga menggunakannya.

Sintaks Dasar PHP

Setiap bahasa pemrograman memiliki aturan coding sendiri. Begitu pula dengan PHP. Sintaks dasarnya dibuka dengan `<?php` dan ditutup dengan `?>` sebagai terlihat di contoh berikut:

```
<?php
    echo "Selamat datang";
?>
```

Inilah penjelasan kode tersebut:

- `<?php` ini adalah kode wajib untuk membuka program PHP.
- `Echo` adalah sebuah perintah untuk menampilkan teks.
- `"Selamat Datang";` teks yang hendak ditampilkan dan ditulis diantara tkita petik dan titik koma.
- `?>` adalah kode untuk mengakhiri PHP dan wajib digunakan saat digabung dengan bahasa pemrograman lain seperti HTML.

Sintaks PHP bersifat case sensitive. Jadi, penggunaan huruf besar atau kecil akan turut mempengaruhi output yang diberikan. Sebagai contoh :

```
<?php
    $alamat  = "Jember";
    echo $alamat;
?>
```

Kode di atas akan menghasilkan output: Jember di halaman website.

Namun, jika dituliskan seperti ini:

```
<?php
    $alamat  = "Jember";
    echo $Alamat;
?>
```

Kita akan mendapati tampilan error. Alasannya, adanya perbedaan antara variabel \$alamat dan \$Alamat.

Pada PHP, Kita juga bisa menuliskan komentar sebagai penjelasan dari kode yang ditulis. Komentar di PHP menggunakan // atau /* dan tidak akan dimunculkan sebagai output di browser. Contohnya sebagai berikut:

```
<?php
    // ini contoh penggunaan komentar yang ditulis satu baris

    /*
    Nah ini juga contoh komentar
    yang ditulis
    lebih dari satu baris
    */
?>
```

Semua kode PHP yang ditulis harus disimpan dengan file ekstensi .php.

2.2 Pengertian Pernyataan Kontrol Atau Kondisional Dalam PHP

Definisi

Struktur Kontrol Pecabangan merupakan suatu algoritma program memiliki suatu kondisi yang dimana kondisi tersebutlah yang akan menentukan perintah-perintah yang akan dijalankan oleh suatu program. Pernyataan kontrol atau kondisional berguna untuk mengecek suatu kondisi dan melakukan suatu kode jika kondisi tersebut benar atau salah.

Macam-Macam Percabangan Pada PHP

Untuk struktur kontrol percabangan pada PHP, terdapat beberapa macam. Ada if else, ada switch case, dan ada ternary.

- **Percabangan If**

Percabangan if adalah percabangan yang paling dasar. Tugasnya adalah memeriksa nilai boolean atau sebuah ekspresi logika. Bentuk dasarnya seperti berikut :

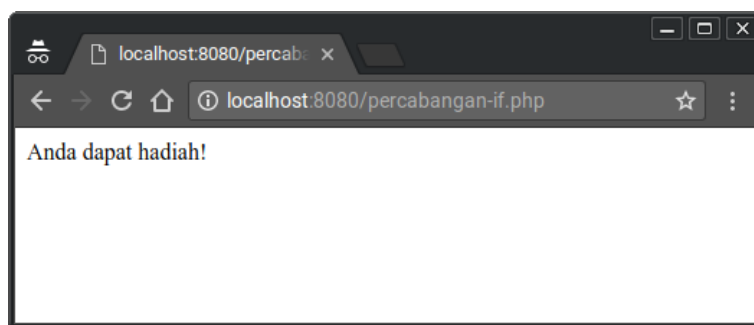
```
if (<kondisi>){  
    // eksekusi kode ini  
}
```

Jika kondisi benar, maka eksekusi kode yang ada di dalamnya. <kondisi> bisa kita isi dengan nilai boolean atau kita bisa buat pernyataan untuk menghasilkan nilai boolean.

Contoh:

```
$total_belanja = 150000;  
  
if($total_belanja > 100000){  
    echo "Anda dapat hadiah!";  
}
```

Hasil :



Kondisi atau pernyataan ini akan bernilai true dan false. Jika true (benar), maka kode yang ada di dalamnya akan dieksekusi. Namun, apabila false maka tidak akan mengeksekusinya.

- **Percabangan If/Else**

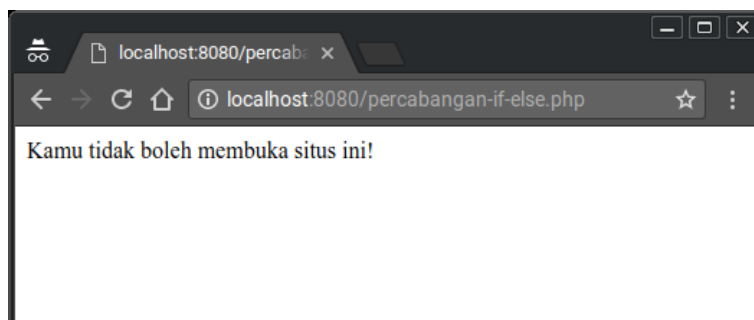
Percabangan If/Else memiliki dua pilihan. Jika <kondisi> bernilai false, maka blok else akan dikerjakan. Bentuk dasarnya seperti berikut :

```
if (kondisi){  
    kode program yang dijalankan jika memenuhi kondisi  
} else{  
    kode program yang dijalankan jika tidak memenuhi kondisi  
}
```

Contoh :

```
<?php  
$umur = 13;  
if ($umur < 18 ){  
    echo "Kamu tidak boleh membuka situs ini!";  
} else {  
    echo "Selamat datang di website kami!";  
}  
?>
```

Hasil :



Karena variabel \$umur = 13 , maka yang di eksekui adalah statemen if, namun jika variabel bernilai false pada kondisi if (\$umur < 18), maka else yang akan di eksekusi. Misal jika variabel \$umur = 19, maka akan muncul “Selamat datang di website kami!”.

- **Percabangan If/Elseif/Else**

Percabangan If/Elseif/Else memiliki lebih dari dua pilihan kondisi.

Bentuk dasarnya seperti berikut :

```
if (kondisi1){
    kode program yang dijalankan jika memenuhi kondisi 1
} else if (kondisi2){
    kode program yang dijalankan jika tidak memenuhi
    kondisi 1 dan memenuhi kondisi 2.
} else{
    kode program yang dijalankan jika tidak memenuhi
    semua kondisi
}
```

Contoh:

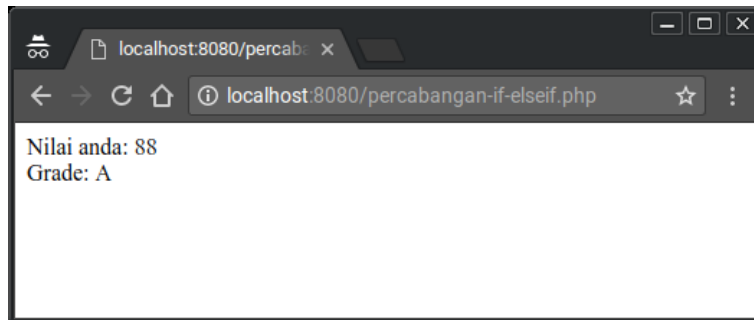
```
<?php
$nilai = 88;

if ($nilai > 90) {
    $grade = "A+";
} elseif($nilai > 80){
    $grade = "A";
} elseif($nilai > 70){
    $grade = "B+";
} elseif($nilai > 60){
    $grade = "B";
} elseif($nilai > 50){
    $grade = "C+";
} elseif($nilai > 40){
    $grade = "C";
} elseif($nilai > 30){
    $grade = "D";
} elseif($nilai > 20){
    $grade = "E";
} else {
    $grade = "F";
}

echo "Nilai anda: $nilai<br>";
echo "Grade: $grade";
```

```
?>
```

Hasil :



- **Percabangan Switch Case**

Percabangan Switch/Case adalah bentuk lain dari percabangan If/Elseif/Else. Format penulisannya seperti ini:

```
<?php

switch($variabel){
    case <konidisi>:
        // eksekusi kode ini
        break;
    case <kondisi2>:
        // eksekusi kode ini
        break;
    default:
        // eksekusi kode ini
}

?>
```

Contoh:

```
<?php
$level = 3;
switch($level){
    case 1:
        echo "Pelajari HTML";
```



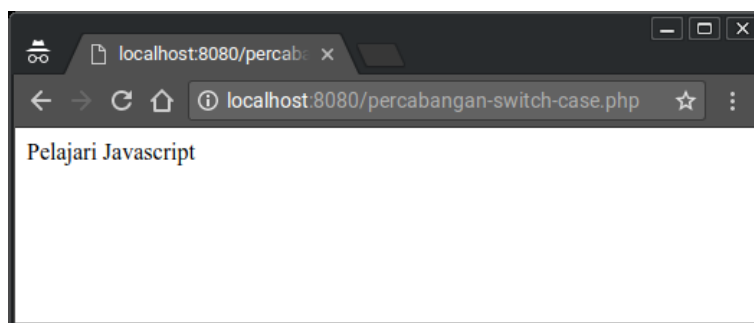
```

        break;
    case 2:
        echo "Pelajari CSS";
        break;
    case 3:
        echo "Pelajari Javascript";
        break;
    case 4:
        echo "Pelajari PHP";
        break;
    default:
        echo "Kamu bukan programmer!";
}
?>

```

Ada 5 pilihan dalam kondisi di atas. Pilihan default akan dipilih apabila nilai variabel \$level tidak ada dalam pilihan case.

Hasilnya:



- **Percabangan dengan Operator Ternary**

Percabangan menggunakan operator ternary adalah bentuk sederhana dari percabangan If/Else. Formatnya seperti ini:

```

<?php
    <kondisi> ? benar : salah;
?>

```

Contoh :

```

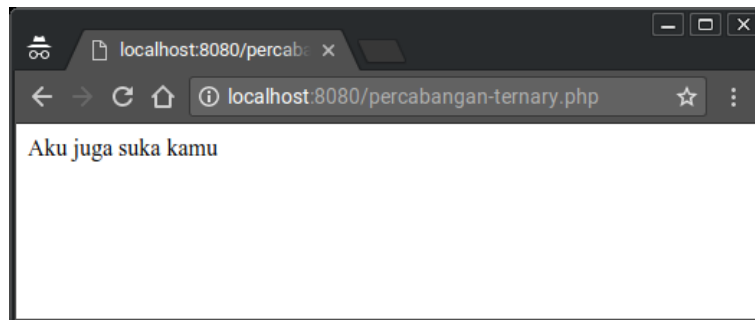
<?php

```

```
$suka = true;  
$suka ? echo "Aku juga suka kamu": echo "Baiklah!";  
?>
```

Artinya: jika variabel \$suka bernilai true maka cetak "Aku juga suka kamu".
Tapi kalau bernilai false, maka cetak "Baiklah!".

Hasil :



2.3 Pengertian Pernyataan Perulangan Dalam PHP

Definisi

Looping (atau perulangan) adalah fungsi pada bahasa pemrograman untuk menjalankan baris kode secara berulang-ulang selama kondisi masih terpenuhi. Ada dua jenis perulangan dalam pemrograman:

- 1) Counted loop;
- 2) Uncounted loop.

Counted loop adalah perulangan yang sudah jelas banyak pengulangannya. Sedangkan Uncounted loop tidak pasti berapa kali dia akan mengulang.



Pada PHP ada 4 jenis perulangan yang bisa kita gunakan:

1. Perulangan For
2. Perulangan While
3. Perulangan Do/While
4. Perulangan Foreach

1. Perulangan For

Perulangan For adalah perulangan yang termasuk dalam counted loop, karena kita bisa menentukan jumlah perulangannya. Bentuk dasar perulangan for:

```
<?php
for ($i = 0; $i < 10; $i++){
    // blok kode yang akan diulang di sini!
}
?>
```

Veriabel \$i dalam perulangan For berfungsi sebagai counter yang menghitung berapa kali ia akan mengulang.

Hitungan akan dimulai dari nol (0), karena kita memberikan nilai \$i = 0.

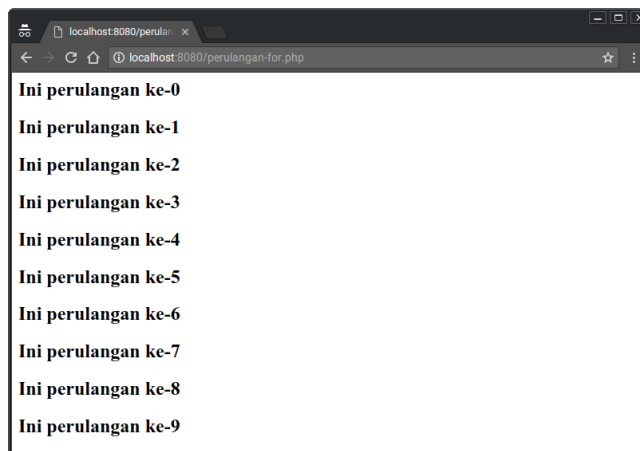
Lalu, perulangan akan diulang selama nilai \$i lebih kecil dari 10. Artinya, perulangan ini akan mengulang sebanyak 10x.

Maksud dari \$i++ adalah nilai \$i akan ditambah 1 disetiap kali melakukan perulangan.

Contoh:

```
<?php
for($i = 0; $i < 10; $i++){
    echo "<h2>Ini perulangan ke-$i</h2>";
}
?>
```

Hasilnya:



2. Perulangan While

Perulangan while adalah perulangan yang termasuk dalam uncounted loop. Karena biasanya digunakan untuk mengulang sesuatu yang belum jelas jumlah pengulangannya. Namun, perulangan while juga bisa digunakan seperti perulangan for sebagai counted loop. Bentuk dasarnya:

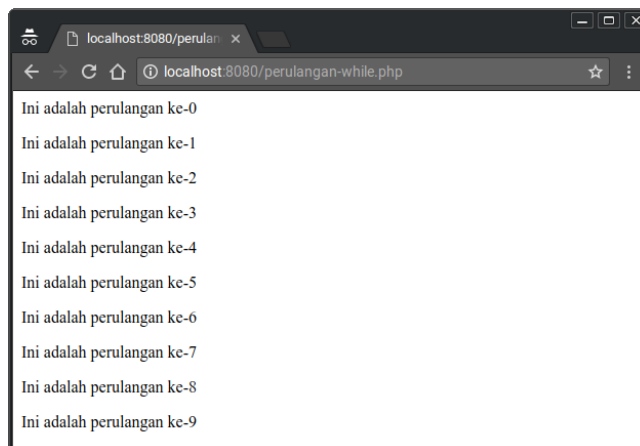
```
<?php
while (<kondisi>){
    // blok kode yang akan diulang di sini
```

```
}  
?>
```

Contoh:

```
<?php  
$ulangi = 0;  
while($ulangi < 10){  
    echo "<p>Ini adalah perulangan ke-$ulangi</p>";  
    $ulangi++;  
}  
?>
```

Hasilnya:



Perulangan while akan terus mengulang selama nilai \$ulangi lebih kecil dari 10.

Lalu di dalam perulangan kita melakukan increment nilai \$ulangi dengan \$ulangi++. Artinya: Tambah 1 disetiap pengulangan.

Hati-hati, jangan sampai lupa menambahkan increment, atau kode yang akan mempengaruhi pengulangan. Karena kalau tidak, pengulangannya tidak akan pernah berhenti dan akan membuat komputer kita hang.

3. Perulangan Do/While

Perulangan Do/While sama seperti perulangan while. Ia juga tergolong dalam uncounted loop. Perbedaan Do/While dengan while terletak pada cara iya memulai pengulangan.

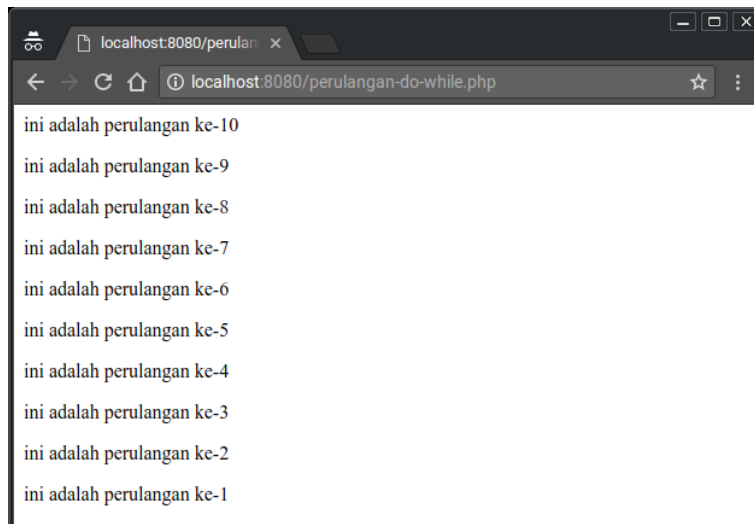
Perulangan Do/While akan selalu melakukan pengulangan sebanyak 1 kali, kemudian melakukan pengecekan kondisi. Sedangkan perulangan while akan mengecek kondisi terlebih dahulu, baru melakukan pengulangan. Bentuk perulangan Do/While:

```
<?php
do {
    // blok kode yang akan diulang
} while (<kondisi>);
?>
```

Contoh:

```
<?php
$ulangi = 10;
do {
    echo "<p>ini adalah perulangan ke-$ulangi</p>";
    $ulangi--;
} while ($ulangi > 0);
?>
```

Hasilnya:



4. Perulangan Foreach

Perulangan foreach sama seperti perulangan for. Namun, ia lebih khusus digunakan untuk mencetak array.

Bentuk perulangan foreach:

```
foreach($array as $data){  
    echo $data;  
}
```

Contoh:

```
<?php  
$books = [  
    "Panduan Belajar PHP untuk Pemula",  
    "Membangun Aplikasi Web dengan PHP",  
    "Tutorial PHP dan MySQL",  
    "Membuat Chat Bot dengan PHP"  
];  
echo "<h5>Judul Buku PHP:</h5>";  
echo "<ul>";  
foreach($books as $buku){  
    echo "<li>$buku</li>";  
}  
echo "</ul>";  
?>
```

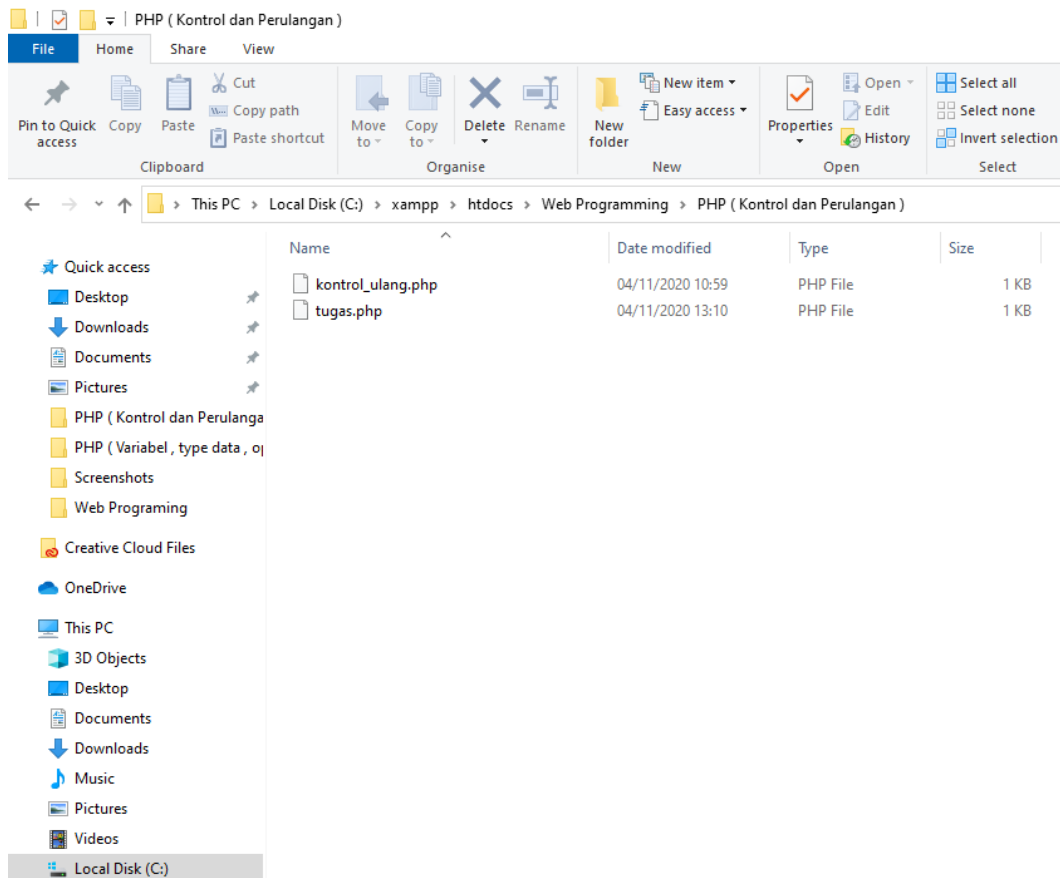
Hasilnya:



BAB III

PEMBAHASAN

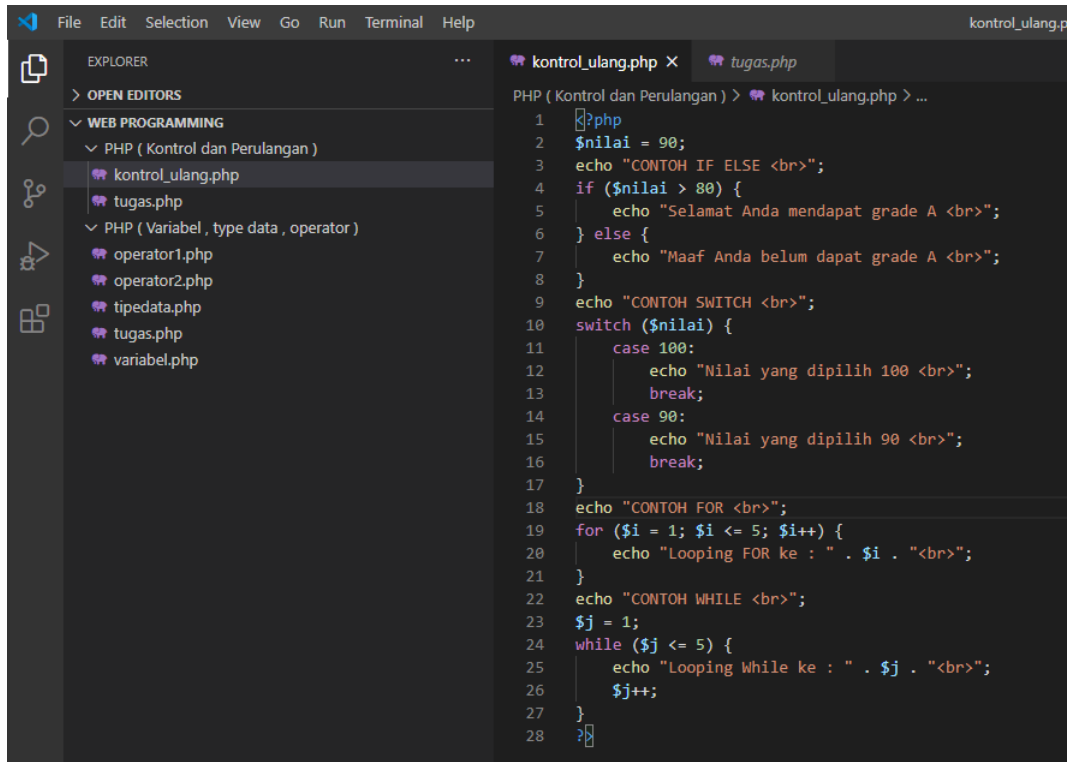
1. Folder dan File



Gambar 1 "Folder Tempat Penyimpanan File Praktikum"

Terdapat 2 file php pada praktikum yang saya lakukan, seperti pada gambar saya menyimpan file tersebut pada "C:\xampp\htdocs\Web Programming\PHP (Kontrol dan Perulangan)" agar php dapat dijalankan menggunakan localhost.

2. Pelaksanaan Praktikum kontrol_ulang.php



```
1 <?php
2 $nilai = 90;
3 echo "CONTOH IF ELSE <br>";
4 if ($nilai > 80) {
5     echo "Selamat Anda mendapat grade A <br>";
6 } else {
7     echo "Maaf Anda belum dapat grade A <br>";
8 }
9 echo "CONTOH SWITCH <br>";
10 switch ($nilai) {
11     case 100:
12         echo "Nilai yang dipilih 100 <br>";
13         break;
14     case 90:
15         echo "Nilai yang dipilih 90 <br>";
16         break;
17 }
18 echo "CONTOH FOR <br>";
19 for ($i = 1; $i <= 5; $i++) {
20     echo "Looping FOR ke : " . $i . " <br>";
21 }
22 echo "CONTOH WHILE <br>";
23 $j = 1;
24 while ($j <= 5) {
25     echo "Looping While ke : " . $j . " <br>";
26     $j++;
27 }
28 <?>
```

Gambar 2.1 “program kontrol_ulang.php”

Penjelasan :

- Skrip PHP dimulai dengan `<?php` dan diakhiri dengan `?>`
- Di PHP, variabel dimulai dengan tanda \$, diikuti dengan nama variabel. Yang termasuk variabel pada skrip adalah `$nilai`, `$i`, `$j` dan tanda = merupakan assignment untuk memberi nilai kepada variabel.
- `echo` digunakan untuk menampilkan data ke layar.
- Pada baris kode ke-4 hingga ke-8 merupakan Percabangan if..else, dimana if ... else statement - mengeksekusi beberapa kode if jika kondisinya benar dan kode else jika pada kondisi if salah. dengan struktur :

```
if (kondisi){
    kode program yang dijalankan jika memenuhi kondisi
} else{
    kode program yang dijalankan jika tidak memenuhi kondisi
}
```

- Kondisi pada if akan dicek terlebih dahulu, dimana (`$nilai > 80`) merupakan kondisi yang akan dicek, yang jika diartikan “apabila

variabel `$nilai` bernilai lebih dari 80 maka akan mengembalikan nilai true dan mengerjakan `echo "Selamat Anda mendapat grade A
"`;

- Else dikerjakan jika kondisi if bernilai false, dan akan menjalankan `echo "Maaf Anda belum dapat grade A
"`;

- Pada baris kode ke-10 hingga ke-17 merupakan Percabangan switch case, dimana switch statement - memilih salah satu dari banyak blok kode untuk dieksekusi. dengan struktur :

```
<?php
switch (n) {
    case label1:
        code to be executed if n=label1;
        break;
    case label2:
        code to be executed if n=label2;
        break;
    case label3:
        code to be executed if n=label3;
        break;
    ...
    default:
        code to be executed if n is different from all
        labels;
}
?>
```

- Cara kerja: Pertama kita memiliki ekspresi tunggal `n` (paling sering sebagai variabel), yang dievaluasi sekali. Nilai ekspresi tersebut kemudian dibandingkan dengan nilai untuk setiap kasus dalam struktur. Jika ada yang cocok, blok kode yang terkait dengan kasus itu dijalankan. `break` digunakan untuk mencegah kode berjalan ke kasus berikutnya secara otomatis. Pernyataan `default` digunakan jika tidak ada kecocokan yang ditemukan.
- Pada program `switch ($nilai)` dimana variabel `$nilai` merupakan ekspresi yang akan dibandingkan
- Lalu `case 100:` dimana 100 merupakan kondisi atau nilai yang akan dicocokkan/dibandingkan dengan `$nilai` untuk case1 dalam struktur. Jika hasil perbandingan mengembalikan nilai true atau sama maka akan mengeksekusi `echo "Nilai yang dipilih 100
"`; `break`;
- Jika case pertama salah maka selanjutnya dicek pada case dibawahnya yaitu `case 90`; dan dilakukan proses yang sama untuk seterusnya.

- Pada baris kode ke-19 hingga ke-21 merupakan Perulangan for. dengan struktur:

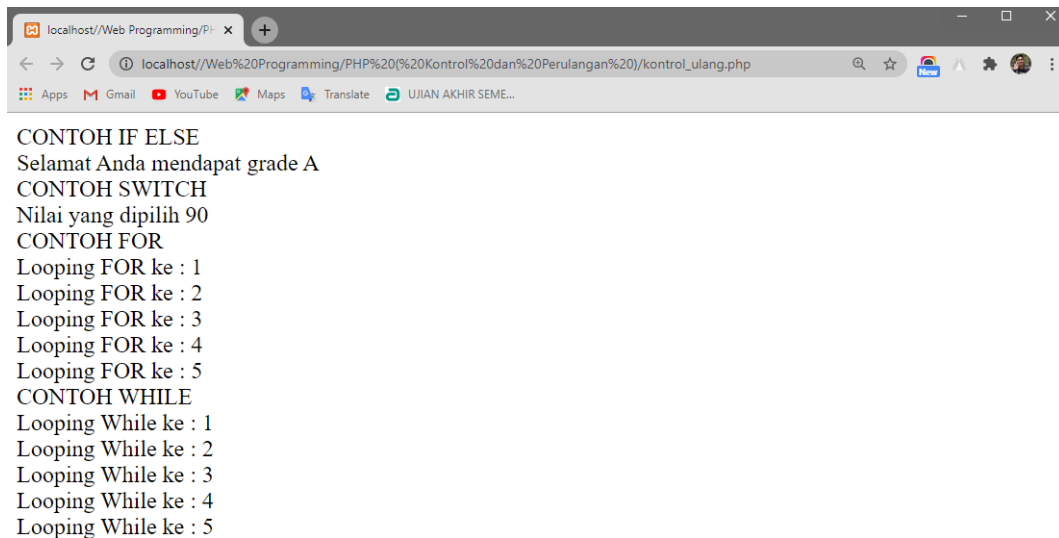
```
<?php
for (init counter; test counter; increment counter) {
    code to be executed for each iteration;
}
?>
```

- Dimana dalam program `for ($i = 1; $i <= 5; $i++)` Variabel `$i` dalam perulangan For berfungsi sebagai counter yang menghitung berapa kali ia akan mengulang.
 - Hitungan akan dimulai dari satu (1), karena kita memberikan nilai `$i = 1`.
 - Lalu, perulangan akan diulang selama nilai memenuhi `$i <= 5`. Artinya, perulangan ini akan mengulang sebanyak 5x dengan mengeksekusi `echo "Looping FOR ke : " . $i . "
";`
 - Maksud dari `$i++` adalah nilai `$i` akan ditambah 1 disetiap kali melakukan perulangan.
- Pada baris kode ke-23 hingga ke-27 merupakan Perulangan while. dengan struktur:

```
<?php
while (condition is true) {
    code to be executed;
}
?>
```

- Perulangan while akan terus mengulang selama nilai `$j` lebih kecil sama dengan 5.
- Lalu di dalam perulangan kita melakukan increment nilai `$j` dengan `$j++`; , Artinya: Tambah 1 disetiap pengulangan.
- Hati-hati, jangan sampai lupa menambahkan increment, atau kode yang akan mempengaruhi pengulangan. Karena kalau tidak, pengulangannya tidak akan pernah berhenti dan akan membuat komputer kita hang.
- perulangan ini akan mengulang sebanyak 5x dengan mengeksekusi `echo "Looping While ke : " . $j . "
";`

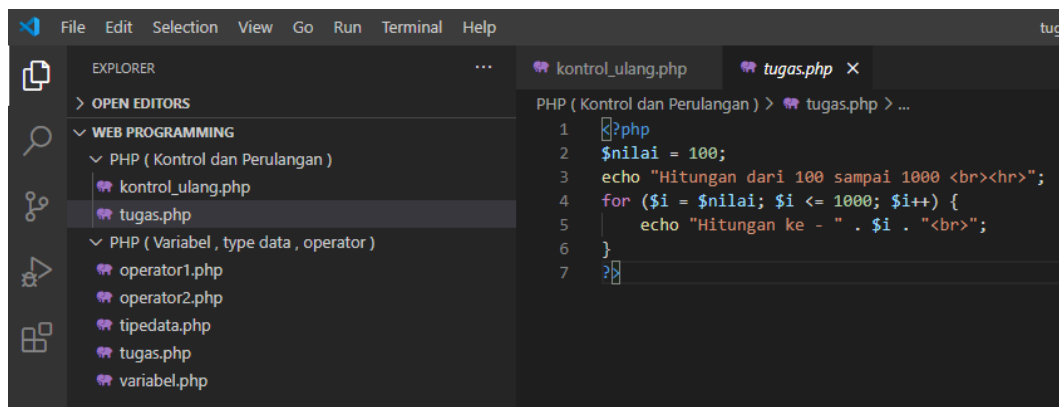
Berikut tampilan pada browser :



Gambar 2.2 “tampilan pada browser”

3. Pelaksanaan Praktikum tugas

- a) **Buat program perulangan yang menghasilkan nilai hitungan dari 100 sampai 1000.**



Gambar 3.1 “program tugas.php”

Penjelasan :

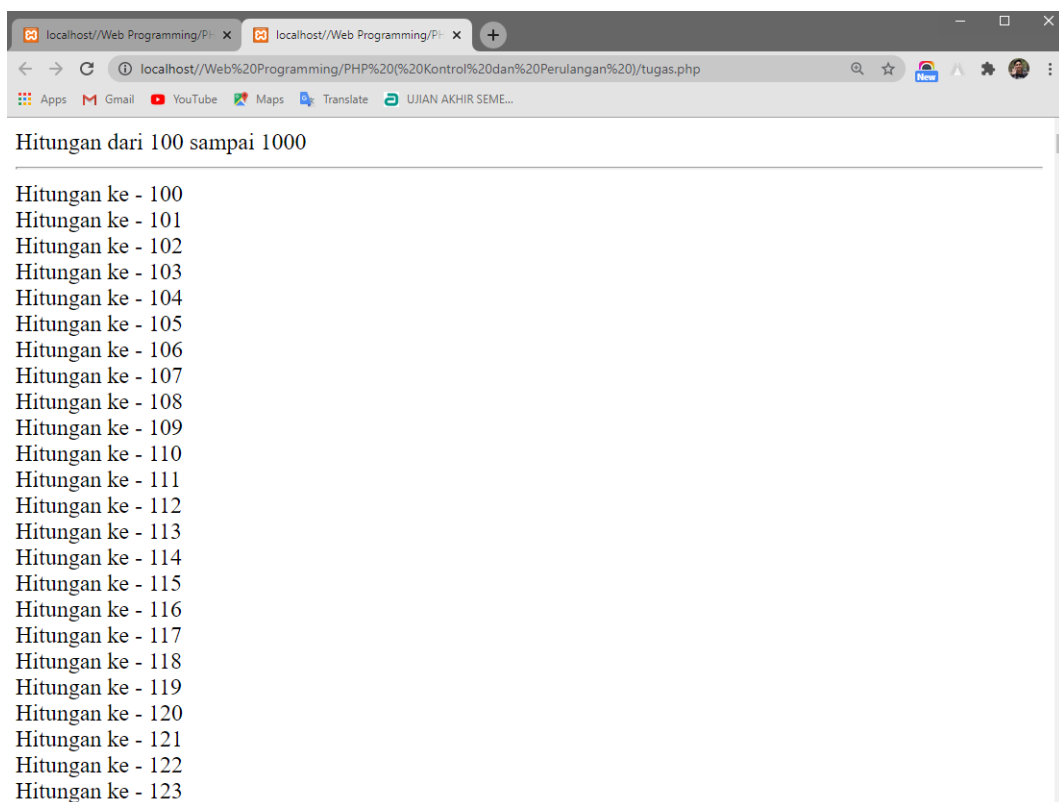
- Skrip PHP dimulai dengan `<?php` dan diakhiri dengan `?>`
- Di PHP, variabel dimulai dengan tanda `$`, diikuti dengan nama variabel. Yang termasuk variabel pada skrip adalah `$nilai` dan tanda `=` merupakan assignment untuk memberi nilai kepada variabel pada program ini variabel `$nilai` diberi nilai 100; yang nantinya akan digunakan sebagai nilai awal counter pada looping for.
- `echo` digunakan untuk menampilkan data ke layar.
- Pada baris kode ke-4 hingga ke-6 merupakan Perulangan for dengan struktur :

```
<?php
```

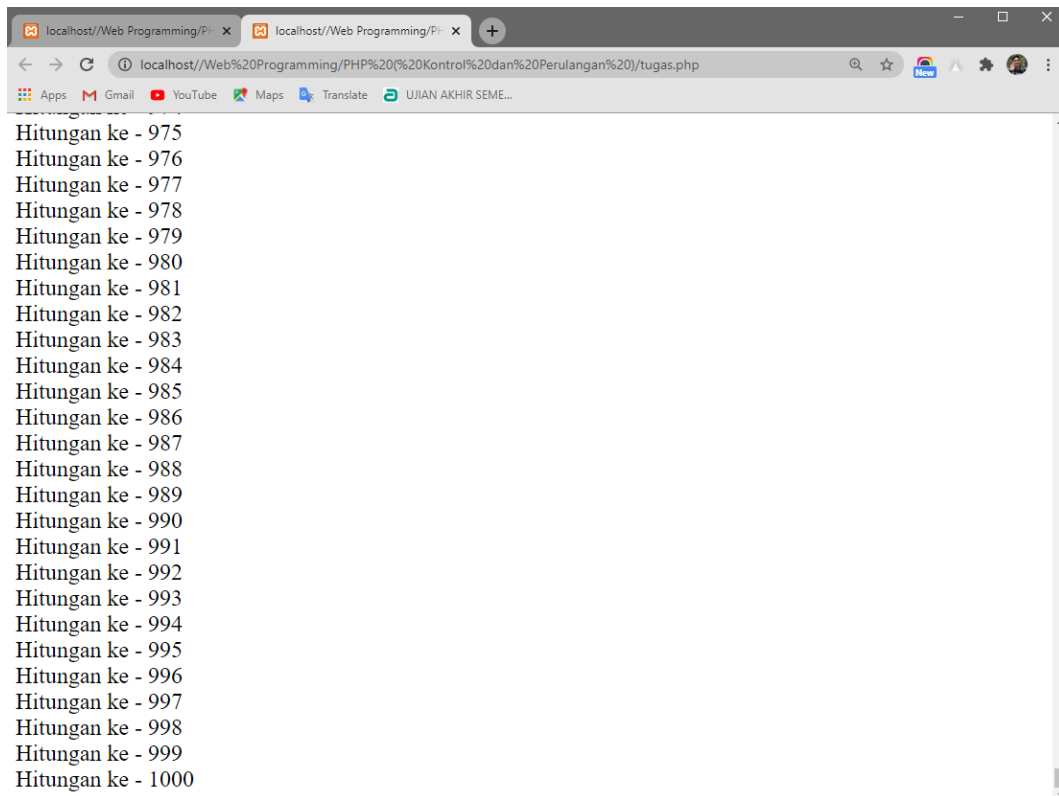
```
for (init counter; test counter; increment counter) {  
    code to be executed for each iteration;  
}  
?>
```

- Dimana dalam program `for ($i = $nilai; $i <= 1000; $i++)` dengan kondisi tersebut akan menghasilkan perulangan sebanyak 900x atau mulai dari 100 hingga 1000.
- Variabel `$i` dalam perulangan For berfungsi sebagai counter yang menghitung berapa kali ia akan mengulang. Pada program variabel `$i = $nilai` artinya variabel `$i` akan bernilai sama dengan variabel `$nilai`.
- Hitungan akan dimulai dari satu (100), karena kita memberikan nilai `$i = $nilai`.
- Lalu, perulangan akan diulang selama nilai memenuhi `$i <= 1000;`. Artinya, perulangan ini akan mengulang sebanyak 900x dengan mengeksekusi `echo "Hitungan ke - " . $i . "
";`
- Maksud dari `$i++` adalah nilai `$i` akan ditambah 1 disetiap kali melakukan perulangan.

Berikut tampilan pada browser :



Gambar 3.2 “tampilan pada browser (top page)”



Gambar 3.3 “tampilan pada browser (bottom page)”

b) Carilah untuk kasus apa logika perulangan dan kontrol banyak digunakan.

Pada dunia nyata atau saat programmer mengerjakan project, logika perulangan dan kontrol banyak digunakan pada :

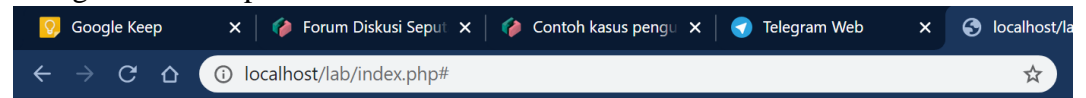
- Membuat algoritma dengan statement kontrol untuk login page.
- Menggunakan statement kontrol untuk melakukan pemilihan metode pembayaran pada program-program e-commerce.
- Menggunakan looping Ketika ingin menampilkan data dari database ke tabel HTML contoh dari

(<https://www.codepolitan.com/forum/thread/detail/1318/contoh-kasus-pengulangan-dalam-kehidupan-sehari-hari-fMKo>)

```
.  
.br/>for ($i=0; $i<$total; $i++)  
{  
    ?>  
  
    <tr>  
        <td><?php echo $karyawan[$i]['nama'];?></td>  
        <td><?php echo $karyawan[$i]['jabatan'];?></td>  
    </tr>  
  
    <?php  
}
```

.

Menghasilkan output :



| Nama | Jabatan |
|---------|---------------|
| Budi | Kepala Gudang |
| Siska | Frontliner |
| Bambang | Security |

- Pemrograman pengolahan citra digital. Misal pada aplikasi Instagram, pada gambar di layar hp yang terdiri dari pixel pixel. ketika kita menerapkan filter / kasih efek sehingga gambarnya jadi lebih cantik, bagus, dll saat itu sebenarnya dibelakang layar melakukan komputasi digital, mengolah setiap pixelnya, tentunya dengan algoritma looping.
- Melakukan looping di JS untuk mengambil data dari API / Ajax berupa JSON untuk mengambil data dan menjadikannya untuk di tampilkan sebagai tabel.

Dan masih banyak lagi contoh lainnya tergantung bagaimana seorang programmer menerapkan logika pemrograman menggunakan statement **perulangan** dan **kontrol** untuk menyelesaikan permasalahan dalam melakukan developing program .

BAB IV

KESIMPULAN

PHP memiliki syntax dasar yang sederhana. Kita sudah belajar bagaimana aturan penulisannya yang case sensitive hingga komentar sebagai penjelasan kodenya.

Pada penggunaannya, PHP bisa digabungkan dengan bahasa pemrograman lain, seperti HTML, CSS, dan Javascript seperti contoh di atas. Bahkan, bahasa pemrograman ini juga mampu bekerja dengan baik bersama MySQL, sebagai sebuah sistem database.

Struktur kontrol memungkinkan kita untuk melakukan tindakan di luar alur default pengeksekusian program: yaitu dari atas ke bawah. Dengan menggunakan struktur kontrol, kita bisa mengeksekusi sebagian kode program, dan tidak mengeksekusi sebagian yang lain.

Di dalam PHP, terdapat dua jenis struktur kontrol: yaitu logika percabangan, dan juga blok kode perulangan. Untuk logika percabangan, terdapat dua jenis utama: yaitu if .. else dan switch .. case. Juga ada alternatif lain seperti ternary dan null coalescing operator.

DAFTAR PUSTAKA

BKPM Pemrograman Web.2020.” PHP (Kontrol dan Perulangan)” (Diakses 4 November 2020).

Achmad Muhardian. 2015.” <https://www.petanikode.com/php-percabangan/>” (Diakses 4 November 2020).

Achmad Muhardian. 2015.” <https://www.petanikode.com/php-perulangan/>” (Diakses 4 November 2020).

Nurul Huda.2020.” <https://jagongoding.com/web/php/dasar/logika-percabangan/>” (Diakses 4 November 2020).

W3schools.2020.”<https://www.w3schools.com/php/default.asp/>” (Diakses 4 November 2020).