

LAPORAN HASIL PRAKTIKUM PEMROGRAMAN WEB & MOBILE 1



NAMA : SATRIA SEPTA ARIANTO

NIM : 193020503026

KELAS : A

MODUL : I (PHP/HYPertext PREPROCESSOR)

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PALANGKA RAYA
TAHUN 2021**

BAB I

TUJUAN DAN LANDASAN TEORI

1.1. Tujuan

1.1.1 Mahasiswa memahami dasar-dasar pemrograman PHP yang biasa digunakan pada umumnya.

1.1.2 Mahasiswa mampu membuat program sederhana menggunakan bahasa pemrograman PHP.

1.2. Landasan Teori

1.2.1 Variabel

Pada pemrograman php, variabel dimulai dengan tanda “\$” dan diikuti dengan nama variabel. Tidak seperti bahasa pemrograman lainnya, php tidak memiliki perintah khusus untuk mendeklarasikan variabel. Tipe data ditentukan ketika nilai dimasukkan kedalam variabel tersebut. Sebuah variabel bisa memiliki nama yang singkat (seperti x dan y) atau nama deskriptif (contoh : umur, namamobil, total_volume). Aturan-aturan untuk variabel php adalah sebagai berikut:

- Sebuah variabel dimulai dengan tanda “\$”, diikuti oleh nama variabel.
- Sebuah nama variabel harus mulai dengan huruf atau karakter garis bawah “_”.
- Sebuah nama variabel tidak boleh dimulai dengan angka.
- Sebuah nama variabel hanya bisa mengandung karakter alpha-numeric dan garis bawah (A-z, 0-9, dan _)
- Nama-nama variabel adalah case-sensitive, yaitu huruf besar dan huruf kecil mempengaruhi perbedaan variabel (\$umur dan \$UMUR adalah dua variabel yang berbeda)

Perintah yang biasa digunakan untuk mencetak isi dari sebuah variabel adalah echo.(Praktikum, 2001.)

Berikut adalah contoh program menentukan nilai variabel dan mencetaknya di browser :

```
<?php
    $txt = "W3Schools.com";
    echo "I love $txt!";
    echo "I love " . $txt . "!"; // menghasilkan keluaran yang sama
    dengan perintah cetak diatas
?>
```

1.2.2 Echo dan Print

Ada dua cara untuk menampilkan output di PHP yaitu dengan menggunakan perintah echo dan print. Terdapat sedikit perbedaan dalam penggunaan echo dan print, echo tidak mengembalikan nilai sedangkan print mengembalikan nilai 1 sehingga bisa digunakan dalam ekspresi. Echo bisa menggunakan banyak parameter sedangkan print bisa menggunakan satu argument. Secara marjinal echo lebih cepat dari print.

```
<?php
    echo "<h2>PHP is Fun!</h2>";
    echo "Hello world!<br>";
    echo "I'm about to learn PHP!<br>";
    echo "This ", "string ", "was ", "made ", "with multiple parameters.";
    print "<h2>PHP is Fun!</h2>";
    print "Hello world!<br>";
    print "I'm about to learn PHP!";
?>
```

1.2.3 Tipe Data

Variabel bisa menyimpan data dengan tipe yang berbeda-beda, berikut adalah tipe-tipe data yang didukung oleh PHP

- String

- Integer
- Float
- Boolean
- Array
- Object
- NULL
- Resource

1.2.4 Fungsi-fungsi Manipulasi String

String merupakan susunan dari karakter yang memiliki fungsi khusus untuk mengolahnya. Seperti fungsi `strlen()`, berfungsi menghitung panjang dari string. Contohnya sebagai berikut:

```
<?php
    echo strlen("Hello world!");
?>
```

Hasil dari kode tersebut adalah panjang dari string “Hello world!” yaitu 12 karakter. Panjang string juga bisa diukur perkata menggunakan fungsi `str_word_count()` seperti contoh berikut:

```
<?php
    echo str_word_count("Hello world!");
?>
```

Keluaran dari kode tersebut berupa jumlah kata pada string “Hello world!” yaitu 2. PHP juga memiliki fungsi yang bisa digunakan untuk membalikkan susunan string, yaitu menggunakan fungsi `strrev()` seperti contoh kode program berikut yang menghasilkan string “!dlrow olleH” dari inputan string “Hello world!”.

```
<?php
    echo strrev("Hello world!");
?>
```

PHP juga memiliki fungsi untuk menemukan teks yang spesifik didalam sebuah string, yaitu dengan menggunakan fungsi `strpos()`. Jika

teks yang dicari ditemukan didalam string, maka fungsi akan mengembalikan index posisi dari karakter awal teks pertama yang cocok. Jika tidak ditemukan, maka fungsi strpos() akan mengembalikan nilai FALSE. Kode program berikut adalah contoh dari penggunaan fungsi strpos() pada string “Hello world!” untuk menemukan teks “world”, sehingga keluaran yang dihasilkan adalah 6.

```
<?php
    echo strpos("Hello world!", "world");
?>
```

PHP juga memiliki fungsi untuk mengganti teks didalam string, yaitu dengan menggunakan fungsi str_replace(). Contoh berikut adalah kode program yang digunakan untuk mengganti teks “world” pada string “Hello world!” dengan teks “Dolly”.

```
<?php
    echo str_replace("world", "Dolly", "Hello world!");
?>
```

1.2.5 Operator PHP

Operator digunakan untuk melakukan operasi pada variabel-variabel dan nilai-nilai. Operator dibagi menjadi beberapa kelompok sebagai berikut:

- Operator Aritmatika
- Operator Assignment(penugasan)
- Operator Perbandingan
- Operator Iterasi
- Operator Logika
- Operator String
- Operator Array

1.2.6 Perulangan

- Perulangan For

```
for (init_awal, kondisi, counter) {  
    statement-yang-diulang;  
}
```

Keterangan :

f init_awal merupakan inisialisasi atau nilai awal variable.

f kondisi merupakan statemen kondisi yang akan membatasi perulangan.

f counter merupakan penambahan atau pengurangan nilai variabel sehingga perulangan tetap berjalan.

- Perulangan While

```
init_awal;  
while (kondisi) {  
    statement-yang-diulang;  
    counter;  
}
```

Keterangan :

f init_awal merupakan inisialisasi atau nilai awal variable.

f kondisi merupakan statemen kondisi yang akan membatasi perulangan.

f counter merupakan penambahan atau pengurangan nilai variabel sehingga perulangan tetap berjalan.

- Perulangan Do...While

```
init_awal;  
do {  
    statement-yang-diulang;  
    counter;  
} while (kondisi);
```

Keterangan :

f *init_awal* merupakan inisialisasi atau nilai awal variable.

f *kondisi* merupakan statemen kondisi yang akan membatasi perulangan.

f *counter* merupakan penambahan atau pengurangan nilai variabel sehingga perulangan tetap berjalan.

f Pada struktur *do...while*, pemeriksaan kondisi ada di bawah, sehingga statement yang berada dalam block *do...while* setidaknya akan dieksekusi sebanyak satu kali.

- Perulangan Foreach

```
foreach (array_expression as $value) statement;  
foreach (array_expression as $key => $value) statement;
```

Keterangan :

f Struktur *foreach* biasanya digunakan untuk melakukan perulangan berdasarkan isi suatu array. Perulangan akan berakhir jika isi array telah habis.

1.2.7 Fungsi dalam PHP

Kekuatan dari PHP berasal dari fungsi-fungsinya. PHP memiliki lebih dari 1000 fungsi built-in. Namun dalam PHP bisa juga dibuat fungsi sendiri. Fungsi adalah blok kode yang bisa digunakan secara berulang-ulang kali dalam program. Fungsi tidak akan dieksekusi secara langsung jika halaman web sedang dimuat. Sebuah fungsi akan dieksekusi oleh fungsi yang memanggilnya. Syntax dari fungsi yang dibuat sendiri adalah sebagai berikut:

```
function nama_fungsi(parameter1,...,n) {  
    statement2;  
}
```

Nama fungsi bisa dimulai dengan huruf atau garis bawah (bukan bilangan). Tentukan nama fungsi berdasarkan apa kegunaannya.

Informasi dari luar fungsi bisa dimasukkan kedalam fungsi melalui argument. Argumen adalah seperti variabel. Argumen dispesifikasikan setelah nama fungsi didalam kurung. Argument bisa dibuat banyak, antara deklarasi argument harus dipisah dengan tanda koma. Jika argumen memiliki nilai default, maka argument tersebut bisa langsung ditentukan nilainya dengan tanda sama dengan. Berikut adalah contoh penentuan argument dengan nilai default.

```
<?php
    function setHeight($minheight = 50) {
        echo "The height is : $minheight <br>";
    }
    setHeight(350);
    setHeight(); // will use the default value of 50
    setHeight(135);
    setHeight(80);
?>
```

Fungsi juga bisa memberikan nilai kembalian. Untuk memberikan nilai kembalian, fungsi tersebut harus menggunakan perintah return. Berikut adalah contoh dari fungsi yang mengembalikan sebuah nilai:

```
<?php
    function sum($x, $y) {
        $z = $x + $y;
        return $z;
    }
    echo "5 + 10 = " . sum(5, 10) . "<br>";
    echo "7 + 13 = " . sum(7, 13) . "<br>";
    echo "2 + 4 = " . sum(2, 4);
?>
```


1.2.8 Array

Array merupakan tipe data terstruktur yang berguna untuk menyimpan sejumlah data yang bertipe sama. Bagian yang menyusun array disebut elemen array, yang masing-masing elemen dapat diakses tersendiri melalui index array. Index array dapat berupa bilangan integer atau string.

Untuk mendeklarasikan atau mendefinisikan sebuah array di PHP bisa menggunakan keyword `array()`. Jumlah elemen array tidak perlu disebutkan saat deklarasi. (Achmad Solichin. S.Kom, 2016)

```
<?php
$arrBuah = array ("Mangga", "Apel", "Pisang", "Jeruk");
echo $arrBuah[0]; //Mangga
echo $arrBuah[3]; //Jeruk

$arrWarna = array();
$arrWarna[] = "Merah";
$arrWarna[] = "Biru";
$arrWarna[] = "Hijau";
$arrWarna[] = "Putih";
echo $arrWarna[0]; //Merah
echo $arrWarna[2]; //Hijau

?>
```

Gambar 1.1 Array

BAB II PEMBAHASAN

2.1 Array untuk Menyimpan Nama

Untuk menyimpan record nama, bisa menggunakan array. Berikut adalah sintaks php untuk menyimpan record nama.

```
$names= ["samuel park", "karen", "aiden", "kim jo jo","trace"];
```

Gambar 2.1 Array Nama

Array dapat menyimpan record tanpa batas ketika batasnya tidak ditentukan dari awal. \$ adalah tanda untuk nama dari array sedangkan *names* adalah nama dari array itu sendiri.

2.2 Program Hitung Jumlah Kata dan Jumlah Huruf

Dari nama yang telah disimpan dalam array *\$names*, dibuatlah fungsi baru untuk menghitung huruf dari nama tersebut. Sebenarnya, sintaks untuk menghitung huruf dan kata sudah tersedia dalam php sendiri. Namun, karena pada sintaks hitung huruf yaitu *strlen* atau string length juga menghitung spasi maka dibuatlah fungsi hitung huruf dengan mengoperasikan *strlen* dikurang spasi yang ada dengan sintaks *substr_count(\$kalimat, ' ')*. *\$kalimat* adalah parameter dari fungsi ini. Keseluruhan fungsinya seperti di bawah ini.

```
function hitunghuruf($kalimat){  
    return strlen($kalimat) - substr_count($kalimat, ' ');  
}
```

Gambar 2.2 Fungsi Hitung Huruf

Setelah membuat fungsi hitung huruf diatas, sekarang tinggal memanggilnya pada badan program pada file php. *str_word_count* adalah sintaks untuk menghitung kata, sedangkan menghitung huruf menggunakan

fungsi yang telah dibuat sebelumnya. *\$name* adalah array nama yang akan masuk ke paramater *\$kalimat* agar dapat diproses.

```
<td><?php echo str_word_count($name)."<br>"; ?></td>  
<td><?php echo hitunghuruf($name)."<br>"; ?></td>
```

Gambar 2.3 Sintaks Hitung Kata dan Hitung Huruf

2.3 Program Kebalikan Nama

Untuk program kebalikan nama sendiri, sudah ada sintaks untuk melakukannya yaitu *strrev* atau string reverse. *\$name* setelah sintaks tersebut adalah array yang menyimpan nama untuk dijadikan objek sintaks sebelumnya untuk diproses.

```
<td><?php echo strrev($name)."<br>"; ?></td>
```

Gambar 2.4 Sintaks Kebalikan Nama

2.4 Program Hitung Vokal dan Konsonan

Untuk menghitung huruf vokal dan konsonan, diperlukan fungsi untuk melakukannya sehingga nantinya tinggal dipanggil pada badan program. Untuk menghitung vokal dibuatlah fungsi *vokal* yang memiliki parameter *\$kalimat* untuk menyimpan objek nantinya. Fungsi ini sendiri mendeklarasikan huruf vokal seperti a, i, u, e, o didalamnya. Sehingga ketika fungsi ini dipanggil fungsi ini hanya akan menghitung huruf vokal yang ada. *\$count* adalah variabel untuk menyimpan nilai huruf vokal dan mengembalikannya pada fungsi.

```
function vokal($kalimat) {  
    $a = substr_count($kalimat, 'a');  
    $i = substr_count($kalimat, 'i');  
    $u = substr_count($kalimat, 'u');  
    $e = substr_count($kalimat, 'e');  
    $o = substr_count($kalimat, 'o');  
    $count = ($a+$i+$u+$e+$o);  
  
    return $count;  
}
```

Gambar 2.5 Fungsi Hitung Huruf Vokal

Sedangkan, untuk menghitung konsonan dibuatlah fungsi *konsonan* yang memiliki parameter *\$kalimat* untuk menyimpan objek nantinya. Untuk menghitung jumlah konsonan yang ada, menggunakan *strlen* untuk menghitung semua huruf yang ada. Mengapa pada variabel *\$jumlah* yang digunakan untuk menyimpan panjang huruf yang ada dilakukan operasi pengurangan kosong? Maksud pengurangan kosong ini adalah menghilangkan spasi, sehingga spasi tidak terhitung sebagai konsonan. Fungsi ini juga mendeklarasikan huruf vokal seperti a, i, u, e, o didalamnya. Pada variabel *\$jumlah* yang telah dibuat diatas kembali dilakukan pengurangan terhadap huruf vokal yang bermaksud untuk menghilangkan huruf vokal yang akan terhitung pada jumlah huruf yang ada. Lalu akan disimpan pada variabel *\$count*. *\$count* adalah variabel untuk menyimpan nilai huruf vokal dan mengembalikannya pada fungsi.

```
function konsonan($kalimat) {
    $jumlah = strlen($kalimat) - substr_count($kalimat, ' ');
    $a = substr_count($kalimat, 'a');
    $i = substr_count($kalimat, 'i');
    $u = substr_count($kalimat, 'u');
    $e = substr_count($kalimat, 'e');
    $o = substr_count($kalimat, 'o');
    $count = $jumlah - ($a+$i+$u+$e+$o);

    return $count;
}
```

Gambar 2.6 Fungsi Hitung Huruf Konsonan

Berikut adalah sintaks pemanggilan fungsi vokal dan konsonan. Dimana setelah fungsi dipanggil, ada *\$name* sebagai variabel untuk parameter fungsi sehingga objek array nama itu yang akan diproses.

```
<td><?= vokal($name); ?></td>
<td><?= konsonan($name); ?></td>
```

Gambar 2.7 Sintaks Panggil Fungsi Vokal dan Konsonan

2.5 Tambahan

Berikut adalah kode program html dan php untuk setiap poin sebelumnya. Terbentuk dengan tabel yang terdiri dari tabel header dan tabel body yang berisi datanya. Bagian ini hanya body dari html karena bagian headnya tidak banyak ditambah. Pada tabel dibuat *border="1"* untuk garis dari tabelnya, *cellpadding="7"* untuk mengatur jarak dari border tabel ke isi teksnya, *cellspacing="1"* untuk mengatur jarak antara garis border luar dan dalam. Untuk tabel headnya dibuat style font, warna teks dan ukurannya. Setelah itu ada nama dari setiap tabel header.

Pada tabel body, dibungkus dengan sintaks php *foreach (\$names as \$name)* adalah perulangan pada array nama dan pada array nama sebelumnya menggunakan *\$names* dimisalkan menjadi *\$name* untuk pemanggilan. Terdapat juga style untuk font, warna font, ukuran dan text align atau posisi teks. *Echo* adalah perintah seperti print untuk mencetak kata atau kalimat atau bisa juga memanggil sintaks dan fungsi.

```
<table border="1" cellpadding="7" cellspacing="1">
  <thead style="font-family: 'Open Sans', sans-serif; color:crimson; font-size: 20px;">
    <tr>
      <th>Nama</th>
      <th>Jumlah Kata</th>
      <th>Jumlah Huruf</th>
      <th>Kebalikan Nama</th>
      <th>Vokal</th>
      <th>Konsonan</th>
    </tr>
  </thead>

  <?php foreach ($names as $name) : ?>
    <tbody style="font-family: 'Open Sans', sans-serif; color:blue;
      font-size: 14px; text-align:center">
      <tr>
        <td><?php echo $name."<br>"; ?></td>
        <td><?php echo str_word_count($name)."<br>"; ?></td>
        <td><?php echo hitunghuruf($name)."<br>"; ?></td>
        <td><?php echo strrev($name)."<br>"; ?></td>
        <td><?= vokal($name); ?></td>
        <td><?= konsonan($name); ?></td>
      </tr>
    </tbody>
  <?php endforeach; ?>
</table>
```

Gambar 2.8 Tabel Header dan Tabel Data

Berikut adalah hasil dari semua program diatas.

Nama	Jumlah Kata	Jumlah Huruf	Kebalikan Nama	Vokal	Konsonan
samuel park	2	10	krap leumas	4	6
karen	1	5	nerak	2	3
aiden	1	5	nedia	3	2
kim jo jo	3	7	oj oj mik	3	4
trace	1	5	ecart	2	3

Gambar 2.9 Hasil



Nama	Jumlah Kata	Jumlah Huruf	Kebalikan Nama	Vokal	Konsonan
samuel park	2	10	krap leumas	4	6
karen	1	5	nerak	2	3
aiden	1	5	nedia	3	2
kim jo jo	3	7	oj oj mik	3	4
trace	1	5	ecart	2	3

Gambar 2.10 Tampilan Web

BAB III

KESIMPULAN

PHP merupakan bahasa pemrograman tingkat atas yang memiliki berbagai banyak fitur untuk menunjang aktivitas programmer. Dimana fitur penting seperti fungsi biasa yang mudah digunakan dan fungsi manipulasi yang sudah built-in pada bahasa PHP sendiri. Operator yang digunakan pun berbagai macam sehingga kita bisa menghasilkan berbagai macam program berbeda dengan menggunakan setiap kegunaannya.

DAFTAR PUSTAKA

- Achmad Solichin. S.Kom. (2016). Pemrograman Web dengan PHP dan MySQL - Achmad Solichin - Google Buku. *Universitas Budi Luhur, June*, 85.
<https://books.google.co.id/books?id=kcD4BQAAQBAJ&printsec=frontcover&dq=aplikasi+berbasis+web+dengan&hl=id&sa=X&ved=0ahUKEwib-fT80ITYAhVBrI8KHT9GD6QQ6AEIJzAA#v=onepage&q=aplikasi+berbasis+web+dengan&f=false>
- Praktikum. (2021). *MODUL PRAKTIKUM PEMROGRAMAN WEB I Jurusan Teknik Informatika Fakultas Teknik Universitas Palangka Raya*.

LAMPIRAN

```
<?php
$arrBuah = array ("Mangga", "Apel", "Pisang", "Jeruk");
echo $arrBuah[0]; //Mangga
echo $arrBuah[3]; //Jeruk

$arrWarna = array();
$arrWarna[] = "Merah";
$arrWarna[] = "Biru";
$arrWarna[] = "Hijau";
$arrWarna[] = "Putih";
echo $arrWarna[0]; //Merah
echo $arrWarna[2]; //Hijau

?>
```

Gambar 1.1 Array

```
$names= ["samuel park", "karen", "aiden", "kim jo jo","trace"];
```

Gambar 2.1 Array Nama

```
function hitunghuruf($kalimat){
    return strlen($kalimat) - substr_count($kalimat, ' ');
}
```

Gambar 2.2 Fungsi Hitung Huruf

```
<td><?php echo str_word_count($name)."<br>"; ?></td>
<td><?php echo hitunghuruf($name)."<br>"; ?></td>
```

Gambar 2.3 Sintaks Hitung Kata dan Hitung Huruf

```
<td><?php echo strrev($name)."<br>"; ?></td>
```

Gambar 2.4 Sintaks Kebalikan Nama

```
function vokal($kalimat) {
    $a = substr_count($kalimat, 'a');
    $i = substr_count($kalimat, 'i');
    $u = substr_count($kalimat, 'u');
    $e = substr_count($kalimat, 'e');
    $o = substr_count($kalimat, 'o');
    $count = ($a+$i+$u+$e+$o);

    return $count;
}
```

Gambar 2.5 Fungsi Hitung Huruf Vokal

```
function konsonan($kalimat) {
    $jumlah = strlen($kalimat) - substr_count($kalimat, ' ');
    $a = substr_count($kalimat, 'a');
    $i = substr_count($kalimat, 'i');
    $u = substr_count($kalimat, 'u');
    $e = substr_count($kalimat, 'e');
    $o = substr_count($kalimat, 'o');
    $count = $jumlah - ($a+$i+$u+$e+$o);

    return $count;
}
```

Gambar 2.6 Fungsi Hitung Huruf Konsonan

```
<td><?= vokal($name); ?></td>
<td><?= konsonan($name); ?></td>
```

Gambar 2.7 Sintaks Panggil Fungsi Vokal dan Konsonan

```

<table border="1" cellpadding="7" cellspacing="1">
  <thead style="font-family: 'Open Sans', sans-serif; color:crimson; font-size: 20px;">
    <tr>
      <th>Nama</th>
      <th>Jumlah Kata</th>
      <th>Jumlah Huruf</th>
      <th>Kebalikan Nama</th>
      <th>Vokal</th>
      <th>Konsonan</th>
    </tr>
  </thead>

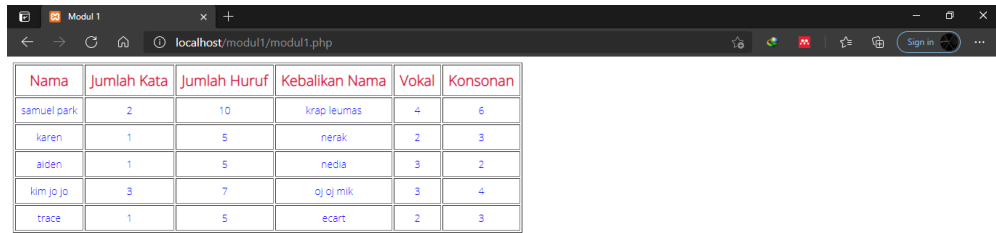
  <?php foreach ($names as $name) : ?>
    <tbody style="font-family: 'Open Sans', sans-serif; color:blue; font-size: 14px; text-align:center">
      <tr>
        <td><?php echo $name."<br>"; ?></td>
        <td><?php echo str_word_count($name)."<br>"; ?></td>
        <td><?php echo hitunghuruf($name)."<br>"; ?></td>
        <td><?php echo strrev($name)."<br>"; ?></td>
        <td><?= vokal($name); ?></td>
        <td><?= konsonan($name); ?></td>
      </tr>
    </tbody>
  <?php endforeach; ?>
</table>

```

Gambar 2.8 Tabel Header dan Tabel Data

Nama	Jumlah Kata	Jumlah Huruf	Kebalikan Nama	Vokal	Konsonan
samuel park	2	10	krap leumas	4	6
karen	1	5	nerak	2	3
aiden	1	5	nedia	3	2
kim jo jo	3	7	oj oj mik	3	4
trace	1	5	ecart	2	3

Gambar 2.9 Hasil



Nama	Jumlah Kata	Jumlah Huruf	Kebalikan Nama	Vokal	Konsonan
samuel park	2	10	krap leumas	4	6
karen	1	5	nerak	2	3
aiden	1	5	media	3	2
kim jo jo	3	7	oj oj milk	3	4
trace	1	5	ecart	2	3

Gambar 2.10 Tampilan Web