

**LAPORAN HASIL PRAKTIKUM
PEMROGRAMAN WEB I**



NAMA : ADITYA ANANDA RAMADHANI
NIM : 193030503072
KELAS : C
MODUL : I (PHP (Hypertext Preprocessor))

JURSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PALANGKA RAYA
2021

BAB I

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. PENGERTIAN PHP

PHP singkatan dari : Hypertext Preprocessor. PHP merupakan bahasa script yang dijalankan pada sisi server (SSS : Server Side Scripting). Database yang didukung PHP antara lain : MySQL, Oracle, dan lain-lain. PHP disebut bahasa pemrograman server side karena PHP diproses pada komputer server. Hal ini berbeda dibandingkan dengan bahasa pemrograman client-side seperti JavaScript yang diproses pada web browser (client). (Erawan, 2014)

Ketika dipanggil dari web browser, program yang ditulis dengan PHP akan di-parsing di dalam web server oleh interpreter PHP dan diterjemahkan ke dalam dokumen HTML, yang selanjutnya akan ditampilkan kembali ke web browser. (Jayanti & Raja, 2020)

PHP dapat digunakan dengan gratis (free) dan bersifat Open Source. PHP dirilis dalam lisensi PHP License, sedikit berbeda dengan lisensi GNU General Public License (GPL) yang biasa digunakan untuk proyek Open Source. (Yuana, 2015)

1.2.2. KARAKTERISTIK PHP

Berikut beberapa karakteristik di dalam PHP :

a. Extension harus .php

Setiap sintak/skrip PHP harus disimpan dengan extension .php, misal test.php. Jika di dalam file .php tidak ada skrip PHP tidak menjadi masalah akan tetap diproses misal didalam file .php isinya hanya skrip html saja maka akan tetap diproses.

b. Sintax harus ditulis dalam dilementer atau sering disebut dengan tag PHP

Sintak PHP tidak akan berjalan atau tidak akan diproses jika sintak tersebut tidak diletakkan di dalam delimeter php yaitu:

<pre><?php dan ?> <?php //pembuka //Disini sintak PHP ?> //penutup</pre>
--

c. Sintak PHP mengikuti bahasa induknya bahasa C

Seperti pada artikel sejarah, php awalnya PHP merupakan sebuah program yang dibuat mengguna-kan bahasa C untuk menangani sebuah form untuk koneksi ke database, sehingga sintak PHP hampir sama dengan baha C, seperti case sensitive dan diakhiri tanda ; di setiap akhir dari sintak.

d. PHP dapat digunakan bersamaan dengan HTML

PHP merupakan bahasa pemograman khusus untuk web maka sangat dengan mudah digunakan bersamaan dengan html, baik tag html di dalam PHP atau sebaliknya.

e. PHP merupakan bahasa server side scripting

Artinya perlu penerjemah atau kompilasi dari sisi server. Salah satu software yang mendukung PHP adalah apache2.

f. PHP Open Source

g. PHP Multi Platform

Dapat dijalankan di berbagai platform OS seperti linux, windows, dan mac yang membuat bahasa pemrograman ini banyak diminati.

1.2.3. ECHO PADA PHP

Fungsi echo() adalah fungsi untuk menampilkan teks ke layar. Fungsi ini dapat digunakan dengan tanda kurung maupun tanpa tanda kurung.

Contoh :

```
<?php  
echo "<h2>Belajar PHP itu mudah!</h2>";  
?>
```

1.2.4. SINTAX PROGRAM PHP

Seperti bahasa pemrograman lainnya, PHP juga memiliki aturan penulisan seperti case sensitivity (perbedaan antara huruf besar dan kecil), cara mengakhiri sebuah baris perintah, dan pengaruh penggunaan spasi dalam pembuatan kode program PHP.

a. Case sensitivity

PHP tidak membedakan huruf besar dan kecil (case insensitive) untuk penamaan fungsi (function), nama class, maupun keyword bawaan PHP seperti **echo**, **while**, dan **class**. Keempat baris berikut dianggap sama dalam PHP.

```
<?php echo "Assalamualaikum Dunia!!!";  
Echo "Assalamualaikum Dunia!!!";  
ECHO "Assalamualaikum Dunia!!!";  
EcHo "Assalamualaikum Dunia!!!";  
?>
```

Akan tetapi, PHP membedakan huruf besar dan huruf kecil (case sensitive) untuk penamaan variable, sehingga \$nama, \$Nama, \$NAMA, \$nAmA akan dianggap sebagai 4 variabel yang berbeda. Sering kali error terjadi karena salah menulis nama variable, yang seharusnya menggunakan huruf kecil, ditulis dengan huruf besar.

b. Penulisan baris perintah dalam PHP

Penulisan baris perintah (statements) di dalam PHP adalah kumpulan perintah PHP yang menginstruksikan PHP untuk melakukan sesuatu. Baris perintah ini bisa terdiri dari satu baris singkat (seperti echo untuk menampilkan text di layer) atau bisa sesuatu yang lebih rumit dan terdiri dari beberapa baris, seperti kondisi if, atau kode perulangan (loop).

c. Karakter **spasi** dan **tab** dalam PHP

Secara umum, karakter spasi dan tab diabaikan di dalam eksekusi program PHP. Pengocoding boleh memecah sebuah statement menjadi beberapa baris, atau menyatukan beberapa statement dalam sebuah baris yang Panjang.

1.2.5. VARIABEL DALAM PHP

Dalam pemrograman, variable merupakan pengenalan di dalam program yang merepresentasikan nilai atau data tertentu. Dengan memasukkan nilai variable, maka dapat mengacu variable tersebut dari tempat lain dalam kode program.

Dalam PHP, variable didefinisikan dengan cara membubuhkan tanda \$ (dollar) di depannya. Berikut contoh penulisan variable yang benar dalam PHP.

```
<?php
$a;
$nim;
>Nama;
$_lokasi_rumah;
$ANGKA_TERBESAR;
?>
```

Berikut merupakan contoh penulisan variable yang salah dalam PHP.

```
<?php
$3gometri; //variabel tidak boleh diawali dengan angka
$_nama depan; //varibel tidak boleh mengandung spasi
$kt*^; //variabel tidak boleh mengandung karakter khusus: * dan ^
?>
```

PHP membedakan variable yang ditulis huruf besar dan kecil, sehingga **\$nama** tidak sama dengan **\$Nama** dan **\$NAMA**, ketiga akan dianggap sebagai tiga variable yang berbeda. Untuk menghindari kesalahan program yang dikarenakan salah merujuk variable, disarankan menggunakan huruf kecil untuk seluruh variable.

Cara memberikan nilai pada variable menggunakan tanda sama dengan (=). Operator sama dengan dikenal dengan istilah Assignment Operator.

Contoh :

```
<?php
$nama = "bang raje";
$umur = 25;
$pesan = "Saya sedang belajar PHP dasar untuk pemula";
?>
```

Variable dalam PHP tidak memerlukan deklarasi terlebih dahulu, kamu bebas membuat variable di tengah-tengah kode program, dan langsung menggunakan tanpa dideklarasikan terlebih dahulu.

Contoh :

```
<?php
$nama = "bang raje";
$umur = 25;
$pesan = "Saya sedang belajar PHP dasar untuk pemula";

echo $nama;
echo $umur;
?>
```

Variable dapat dikatakan sebagai inti dari sebuah bahasa pemrograman, karena melalui variable bisa memanipulasi data input-an agar menjadi nilai yang diinginkan.

1.2.6. TIPE DATA DALAM PHP

Tipe data merupakan jenis sebuah nilai yang ditampung pada sebuah variable, dalam semua bahasa pemrograman pasti memiliki tipe data yang umumnya tidak berbeda. Di dalam PHP sendiri mendukung delapan fitur tipe nilai atau tipe data. Empat nilai scalar (single-value): **integer**, **floating-point**, **string**, dan **Boolean**; dua nilai koleksi: **array** dan **objek**. Sisanya adalah tipe khusus: **resource** dan **null**.

A. Tipe data **Char** dan **String**

Char adalah tipe data yang terdiri dari karakter.

Contoh :

```
<?php
$huruf = 'R';
?>
```

String adalah tipe data yang terdiri dari kumpulan karakter

Contoh :

```
<?php
$alamat = "Jl. Sepakat, Pontianak";
?>
```

B. Tipe data **Integer**

Integer adalah tipe data angka.

Contoh :

```
<?php
$nilai = 07; // angka positif
$point = -07; // angka negatif
?>
```

C. Tipe data **Float-Point / Float**

Float adalah tipe data bilangan pecahan.

Contoh :

```
<?php
$panjang = 12.12;
$lebar = 11.123;
?>
```

D. Tipe data **Boolean**

Tipe data Boolean adalah tipe data yang bernilai true dan false.

Contoh :

```
<?php
$penyakitngantuk = false;
$lulusujian = true;
?>
```

E. Tipe data **Array**

Array adalah tipe data yang berisi sekumpulan data.

Contoh :

```
<?php
$minuman = array("Kopi Pancong", "Teh Es Manis", "Jus Jeruk");
$makanan = ["Nasi Goreng", "Ayam Geprek", "Bubur Subi bang raje"];
?>
```

F. Tipe data **Objek**

Tipe data objek adalah tipe data abstrak yang berisi data method.

Contoh :

```
<?php
$user = new User();
?>
```

Tipe data objek lebih sering disebut instance dari sebuah class.

Contoh di atas user () adalah class yang di-instance di variable \$user.

G. Tipe data **Null**

Tipe data null adalah tipe data yang menyatakan kosong. Maksudnya adalah jika kamu mengisi variable dengan nilai null, maka variable tersebut akan dianggap kosong atau tidak punya nilai.

Contoh :

```
<?php
$kelas = NULL;
?>
```

1.2.7. MANIPULASI STRING

a. Menghitung jumlah karakter dengan strlen()

untuk menghitung jumlah dari sebuah string dapat menggunakan fungsi strlen(). untuk contoh dan cara penulisan dari fungsi strlen untuk menghitung jumlah karakter di string php dapat di perhatikan pada contoh berikut ini :

```
<?php
$kalimat = "Belajar pemrograman";
echo strlen($kalimat);
?>
```

b. Menghitung jumlah kata pada String dengan str_word_count()

untuk menghitung jumlah kata yang terdapat pada sebuah string bisa menggunakan fungsi yang sudah di sediakan oleh php yaitu fungsi str_word_count(). berikut adalah cara penulisan fungsi str_word_count() untuk menghitung jumlah kata pada sebuah string.

```
<?php
$kalimat = "Belajar pemrograman";
echo str_word_count($kalimat);
?>
```

c. Membalik String Dengan strrev()

Fungsi strrev digunakan untuk membalikkan urutan karakter pada string. langsung saja untuk cara penulisan fungsi strrev() pada php seperti contoh berikut ini:

```
<?php
    $kalimat = "Belajar pemrograman di malasngoding";
    echo strrev($kalimat);
?>
```

d. Memfilter dan mengganti string dengan str_replace()

fungsi str_replace berfungsi sebagai pengganti string. misalnya di dalam sebuah string terdapat kata atau huruf yang ingin di ganti dengan kata lain maka bisa menggunakan fungsi str_replace(). untuk format penulisan nya seperti contoh berikut ini.

```
str_replace("yang ingin di ganti","pengganti","isi string");
```

dapat di lihat pada contoh penulisan fungsi str_replace() php di atas bahwa fungsi str_replace memiliki 3 parameter. parameter pertama adalah kata yang ingin di ganti, parameter kedua adalah kata yang digunakan sebagai pengganti, dan parameter ketiga adalah isi string. untuk contoh penggunaannya perhatikan pada contoh di bawah ini.

```
<?php
    $kalimat = "Belajar pemrograman ";
    echo str_replace("Belajar","Tutorial",$kalimat);
?>
```

1.2.8. PERULANGAN PADA PHP

a. 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!
    }
?>
```

Variabel \$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>";
    }
?>
```

b. 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++;
    }
?>
```

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.

c. 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);
?>
```

d. Perulangan Foreach

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

```
<?php
    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>";
?>
```

1.2.9. Operator-operator iterasi

Iterasi memiliki dua jenis, iterasi penambahan dan iterasi pengurangan. operator iterasi penambahan digunakan untuk menambah nilai variabel, sedangkan operator iterasi pengurangan digunakan untuk mengurangi nilai variabel.

Operator	Nama	Keterangan
++\$x	Pre-increment	Menambah nilai x dengan satu, kemudian kembalikan nilai x
\$x++	Post-increment	Kembalikan nilai x, kemudian tambah nilai x dengan satu
--\$x	Pre-decrement	Mengurangi nilai x dengan satu, kemudian kembalikan nilai x
\$x--	Post-decrement	Kembalikan nilai x, kemudian kurangi nilai x dengan satu

1.2.10. FUNCTION PADA PHP

Fungsi adalah sekumpulan intruksi yang dibungkus dalam sebuah blok. Fungsi dapat digunakan ulang tanpa harus menulis ulang instruksi di dalamnya. Fungsi pada PHP dapat dibuat dngan kata kunci function, lalu diikuti dengan nama fungsinya.

Contoh :

```
function namaFungsi(){  
    //...  
}
```

Kode intruksi dapat di tulis di dalam kurung kurawal ({...}).

Contoh :

```
function perkenalan(){  
    echo "Assalamulaikmu, ";  
    echo "Perkenalkan, nama saya Naruto<br/>";  
    echo "Senang berkenalan dengan anda<br/>";  
}
```

Fungsi yang sudah dibuat tidak akan menghasilkan apapun kalau tidak dipanggil. Kita dapat memanggil fungsi dengan menuliskan namanya.

Contoh :

```
Pekenalan();
```

Jadi, kode lengkapnya seperti ini:

```
<?php  
    // mmbuat fungsi  
    function perkenalan(){  
        echo "Assalamulaikmu, ";  
        echo "Perkenalkan, nama saya Ardianta<br/>";  
        echo "Senang berkenalan dengan anda<br/>";  
    }  
  
    // memanggil fungsi yang sudah dibuat  
    perkenalan();  
  
    echo "<hr>";  
  
    // memanggilnya lagi  
    perkenalan();  
?>
```

BAB II

PEMBAHASAN

pembahasan berdasarkan materi soal yang ada pada modul Pemograman Web adalah sebagai berikut :

1. Nama-nama disimpan didalam variabel array.

```
$family=["Aditya Ananda Rama","Muhammad Zihan","Ella Mariani","Venicia Putri"];
```

Pada bagian ini nama-nama dari keluarga disimpan pada variable array dengan nama variable yaitu “family” yang menyimpan nama- nama yaitu :

- a. Aditya Ananda Rama
- b. Muhammad Zihan
- c. Ella Mariani
- d. Venicia Putri

2. Program menampilkan jumlah kata dan jumlah huruf pada nama yang dituliskan.

```
foreach ($family as $nama) {  
    echo "<br>Jumlah kata dari nama ".$nama." :  
    ".str_word_count($nama)."<br>";  
}
```

Pada bagian ini menggunakan perulangan foreach karena perulangan foreach hanya bekerja pada array, dan digunakan untuk perulangan melalui setiap nilai pada array. Pada perulangan foreach ini dimasukan array \$family yang kemudian menjadi \$nama yang merupakan ‘variabel perantara’ yang berisi data array pada perulangan tersebut. Didalam perulangan foreach terdapat echo yang berfungsi untuk menampilkan string “jumlah kata dari nama”, kemudian menampilkan data array dengan menggunakan ‘variabel perantara’ yaitu \$nama dan setelah itu menampilkan str_word_count(\$nama). Kegunaan dari str_word_count adalah untuk menghitung jumlah kata yang terdapat pada sebuah string.

```
foreach ($family as $nama) {  
    $split = str_replace(" ", "", $nama);  
}
```

```
echo "<br>Jumlah huruf dari nama ".$nama"." :  
".strlen($nama)."<br>";  
}
```

Pada bagian ini juga menggunakan perluangan foreach karena perulangan foreach hanya bekerja pada array, dan digunakan untuk perulangan melalui setiap nilai pada array. Pada perulangan foreach ini dimasukan array \$family yang kemudian menjadi \$nama yang merupakan ‘variabel perantara’ yang berisi data array pada perulangan tersebut. Didalam perulangan terdapat variable \$split yang memiliki nilai str_replace(" ","", \$nama);. Fungsi dari str_replace adalah berfungsi sebagai pengganti. Pada bagian **\$split = str_replace(" ","", \$nama);** digunakan untuk menghilangkan spasi dengan cara memasukan pada parameter pertama kata yang ingin diganti “(spasi), parameter kedua adalah kata yang digunakan sebagai pengganti “”(tidak ada spasi), dan parameter ketiga adalah isi string yaitu \$nama. Selanjutnya terdapat echo yang berfungsi untuk menampilkan string “jumlah kata dari nama”, kemudian menampilkan data array dengan menggunakan ‘variabel perantara’ yaitu \$nama dan setelah itu menampilkan strlen(\$nama). Strlen berfungsi untuk menghitung jumlah karakter dari sebuah string.

3. Program menampilkan kebalikan dari nama yang diinputkan.

```
foreach ($family as $nama) {  
echo "<br>Jumlah kebalikan dari nama ".$nama"." :  
".strrev($nama)."<br>";  
}
```

Pada bagian ini juga menggunakan perluangan foreach karena perulangan foreach hanya bekerja pada array, dan digunakan untuk perulangan melalui setiap nilai pada array. Pada perulangan foreach ini dimasukan array \$family yang kemudian menjadi \$nama yang merupakan ‘variabel perantara’ yang berisi data array pada perulangan tersebut. Didalam perulangan foreach terdapat echo yang berfungsi untuk menampilkan string “jumlah kata dari nama”, kemudian menampilkan data array dengan menggunakan ‘variabel perantara’ yaitu \$nama dan setelah itu

menampilkan `strrev($nama)`. Fungsi dari `strrev` adalah untuk membalikkan urutan karakter pada string.

4. Program menampilkan jumlah konsonan dan jumlah vokal pada nama yang telah diinputkan.

```
function vocal_kosonan($word){
    $word_len = strlen($word) - 1;
    $vocal_words = array("a","e","i","o","u");

    $vocal = 0;
    $kosonan = 0;

    for($i = 0; $i <= $word_len; $i++){
        if(in_array($word[$i], $vocal_words)){
            $vocal++;
        }else{
            $kosonan++;
        }
    }
    echo "<br>jumlah huruf vocal $word : ".$vocal."<br>";
    echo "<br>jumlah huruf kosonan $word : ".$kosonan."<br>";
}
foreach($family as $data){
    vocal_kosonan($data);
}
```

Pada bagian ini menggunakan Fungsi (atau Function) dalam bahasa pemrograman adalah kode program yang dirancang untuk menyelesaikan sebuah tugas tertentu, dan merupakan bagian dari program utama. function dideklarasikan sebagai `vocal_kosonan` dengan parameter variable `$word`. Didalam function terdapat variable `$word_len` yang memiliki nilai `strlen($word) - 1`; `Strlen` berfungsi untuk menghitung jumlah dari sebuah string. variable `$vocal` yang memiliki array, didalamnya terdapat nilai yaitu a, i, u, e, dan o. selanjutnya terdapat variable `vocal` dan `kosonan` yang memiliki nilai 0. Berikutnya terdapat perulangan `for` yang memiliki parameter nilai `i` dengan nilai 0, jika nilai variable `i` kurang dari atau sama dengan variable `word_len` maka nilai `i` di increment. Di dalam perulangan `for` terdapat pengkondisian `if` dengan parameter `in_array($word[$i], $vocal_words)`. `in_array` merupakan sebuah fungsi untuk memeriksa suatu

nilai yang terdapat di dalam sebuah array. Maka pengkondisian if ini akan berjalan jika nilai array pada array \$word sama dengan nilai yang dicari oleh \$vocal_words. Jika pengkondisian if ini terpenuhi maka nilai dari vocal akan increment dan jika tidak maka nilai kosonan akan increment. Dan yang terakhir pada bagi function adalah terdapat echo yang berfungsi untuk menampilkan string “jumlah huruf vocal”, kemudian menampilkan data array dengan menggunakan parameter yang dimasukan yaitu \$word dan setelah itu menampilkan \$vocal. Berikut nya echo yang berfungsi untuk menampilkan string “jumlah huruf kosonan”, kemudian menampilkan data array dengan menggunakan parameter yang dimasukan yaitu \$word dan setelah itu menampilkan \$kosonan.

Kemudian setelah function selesai, maka dilakukan pemanggilan function menggunakan foreach. Pada perulangan foreach ini dimasukan array \$family yang kemudian menjadi \$data yang merupakan ‘variabel perantara’ yang berisi data array pada perulangan tersebut. Pada bagian dalam foreach terdapat pemanggilan function vocal_kosonan dengan parameter masukannya adalah variable \$data.

Berikut merupakan keseluruhan bagian dari soal jika digabungkan pada sebuah coding :

```
<?php
$family=["Aditya Ananda Rama","Muhammad Zihan","Ella
Mariani","Venicia Putri"];

foreach ($family as $nama) {
    echo "<br>Jumlah kata dari nama ".$nama." :
    ".str_word_count($nama)."<br>";
}

foreach ($family as $nama) {
    $split = str_replace(" ","", $nama);
    echo "<br>Jumlah huruf dari nama ".$nama." :
    ".strlen($nama)."<br>";
}

foreach ($family as $nama) {
    echo "<br>Jumlah kebalikan dari nama ".$nama." :
    ".strrev($nama)."<br>";
}

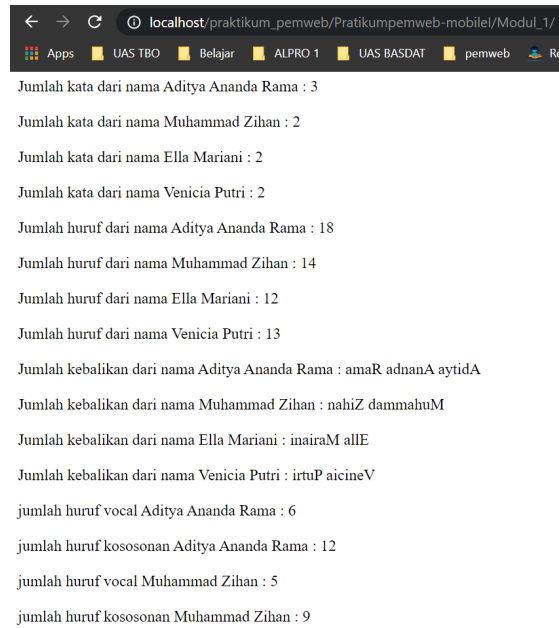
function vocal_kosonan($word){
    $word_len = strlen($word) - 1;
    $vocal_words = array("a","e","i","o","u");

    $vocal = 0;
    $kosonan = 0;

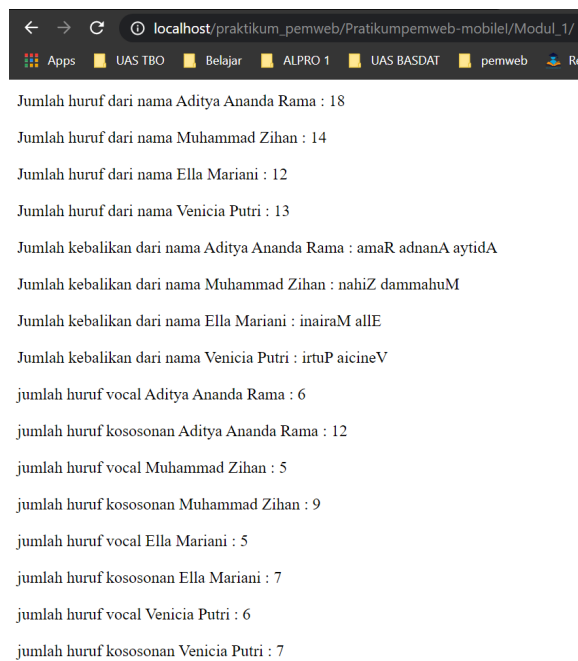
    for($i = 0; $i <= $word_len; $i++){
        if(in_array($word[$i], $vocal_words)){
            $vocal++;
        }else{
            $kosonan++;
        }
    }
    echo "<br>jumlah huruf vocal $word : ".$vocal."<br>";
    echo "<br>jumlah huruf kosonan $word : ".$kosonan."<br>";
}
foreach($family as $data){
    vocal_kosonan($data);
}

?>
```

Berikut merupakan output dari PHP :



Gambar 2.1 Output PHP (1)



Gambar 2.2 Output PHP (2)

BAB III

KESIMPULAN

PHP adalah bahasa pemrograman server side karena PHP diproses pada komputer server. Ketika dipanggil dari web browser, program yang ditulis dengan PHP akan di-parsing di dalam web server oleh interpreter PHP dan diterjemahkan ke dalam dokumen HTML, yang selanjutnya akan ditampilkan kembali ke web browser. pada program yang dibuta menggunakan Perulangan foreach yang sama seperti perulangan for. Namun, ia lebih khusus digunakan untuk mencetak array. Dan juga menggunakan Fungsi yang merupakan sekumpulan intruksi yang dibungkus dalam sebuah blok.

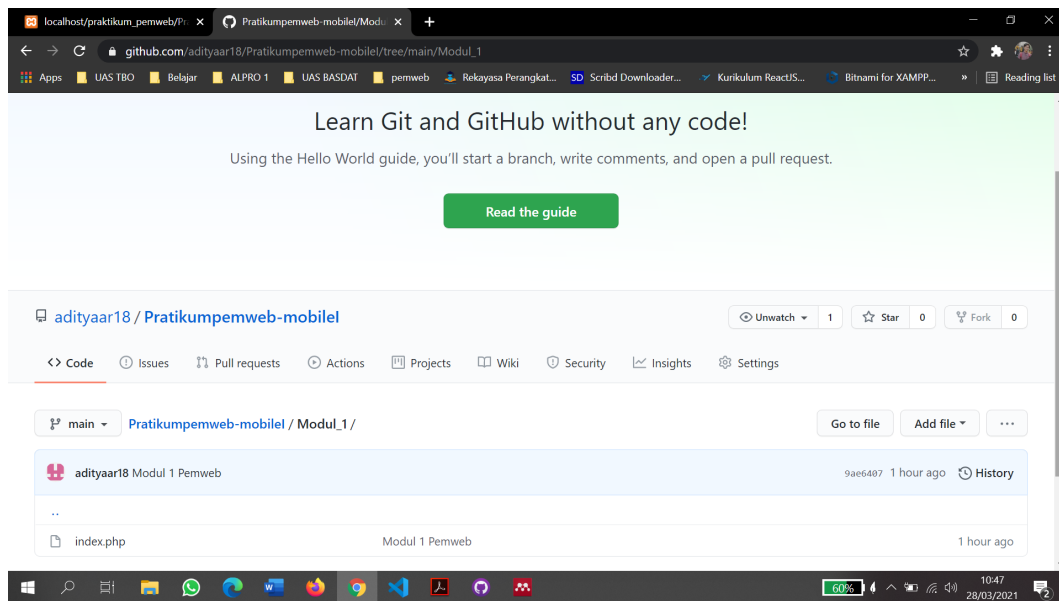
DAFTAR PUSTAKA

- Alfarabi H., Diki. *Belajar PHP Part 6 : Manipulasi String Pada PHP*.
<https://www.malasngoding.com/belajar-php-manipulasi-string-pada-php/>
(Diakses pada 27 April 2021)
- Andre. (2014). *Tutorial Belajar PHP Part 39: Pengertian Fungsi (function) PHP dan Cara Penggunaan Fungsi PHP*.
<https://www.duniailkom.com/tutorial-belajar-php-pengertian-fungsi-function-php-dan-cara-penggunaan-fungsi-php/> (Diakses pada 27 April 2021)
- Erawan, L. (2014). *Dasar-Dasar PHP. Udinus*, 1–47.
http://dinus.ac.id/repository/docs/ajar/1-PHP_DASAR-DASAR.pdf (Diakses pada 27 April 2021)
- Jayanti, S., & Raja, J. (2020). *Jago Ngoding Pemrograman Web dengan PHP* (Issue January).
- Muhammad, Ahmad. (2017). *Belajar PHP: Memahami Prosedur dan Fungsi*.
<https://www.petanikode.com/php-prosedur-fungsi/> (Diakses pada 27 April 2021)
- Muhammad, Ahmad. (2015). *Belajar PHP: Memahami 4 Jenis Perulangan dalam Pemrograman PHP*.
<https://www.petanikode.com/php-perulangan/> (Diakses pada 27 April 2021)
- Muhammad, Ahmad. (2018). *Belajar PHP: Perbedaan Fungsi echo(), print(), dan printf() pada PHP*.
<https://www.petanikode.com/php-echo-print/> (Diakses pada 27 April 2021)

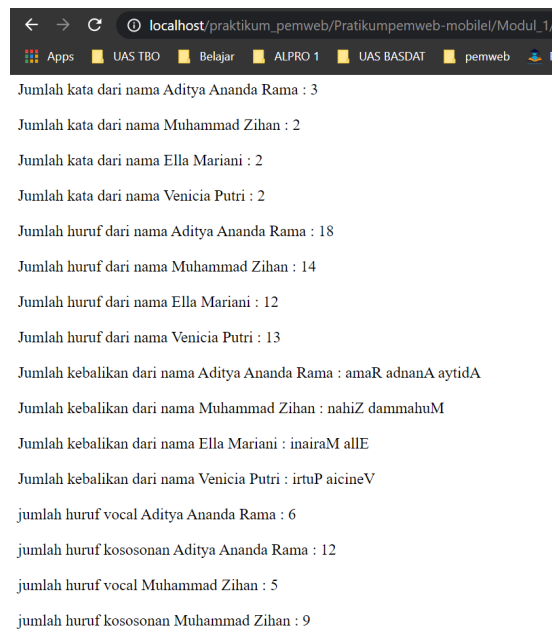
Tim Dosen Algoritma Pemrograman. *Modul Praktikum Pemrograman Web I*.
2021. Palangka Raya. Jurusan Teknik Informatika Fakultas Teknik
Universitas Palangka Raya (UPR).

Yuana, R. A. (2015). *Belajar Dasar PHP*. 1–25.

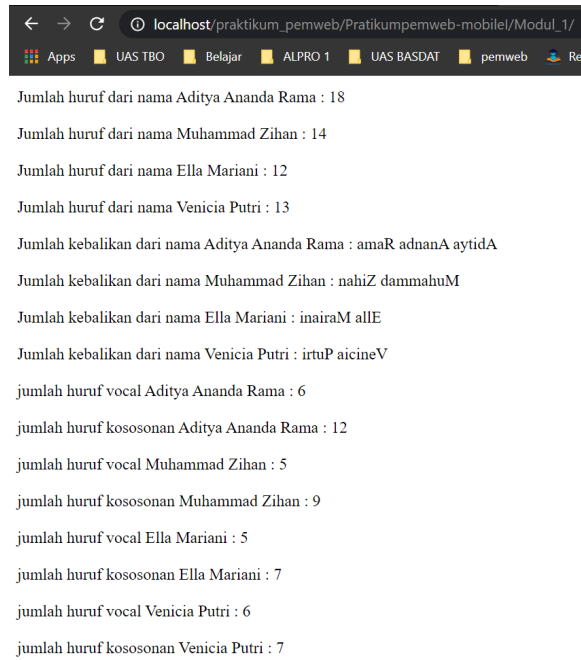
LAMPIRAN



<https://github.com/adityaar18/Pratikumpemweb-mobile>



Gambar 2.1 Output PHP (1)



Gambar 2.2 Output PHP (2)