

# **Laporan Praktikum**

## **Pemrograman Web 2**

**Dosen pengampu : Siti Maesyaroh, S.Kom., M.Kom.**

### **Modul 2**



**Nama : Rio Andika Andriansyah**

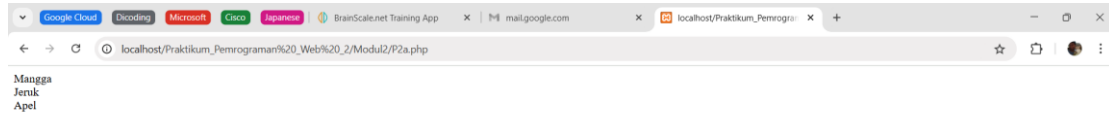
**NIM : 20230810155**

**Kelas : TINFC – 2023 – 04**

**Teknik Informatika**  
**Fakultas Ilmu Komputer**  
**Universitas Kuningan**

# Praktikum

## P2a



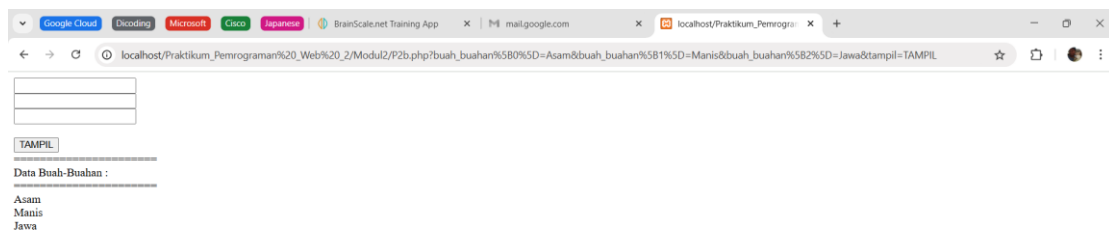
### Script Code:

```
<?php
$buah_buahan = Array("Mangga", "Jeruk", "Apel");
for($i = 0; $i <= 2; $i++)
{
    echo "$buah_buahan[$i] <br>";
}
?>
```

### Analisis:

Kode Program di atas mendemonstrasikan Penggunaan Array dan For Loop dengan mendeklarasikan array \$buah\_buahan yang berisi tiga elemen string yaitu "Mangga", "Jeruk", dan "Apel". Array tersebut secara bawaan memiliki indeks 0, 1, dan 2. Selanjutnya perulangan for digunakan dimulai dari “\$i = 0” hingga “\$i <= 2” yang berarti perulangan akan berjalan tiga kali untuk mengakses seluruh elemen array melalui indeks. Pada For Loop terdapat “echo” yang digunakan untuk menampilkan elemen array berdasarkan indeks \$i yang berjalan lalu ditambahkan tag “<br>” agar output ditampilkan ke browser secara satu per satu hingga semua elemen tertampilkan.

## P2b



### Script Code:

```
<HTML>
<BODY>
<FORM NAME="FORM1" METHOD="GET" action="P2b.php">
<?php
for($i = 0; $i <= 2; $i++) {
    echo "<input type='text' name='buah_buahan[$i]'><br>";
}
echo "<br><input type='submit' name='tampil' value='TAMPIL'>";

if (isset($_GET['buah_buahan'])) {
    $buah_buahan = $_GET['buah_buahan'];
    echo "<br>=====";
    <br> Data Buah-Buahan : <br>
    ===== <br>";
    for($i = 0; $i <= 2; $i++){
        echo "$buah_buahan[$i]<br>";
    }
}
?>
</FORM>
</BODY>
</HTML>
```

### Analisis :

Kode Program di atas mendemonstrasikan penggunaan Array yang diinput secara dinamis dengan membentuk sebuah form input menggunakan metode GET dan mengirim data ke file “P2b.php”. Di dalam tag “<FORM>” PHP digunakan untuk membuat tiga buah input secara dinamis menggunakan Loop For dari indeks 0 sampai 2. Masing-masing input yang diterima diberi “name='buah\_buahan[\$i]” yang berarti dibentuknya array “buah\_buahan[]” ketika data terkirim. Setelah input diterima akan disediakan tombol submit bernama tampil dengan label "TAMPIL". Ketika form dikirim maka data yang diinputkan user akan dikirim lewat URL menggunakan metode GET. Setelah tombol diklik dan form terkirim maka kode memeriksa array \$\_GET['buah\_buahan'] yang terisi menggunakan isset() dan data yang dikirim oleh input teks sebelumnya diambil dan disimpan ke dalam variabel “\$buah\_buahan”. Selanjutnya dilakukan Loop For dari 0 sampai 2 untuk mencetak kembali masing-masing elemen array \$buah\_buahan ke layar menggunakan “echo” untuk menampilkan kembali data buah-buahan yang diinputkan secara satu per satu di baris baru menggunakan tag “<br>”.

## P2c



## Script Code:

```
<?php
$kalimat = "Saya belajar PHP";
$kata = explode(" ", $kalimat);
$jumlahArray = count($kata);
for ($i=0; $i<$jumlahArray; $i++){
    echo "Kata[$i] = $kata[$i] <BR>";
}
$gabung = implode(" ", $kata);
echo "<BR> Hasil Penggabungan = $gabung";
?>
```

## Analisis:

Kode Program di atas mendemonstrasikan proses pemisahan string menjadi array kata dengan function “explode()” dengan mendeklarasikan variabel \$kalimat yang berisi string "Saya belajar PHP". Lalu fungsi explode(" ", \$kalimat) digunakan untuk memecah string menjadi beberapa array berdasarkan spasi sehingga menghasilkan array \$kata dengan memiliki tiga elemen yaitu "Saya", "belajar" dan "PHP". Setelah itu menggunakan fungsi count(\$kata) untuk menghitung jumlah elemen didalam array yang kemudian disimpan dalam variabel “\$jumlahArray”. Selanjutnya Loop For dilakukan dimulai dari indeks 0 sampai kurang dari “\$jumlahArray” untuk mencetak tiap elemen array dengan “format Kata[i] = isi\_kata” ke browser yang masing-masing di baris baru. Setelah seluruh kata ditampilkan fungsi implode(" ", \$kata) digunakan untuk menggabungkan kembali seluruh elemen array menjadi satu string utuh dengan spasi sebagai pemisah kemudian hasilnya ditampilkan menjadi "Hasil Penggabungan = Saya belajar PHP" ke browser.

## P2d



### Script Code:

```
<?php
function fungsi_nilai($X){
    $X = 300;
    return $X;
}
$Y = 200;
echo "Nilai Y Semula = $Y <br>";
echo "Nilai Y Sekarang = ".fungsi_nilai($Y);
?>
```

### Analisis:

Kode Program di atas mendemonstrasikan penggunaan fungsi yang memiliki parameter dengan pass-by-value didalam PHP. Fungsi “fungsi\_nilai(\$X)” dideklarasikan agar dapat menerima parameter “\$X” yang kemudian diubah nilainya menjadi “300” di fungsi tersebut. Namu, nilai asli dari variabel “\$Y” yang di luar fungsi tidak terpengaruh karena PHP menggunakan mekanisme pass-by-value yang berarti bahwa nilai yang diteruskan ke fungsi adalah salinan dari nilai asli. Pada bagian utama kode nilai awal “\$Y” adalah 200 yang ditampilkan dengan tag “echo”. Kemudian fungsi “fungsi\_nilai(\$Y)” dipanggil dan meskipun fungsi mengubah nilai “\$X” menjadi 300 nilai yang dikembalikan oleh fungsi ditampilkan dan hasilnya adalah 300 sementara nilai asli “\$Y” tetaplah 200.

### P2e



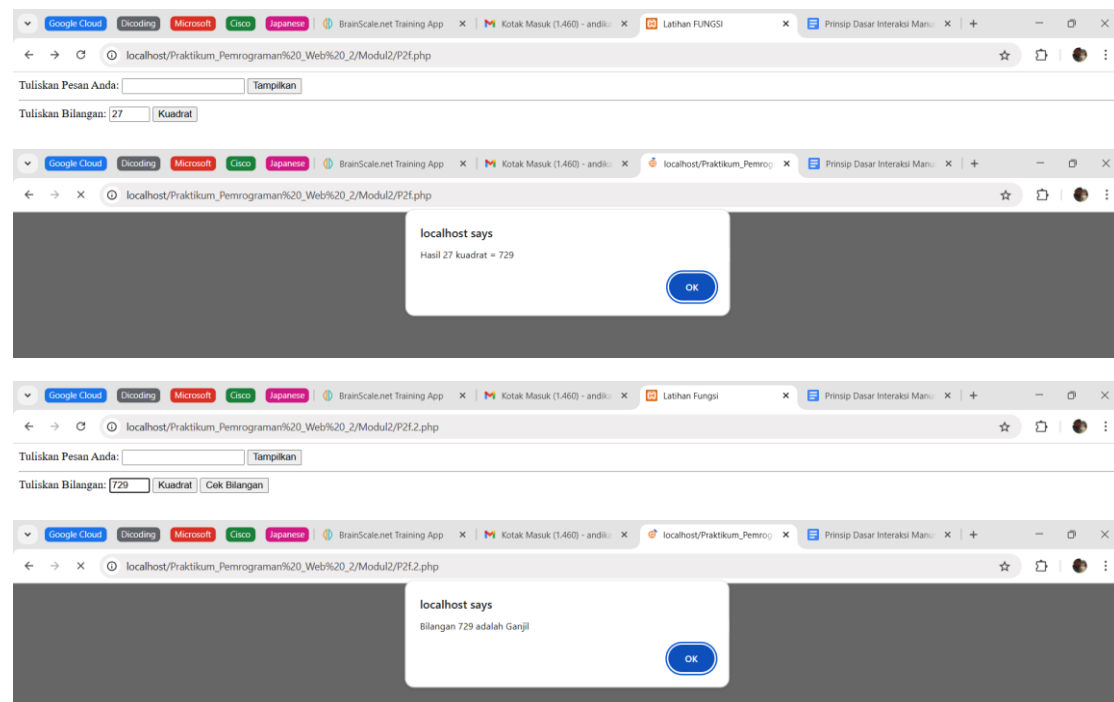
### Script Code:

```
<?php
function fungsi_1($angka){
    $angka = 300;
    return $angka;
}
function fungsi_2($angka){
    return $angka;
}
$angka = 200;
echo "Angka (variabel lokal) = ".fungsi_1($angka);
echo "<br>Angka (variabel eksternal) = ".fungsi_2($angka);
?>
```

### Analisis :

Kode Program di atas mendemonstrasikan pass-by-value dan pengaruh perubahan dalam fungsi terhadap variabel eksternal dan local dimana dua fungsi yang masing-masing menerima parameter \$angka. Fungsi pertama adalah “fungsi\_1(\$angka)” mengubah nilai parameter “\$angka” menjadi 300 di dalam fungsi. Namun perubahan tersebut hanya memengaruhi variabel lokal di dalam fungsi tersebut karena PHP menggunakan pass-by-value yang berarti bahwa hanya salinan dari nilai asli yang dikirim ke fungsi. Sehingga meskipun “\$angka” diubah menjadi 300 di dalam fungsi tetapi nilai aslinya yang berada di luar fungsi tetap tidak terpengaruh dan yang ditampilkan adalah 300. Fungsi kedua yaitu “fungsi\_2(\$angka)” hanya mengembalikan nilai parameter “\$angka” tanpa melakukan perubahan Karena tidak ada perubahan pada parameter jadi nilai yang dikembalikan sama dengan nilai awal dari variabel eksternal yaitu 200.

## P2f



### Script Code:

```
<?php
if ($_SERVER['REQUEST_METHOD'] === 'POST') { // Sebelum Modifikasi
    $submit = $_POST['submit'] ?? null;
    $pesan = $_POST['pesan'] ?? '';
    $bilangan = $_POST['bilangan'] ?? null;
    function tampilAlert($pesan) {
        echo "<script>alert('Pesan: $pesan');
        window.history.go(-1);</script>";
    }
    function kuadrat($bilangan) {
        if (!is_numeric($bilangan)) {
            echo "<script>alert('Input harus berupa angka!');
            window.history.go(-1);</script>";
            return;
        }
        $kuadrat = pow($bilangan, 2);
        echo "<script>alert('Hasil $bilangan kuadrat = $kuadrat');
        window.history.go(-1);</script>";
    }

    switch ($submit) {
        case "Tampilkan":
            if (!empty($pesan)) {
                tampilAlert($pesan);
            } else {
                echo "<script>alert('Pesan tidak boleh kosong!');
                window.history.go(-1);</script>";
            }
            break;
        case "Kuadrat":
            kuadrat($bilangan);
            break;
        default:
            echo "<script>alert('Aksi tidak valid!');
            window.history.go(-1);</script>";
            break;
    }
}
?>
<HTML>
<HEAD>
    <TITLE>Latihan Fungsi</TITLE>
```

```

</HEAD>
<BODY>
  <FORM METHOD="POST" ACTION="P2f.php">
    Tuliskan Pesan Anda: <INPUT TYPE="text" NAME="pesan">
    <INPUT TYPE="submit" name="submit" value="Tampilkan">
    <BR><HR>
    Tuliskan Bilangan: <INPUT TYPE="text" NAME="bilangan" size=3>
    <INPUT TYPE="submit" name="submit" value="Kuadrat">
  </FORM>
</BODY>
</HTML>

<?php
if ($_SERVER['REQUEST_METHOD'] === 'POST') { // Sesudah Modifikasi
  $submit = $_POST['submit'] ?? null;
  $pesan = $_POST['pesan'] ?? "";
  $bilangan = $_POST['bilangan'] ?? null;

  function tampilAlert($pesan) {
    echo "<script>alert('Pesan: $pesan');
    window.history.go(-1);</script>";
  }

  function kuadrat($bilangan) {
    if (!is_numeric($bilangan)) {
      echo "<script>alert('Input harus berupa angka!');
      window.history.go(-1);</script>";
      return;
    }
    $kuadrat = pow($bilangan, 2);
    echo "<script>alert('Hasil $bilangan kuadrat = $kuadrat');
    window.history.go(-1);</script>";
  }

  function cGanGen($bilangan) {
    if (!is_numeric($bilangan)) {
      echo "<script>alert('Input harus berupa angka!');
      window.history.go(-1);</script>";
      return;
    }
    if ($bilangan % 2 == 0) {
      echo "<script>alert('Bilangan $bilangan adalah Genap');
      window.history.go(-1);</script>";
    } else {
      echo "<script>alert('Bilangan $bilangan adalah Ganjil');
      window.history.go(-1);</script>";
    }
  }

  switch ($submit) {
    case "Tampilkan":
      if (!empty($pesan)) {
        tampilAlert($pesan);
      } else {
        echo "<script>alert('Pesan tidak boleh kosong!');
        window.history.go(-1);</script>";
      }
      break;
    case "Kuadrat":
      kuadrat($bilangan);
      break;
    case "Cek Bilangan":
      cGanGen($bilangan);
      break;
    default:
      echo "<script>alert('Aksi tidak valid!');
      window.history.go(-1);</script>";
      break;
  }
}
?>
<HTML>
  <HEAD>
    <TITLE>Latihan Fungsi</TITLE>
  </HEAD>

```

```

<BODY>
<FORM METHOD="POST" ACTION="P2f.2.php">
  Tuliskan Pesan Anda: <INPUT TYPE="text" NAME="pesan">
  <INPUT TYPE="submit" name="submit" value="Tampilkan">
  <BR><HR>
  Tuliskan Bilangan: <INPUT TYPE="text" NAME="bilangan" size=3>
  <INPUT TYPE="submit" name="submit" value="Kuadrat">
  <INPUT TYPE="submit" name="submit" value="Cek Bilangan">
</FORM>
</BODY>
</HTML>

```

### Analisis:

Kode Program di atas mendemonstrasikan interaksi pengguna melalui form HTML yang diproses oleh PHP dimana terdapat form HTML yang terdiri dari dua bagian input yaitu untuk menampilkan pesan dan untuk menghitung kuadrat dari bilangan yang dimasukkan. Form ini dikirimkan dengan metode “POST” ke halaman "P2f.php". Di dalam kode PHP tersebut pertama-tama memeriksa request metode POST yang diterima menggunakan “\$\_SERVER['REQUEST\_METHOD']”. Jika benar maka data dari form diambil melalui “\$\_POST” seperti nilai pesan dan bilangan. Variabel “\$submit” menyimpan tombol submit yang diklik oleh pengguna yaitu "Tampilkan" atau "Kuadrat" yang dapat menentukan aksi selanjutnya. Ada dua fungsi utama dalam kode ini yaitu “tampilAlert(\$pesan)” dan “kuadrat(\$bilangan)”. Fungsi “tampilAlert(\$pesan)” berfungsi untuk menampilkan pesan yang diinputkan oleh pengguna dalam form melalui JavaScript “alert()”. Fungsi tersebut juga menggunakan “window.history.go(-1)” untuk mengarahkan pengguna kembali ke halaman awal setelah alert ditampilkan. Sementara itu fungsi “kuadrat(\$bilangan)” memeriksa input yang diberikan berupa angka menggunakan “is\_numeric()”. Jika input bukan angka maka sebuah alert akan muncul menhinformasikan pengguna bahwa input harus angka. Jika input valid maka fungsi akan menghitung kuadrat dari bilangan yang diinputkan menggunakan “pow()” dan menampilkan hasil dalam alert. Jika tombol submit yang ditekan adalah selain "Tampilkan" atau "Kuadrat" atau jika input kosong maka akan menampilkan pesan kesalahan menggunakan JavaScript.

Selanjutnya untuk modifikasi kode di bawah memiliki perbedaan utama yang terletak pada penambahan fungsi baru ‘cGanGen(\$bilangan)’ yang berguna untuk memeriksa bilangan yang dimasukkan genap atau ganjil. Fungsi ini bekerja dengan memastikan input berupa angka dengan “is\_numeric()” kemudian memeriksa bilangan tersebut genap atau ganjil menggunakan operator modulus “(%)”. Jika valid maka hasilnya ditampilkan menggunakan JavaScript alert yang meninformasikan pengguna apakah bilangan tersebut genap atau ganjil. Selain itu perubahan lainnya ada pada form HTML yang memiliki tombol submit dengan label "Cek Bilangan" yang berguna agar pengguna dapat memilih aksi selain aksi "Tampilkan" untuk menampilkan pesan dan "Kuadrat" untuk menghitung kuadrat bilangan. Kode PHP tersebut diupdate dengan menambahkan case baru pada switch statement untuk aksi "Cek Bilangan" dengan memanggil fungsi “cGanGen(\$bilangan)” untuk memproses input.

## Tugas

1. Buat program untuk mengurutkan data secara ascending dengan tampilan sebagai berikut

The top screenshot shows a web browser window with the URL 'localhost/Praktikum\_Pemrograman920\_Web%202/Modul2/T1.php'. It contains a form with a label 'Jumlah Data' and a value of '4'. Below it is a button labeled 'Input Data'.

The bottom screenshot shows the same browser window after the form is submitted. It displays the following output:

```

Inputan Bilangan:
Angka [1]: 3
Angka [2]: 4
Angka [3]: 7
Angka [4]: 9
Total: 23
Rata-rata: 5.75

```

## Script Code:

```
<HTML>
<BODY>
  <FORM METHOD="POST" ACTION="">
    <?php
      if (!isset($_POST['step'])) {
        echo "Jumlah Data: <input type='number' name='total' min='1' required><br>";
        echo "<input type='hidden' name='step' value='2'>";
        echo "<input type='submit' value='Input Data'>";
      } elseif ($_POST['step'] == 2) {
        $total = intval($_POST['total']);
        echo "<input type='hidden' name='total' value='$total'>";
        echo "<input type='hidden' name='step' value='3'>";
        echo "Inputkan Bilangan:<br>";
        for ($i = 0; $i < $total; $i++) {
          echo "Angka[" . ($i + 1) . "]: <input type='number' name='angka[" . ($i + 1) . "]' required><br>";
        }
        echo "<input type='submit' value='Tampil'>";
      } elseif ($_POST['step'] == 3) {
        $angka = $_POST['angka'];
        sort($angka);
        $total = array_sum($angka);
        $rata2 = $total / count($angka);
        echo "Inputan Bilangan:<br>";
        foreach ($angka as $index => $value) {
          echo "Angka [" . ($index + 1) . "]: " . htmlspecialchars($value) . "<br>";
        }
        echo "Total: " . $total;
        echo "<br>Rata-rata: " . $rata2;
      }
    ?>
  </FORM>
</BODY>
</HTML>
```

2. PHP memiliki fungsi bawaan diantaranya fungsi string, fungsi aritmatika dan sebagainya. Jelaskan dan beri contoh untuk fungsi yang ada dalam PHP baik fungsi bawaan maupun fungsi buatan.

### Fungsi Bawaan

#### a. Fungsi String

strlen(): untuk menghitung panjang String.

strtoupper(): untuk mengubah setiap huruf pada String menjadi kapital.

strtolower(): untuk mengubah setiap huruf pada String menjadi kecil semua.

substr(): untuk mengambil Sebagian String.

#### b. Fungsi Aritmatika

abs(): untuk mengubah nilai menjadi nilai absolut.

pow(): untuk mengangkat nilai

sqrt(): untuk mengakar kuadratkan nilai

round(): untuk membulatkan angka

#### c. Fungsi Array

count(): Untuk menghitung jumlah elemen di dalam array

array\_merge(): untuk menggabungkan dua array menjadi satu array.

in\_array(): Untuk mengecek apakah nilai ada di dalam suatu array.



## Fungsi Bawaan

### Script Code:

```
<?php
    function luasLingkaran($jarijr) {
        return 3.14 * $jarijr * jarijr;
    }

    echo luasLingkaran(7);
?>
```