**LAPORAN PRAKTIKUM**

PHP LANJUT



Nama : Aqilah Akmalia Dewi

Kelas : 1 D4 IT B

NRP : 3120600046

PROGRAM STUDI D4 TEKNIK INFORMATIKA

POLITEKNIK ELEKTRONIKA NEGERI SURABAYA

2021

DAFTAR ISI

[PERCOBAAN 3](#_Toc66194858)

[1. PHP Condition 3](#_Toc66194859)

[2. Looping 6](#_Toc66194860)

[3. PHP Function 8](#_Toc66194861)

[4. PHP Arrays 10](#_Toc66194862)

[5. PHP Array Sorting 14](#_Toc66194863)

[6. PHP Date and Time 16](#_Toc66194864)

[TUGAS 18](#_Toc66194865)

# PERCOBAAN

## PHP Condition

1. Listing Program

<!DOCTYPE html>

<html>

<body>

<?php

$t = date("H");

if ($t < "20") {

    echo "Have a good day!";

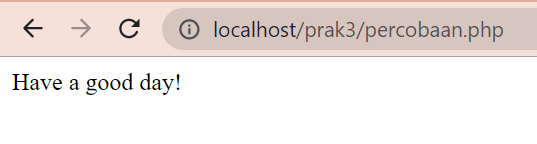
}

?>

</body>

</html>

1. Output



Analisa : Fungsi date digunakan untuk menampilkan data tanggal sesuai dengan format yang diinginkan. “H” berarti Jam dengan format 24-jam, dengan awalahn 0 (dua digit jam) 00 s.d 23

1. Listing Program

<!DOCTYPE html>

<html>

<body>

<?php

$t = date("H");

if ($t > "20") {

    echo "Have a good day!";

} else {

    echo "Have a good night!";

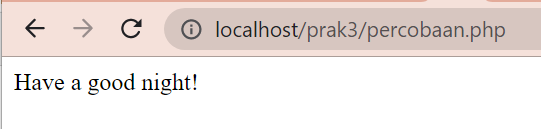
}

?>

</body>

</html>

1. Output



Analisa : Terdapat 2 kondisi, jika kondisi benar maka akan tercetak have a good day, namun jika kondisi salah maka akan tercetak have a good night

1. Listing Program

<!DOCTYPE html>

<html>

<body>

<?php

$t = date("H");

echo "<p>Waktu di Server Komputer adalah " . $t;

echo ", Pesan untuk anda :<p>";

if ($t < "10") {

    echo "Selamat Pagi";

} elseif ($t < "14") {

    echo "Selamat Siang";

} elseif ($t < "18") {

    echo "Selamat Sore";

} else {

    echo "Have a good night!";

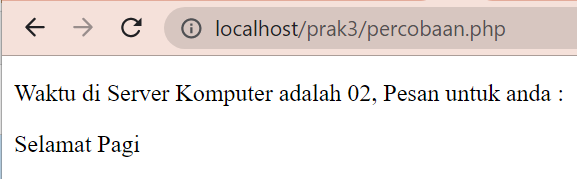
}

?>

</body>

</html>

1. Output



Analisa : Jika ada kondisi banyak (lebih dari 2) maka bisa menggunakan else if. Jadi jika digabungkan, ketiga percobaan ini menggunakan function date(“H”) untuk mengetahui waktu di server computer dan menggunakan decision if, else if dan else untuk menentukan kondisi dan output yang akan ditampilkan.

1. Listing Program

<!DOCTYPE html>

<html>

<body>

<?php

$favcolor = "red";

switch ($favcolor) {

    case "red" :

        echo "<p style='color: red'> Warna terpilih MERAH! </p>";

        break;

    case "blue":

        echo "<p style=’color: red;’> Warna terpilih BIRU! </p>";

        break;

    case "green":

        echo "<p style=’color: red;’> Warna terpilih HIJAU! </p>";

        break;

    default:

        echo "Coba lagi!";

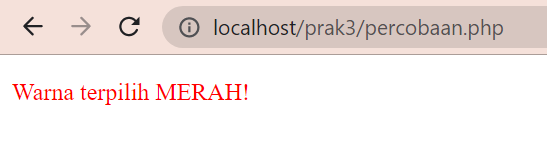
}

?>

</body>

</html>

1. Output



Analisa : Selain menggunakan else if, kita juga bisa menggunakan switch case untuk kondisi lebih dari 2

## Looping

1. Listing Program

<!DOCTYPE html>

<html>

<body>

<?php

$x = 1;

while ($x <= 5) {

    echo "Angka yang dimasukkan : $x <br>";

    $x++;

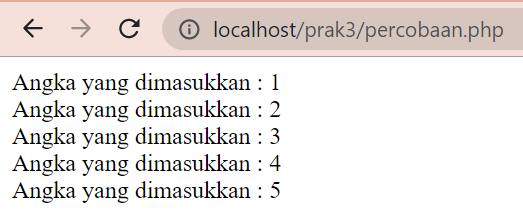
}

?>

</body>

</html>

1. Output



Analisa : pada awal percobaan ini terdapat pendeklarasian variable yang sudah diinisialisasi. Pada percobaan ini juga terdapat perulangan/looping yaitu while dimana dia akan teus melakukan perulangan apabila nilai dari variable yang di deklarasikan diawal kurang dari sama dengan 5.

1. Listing Program

<!DOCTYPE html>

<html>

<body>

<?php

$x = 6;

do {

    echo " Angka : $x <br>";

    $x++;

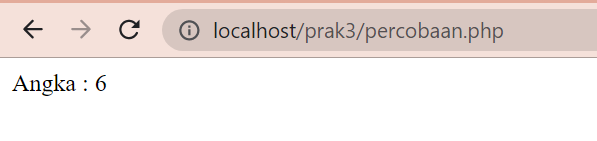
} while ($x <= 5);

?>

</body>

</html>

1. Output



Analisa : Pada percobaan ini menggunakan perulang do while. Dimana pada perulangan ini berbeda dengan while, jikawhile melakukan pengecekan kondisi terlebih dahulu, pada do while dilakukan eksekusi sebanyak sekali terlebih dahulu, lalu dilakukan pengecekan kondisi. Apabila kondisi tersebut bernilai true maka, perulangan akan terus berlanjut, jika kondisi bernilai false, maka perulangan tidak dilanjutkan.

1. Listing Program

<!DOCTYPE html>

<html>

<body>

<?php

for ($x = 0; $x <= 10; $x++) {

   echo "Angka : $x <br>";

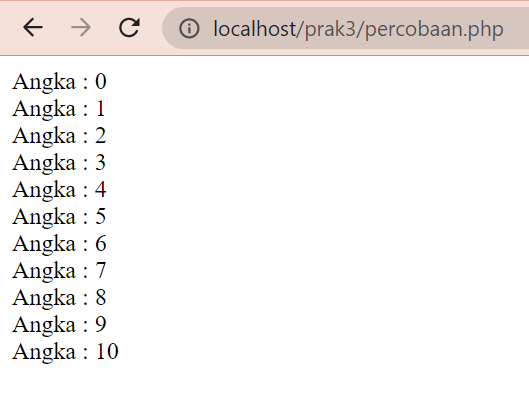
}

?>

</body>

</html>

1. Output



Analisa : pada pecobaan ini terdapat perulangan yang menggunakan for. Dimana perulangan tersebut dimulai dari 0 dan berhenti apabila >= 10.

1. Listing Program

<!DOCTYPE html>

<html>

<body>

<?php

$colors = array("merah", "hijau", "biru", "kuning");

foreach ($colors as $value) {

   echo "$value <br>";

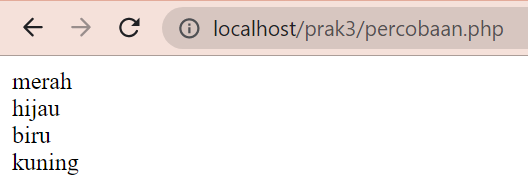
}

?>

</body>

</html>

1. Output



Analisa : pada percobaan ini, terdapat variable yang menampung data array. Kemudian untuk menampilkan data array tersebut, digunakan perulangan foreach.

## PHP Function

1. Listing Program

<!DOCTYPE html>

<html>

<body>

<?php

function writeMsg() {

    echo "Hello world!";

}

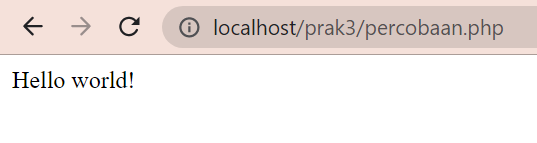
writeMsg();

?>

</body>

</html>

1. Output



Analisa : pada percobaan ini, terdapat function bernama writeMsg yang didalamnya mencetak kalimat “Hello World!”. Maka dari itu, ketika function tersebut dipanggil akan menghasilkan output seperti pada gambar

1. Listing Program

<!DOCTYPE html>

<html>

<body>

<?php

function familyName($fname) {

    echo "$fname Refsnes.<br>";

}

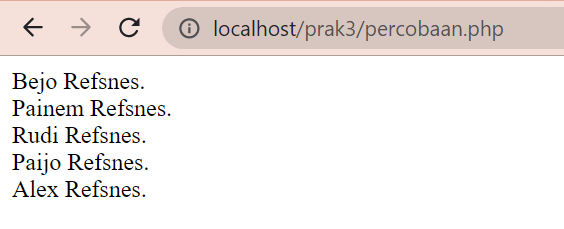
familyName("Bejo"); familyName("Painem"); familyName("Rudi"); familyName("Paijo"); familyName("Alex");

?>

</body>

</html>

1. Output



Analisa : pada percobaan ini terdapat function bernama familyName. Pada function ini membutuhkan sebuah parameter ketika dipanggil. Parameter ini digunakan sebagai bahan untuk menjalankan program yang ada di dalam function tersebut. Bejo, Palinem, Rudi, Paijo dan Alex merupakan parameter yang dikirim pada saat pemanggilan function familyName().

1. Listing Program

<!DOCTYPE html>

<html>

<body>

<?php

function setHeight($minheight = 50) {

    echo "The height is : $minheight <br>";

}

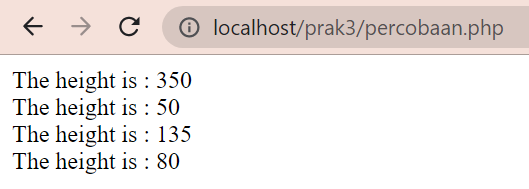
setHeight(350); setHeight(); setHeight(135); setHeight(80);

?>

</body>

</html>

1. Output



Analisa : pada percobaan ini terdapat function yang bernama setHeight dan juga parameternya yang sudah diberi nilai 50. Tujuan pemberian nilai pada parameter ini adalah apabila ketika memanggil function tidak diberi parameter, maka 50 inilah yang akan menjadi parameter nya. Output pada baris kedua merupakan contoh pemanggilan function yang tidak diberi parameter sehingga parameter otomatis bernilai 50.

## PHP Arrays

1. Listing Program

<!DOCTYPE html>

<html>

<body>

<?php

$cars = array("Volvo", "BMW", "Toyota");

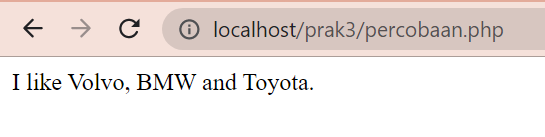
echo "I like " . $cars[0] . ", " . $cars[1] . " and " . $cars[2] . ".";

?>

</body>

</html>

1. Output



Analisa : pada percobaan ini terdapat variable yang menyimpan data array. Untuk pemanggilan data arraynya dengan menggunakan cara yaitu menuliskan index dari data yang ingin dipanggil.

1. Listing Program

<!DOCTYPE html>

<html>

<body>

<?php

$cars = array("Volvo", "BMW", "Toyota");

$arrlength = count($cars);

for($x = 0; $x < $arrlength; $x++) {

    echo $cars[$x];

    echo "<br>";

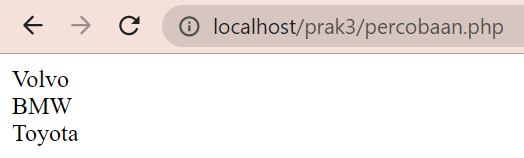
}

?>

</body>

</html>

1. Output



Analisa : pada percobaan ini terdapat variable yang menyimpan data array. Untuk memanggil seluruh data yang ada, pada percobaan ini menggunakan perulangan for untuk mengulang indeks. Perulangan akan terus dilakukan apabila indeks < jumlah data.

1. Listing Program

<!DOCTYPE html>

<html>

<body>

<?php

$age = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");

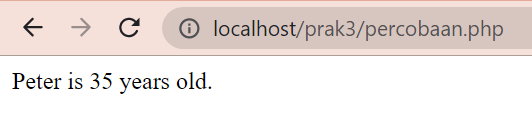
echo "Peter is " . $age['Peter'] . " years old.";

?>

</body>

</html>

1. Output



Analisa : pada percobaan ini terdapat variable yang menyimpan data array. Array yang digunakan adalah Associative array dimana index yang digunakan bukan berupa angka. Untuk pemanggilan data array dilakukan dengan cara memanggil variable disertai dengan index yang ingin dipanggil. Pada percobaan yang dipanggil adalah data yang berindeks “peter”.

1. Listing Program

<!DOCTYPE html>

<html>

<body>

<?php

$age = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");

foreach($age as $x => $x\_value) {

    echo "Key=" . $x . ", Value=" . $x\_value;

    echo "<br>";

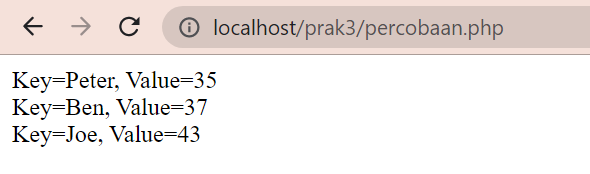
}

?>

</body>

</html>

1. Output



Analisa : pada percobaan ini terdapat variable yang menyimpan data array. Array yang digunakan adalah Associative array dimana index yang digunakan bukan berupa angka.

Untuk pemanggilan seluruh data array dilakukan dengan cara melakukan perulangan foreach. Pada percobaan ini key dari array dijadikan variabel $x, dan value dijadikan variabel $ x\_value, sehingga jika kita ingin menampilkan nama key dan value, kita tinggal menggunakan variabel $ x dan $ x\_value.Listing Program

<!DOCTYPE html>

<html>

<body>

<?php

$cars = array

   (

   array("Volvo",22,18),

   array("BMW",15,13),

   array("Saab",5,2),

   array("Land Rover",17,15)

   );

echo $cars[0][0].": In stock: ".$cars[0][1].", sold: ".$cars[0][2].".<br>";

echo $cars[1][0].": In stock: ".$cars[1][1].", sold:

".$cars[1][2].".<br>";

echo $cars[2][0].": In stock: ".$cars[2][1].", sold: ".$cars[2][2].".<br>";

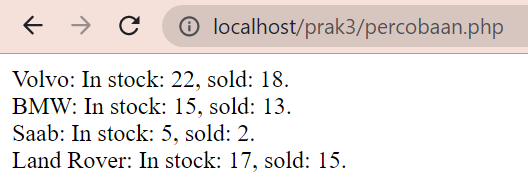
echo $cars[3][0].": In stock: ".$cars[3][1].", sold: ".$cars[3][2].".<br>";

?>

</body>

</html>

1. Output



Analisa : pada percobaan ini terdapat variable yang menyimpan data array. Array yang digunakan adalah Multidimensional Array dimana kita bisa membuat array didalam array. Untuk pemanggilan datanya, kita hanya perlu menuliskan variable disertai indeks pertama dan keduanya.

## PHP Array Sorting

1. Listing Program

<!DOCTYPE html>

<html>

<body>

<?php

$cars = array("Volvo", "BMW", "Toyota"); rsort($cars);

$clength = count($cars);

for($x = 0; $x < $clength; $x++) {

    echo $cars[$x];

    echo "<br>";

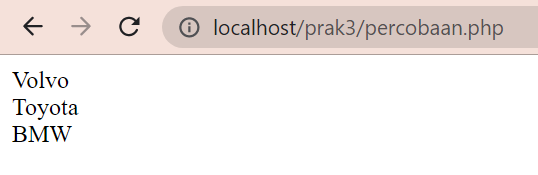
}

?>

</body>

</html>

1. Output



Analisa : pada percobaan tersebut terdapat function rsort() yaitu digunakan untuk menyortir elemen-elemen dari array yang diindeks dalam descending order. Untuk menampilkan seluruh datanya menggunakan looping for, dimana looping akan berhenti ketika indeks >= jumlah data.

1. Listing Program

<!DOCTYPE html>

<html>

<body>

<?php

$numbers = array(4, 6, 2, 22, 11);

sort($numbers);

$arrlength = count($numbers);

for($x = 0; $x < $arrlength; $x++) {

    echo $numbers[$x];

    echo "<br>";

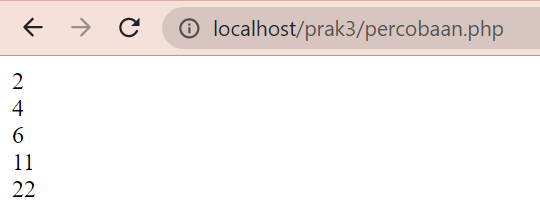
}

?>

</body>

</html>

1. Output



Analisa : pada percobaan tersebut terdapat function sort() yaitu digunakan untuk menyortir elemen-elemen dari array yang diindeks dalam ascending order. Untuk menampilkan seluruh datanya menggunakan looping for, dimana looping akan berhenti ketika indeks >= jumlah data.

1. Listing Program

<!DOCTYPE html>

<html>

<body>

<?php

$numbers = array(4, 6, 2, 22, 11); rsort($numbers);

$arrlength = count($numbers);

for($x = 0; $x < $arrlength; $x++) {

    echo $numbers[$x];

    echo "<br>";

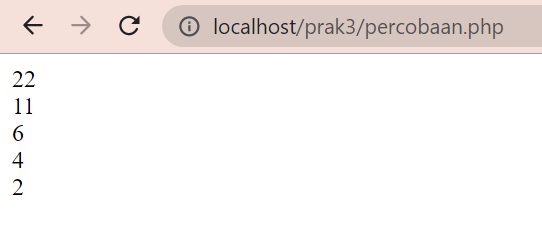
}

?>

</body>

</html>

1. Output



Analisa : pada percobaan tersebut terdapat function rsort() yaitu digunakan untuk menyortir elemen-elemen dari array yang diindeks dalam descending order. Untuk menampilkan seluruh datanya menggunakan looping for, dimana looping akan berhenti ketika indeks >= jumlah data.

## PHP Date and Time

1. Listing Program

<!DOCTYPE html>

<html>

<body>

<?php

echo "Tanggal hari ini " . date("Y/m/d") . "<br>";

echo "Tanggal hari ini " . date("Y.m.d") . "<br>";

echo "Tanggal hari ini " . date("Y-m-d") . "<br>";

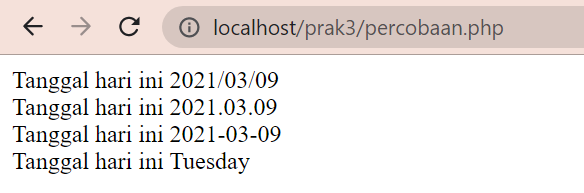
echo "Tanggal hari ini " . date("l");

?>

</body>

</html>

1. Output



Analisa : Fungsi date digunakan untuk menampilkan data tanggal sesuai dengan format yang diinginkan, Y berarti tahun, m berarti bulan, d berarti tanggal, dan l berarti hari

1. Listing Program

<!DOCTYPE html>

<html>

<body>

<?php date\_default\_timezone\_set("Asia/Jakarta");

echo "Sekarang waktu Jakarta : " . date("h:i:sa"); echo "<br/>"; date\_default\_timezone\_set("Asia/Tokyo");

echo "Sekarang waktu Tokyo : " . date("h:i:sa"); echo "<br/>"; date\_default\_timezone\_set("America/New\_York");

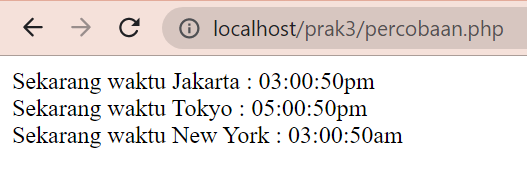
echo "Sekarang waktu New York : " . date("h:i:sa");

?>

</body>

</html>

1. Output



Analisa : Fungsi date digunakan untuk menampilkan data tanggal sesuai dengan format yang diinginkan, h berarti jam, i berarti menit, dan s berarti detik

# TUGAS

1. Mencari jumlah huruf vocal pada suatu kata
2. Listing Program

<?php

$hitung = 0;

$inputan = 'Selasa';

echo "Jumlah huruf vokal dari kata 'Selasa' adalah ";

for($i = 0; $i < strlen($inputan); $i++){

    if($inputan[$i] == 'A' || $inputan[$i] == 'a'){

        $hitung++;

    }

    else if($inputan[$i] == 'I' || $inputan[$i] == 'i'){

        $hitung++;

    }

    else if($inputan[$i] == 'U' || $inputan[$i] == 'u'){

        $hitung++;

    }

    else if($inputan[$i] == 'E' || $inputan[$i] == 'e'){

        $hitung++;

    }

    else if($inputan[$i] == 'O' || $inputan[$i] == 'o'){

        $hitung++;

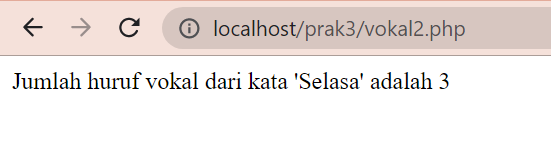
    }

}

echo $hitung;

?>

1. Output



Analisa : Pada program ini saya langsung mendeklarasikan inputan di awal, kemudian menggunakan looping for untuk mengecek berapakah huruf vocal yan terdapat pada kata tersebut. Loping dimulai dari 0 dan berhenti sebelum sebanyak jumlah huruf inputan. Pada looping tersebut di dalamnya terdapat 5 kondisi menggunakan else if. Pengecekan berjalan dimulai dari huruf pertama, apabila huruf tsb termasuk kondisi yang dimaksudkan maka $hitung akan bertambah 1, begitu pula setersunya sampai huruf dari inputan selesai dicek.

1. Membuat fungsi untuk memvalidasi inputan yang dimasukkan harus angka
2. Listing Program

<?php

$hitung = 0;

$huruf = 0;

$inputan = "yukbisa";

echo "Hasil perhitungan input adalah : <br>";

for($i = 0; $i < strlen($inputan); $i++){

    if($inputan[$i] == '0'){

        $hitung++;

    }

    else if($inputan[$i] == '1'){

        $hitung++;

    }

    else if($inputan[$i] == '2'){

        $hitung++;

    }

    else if($inputan[$i] == '3'){

        $hitung++;

    }

    else if($inputan[$i] == '4'){

        $hitung++;

    }

    else if($inputan[$i] == '5'){

        $hitung++;

    }

    else if($inputan[$i] == '6'){

        $hitung++;

    }

    else if($inputan[$i] == '7'){

        $hitung++;

    }

    else if($inputan[$i] == '8'){

        $hitung++;

    }

    else if($inputan[$i] == '9'){

        $hitung++;

    }

    else

        $huruf++;

}

if($hitung == 0){

    echo "<br> Inputan anda belum sesuai";

    echo "<br> Huruf yang anda masukkan berjumlah " . $huruf;

}

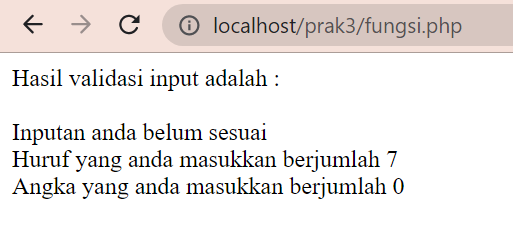
else

    echo "<br> Inputan anda sudah sesuai";

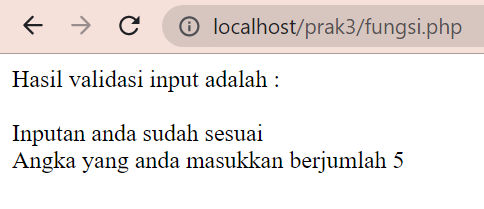
echo "<br> Angka yang anda masukkan berjumlah " . $hitung;

1. Output









Analisa : Pada program ini saya langsung mendeklarasikan inputan, kemudian menggunakan looping for untuk melakukan pengecekan inputan. Looping dimulai dari 0 dan berhenti sampai inputan selesai dicek. Di dalam looping tersebut terdapat else if, yang berisi kondisi dari angka 0-9, apabila saat pengecekan berlangsung inputan mengandung angka maka $hitung akan bertambah 1, namun apabila ternyata tidak ada angka maka $huruf yang akan bertambah. Kemudian bisa didapatkan data, jika $hitung == 0, maka otomatis yang diinputkan adalah huruf semua dan tentunya belum sesuai. Namun jika hitung != 0, maka jelas inputan adalah berupa angka.