

The background of the slide is a soft, teal-colored landscape. It features a calm body of water in the foreground, reflecting the surrounding environment. In the distance, there are rolling hills or mountains, some of which are partially obscured by a light mist or fog. The overall color palette is a range of blues and greens, creating a serene and atmospheric setting.

# PHP 2:

Array, Fungsi, Modularisasi

# Overview

- Array
- Fungsi
- Modularisasi

# Array

- Array merupakan variabel yang berisi lebih dari 1 data, atau dapat dikatakan sebagai variabel jamak.
- Variabel array pada PHP, kadang tidak perlu kita definisikan. Namun untuk indeks array, kita menggunakan indeks mulai dari 0 hingga sejumlah elemen array dikurang 1.
- Sebagai analogi, ketika kita mempunyai banyak nama buah (variabel), maka daripada kita menyatakan tiap nama buah dengan variabel \$buah1, \$buah2, \$buah3, ...dst.

# Array

Cara pemberian nilai pada variabel Array:

- **Contoh 1 :**

```
$buah[0] = "nanas";
```

```
$buah[1] = "mangga";
```

```
$buah[2] = "jambu";
```

- **Contoh 2 :**

```
$buah[] = "nanas";
```

```
$buah[] = "mangga";
```

```
$buah[] = "jambu";
```

- **Contoh 3 :**

```
$buah = array("nanas", "mangga", "jambu");
```

# Fungsi

- Fungsi dapat dikategorikan kedalam:
  - Built-in
  - User Defined Function (UDF)
  - external

# Fungsi Built-in

Merupakan fungsi yang sudah ada pada PHP,  
Contoh:

- Fungsi Aritmatika: sin, cos, tan, asin, deg2rad, dll.
- Fungsi Tanggal & waktu : date, checkdate
- Fungsi String : strlen, strpos, strtolower, strtoupper, substr, dll.

# Fungsi

## Contoh Fungsi Aritmatik

```
<?php
```

```
$a = pow(2,10);      //Fungsi perpangkatan
```

```
$b = sqrt(100);      //Fungsi akar
```

```
$c = ceil(4.25);     //Pembulatan keatas
```

```
$d = floor(4.25);    //Pembulatan kebawah
```

```
echo "2 pangkat 10 = $a <br>";
```

```
echo "akar 100 = $b <br>";
```

```
echo "ceil(4.25) = $c <br>";
```

```
echo "floor(4.25) = $d <br>";
```

```
?>
```

2 pangkat 10 = 1024

akar 100 = 10

ceil(4.25) = 5

floor(4.25) = 4

# Fungsi

## Contoh Fungsi Date:

```
<?
```

```
$skr = date("d/m/Y");
```

```
echo "Sekarang adalah $skr <br>";
```

```
$waktu = date("h:i:s A"); //A menunjukkan AM atau PM
```

```
echo "Jam menunjukkan pukul : $waktu";
```

```
?>
```



# Fungsi

Contoh Fungsi String:

```
<?php
```

```
$str = "Belajar PHP ternyata Menyenangkan";
```

```
echo strtolower($str); //Ubah huruf ke kecil semua
```

```
echo "<br>";
```

```
echo strtoupper($str); //Ubah huruf ke besar semua
```

```
echo "<br>";
```

```
echo str_replace("Menyenangkan", "mudah lho", $str); //Mengganti  
string
```

```
?>
```

# Fungsi UDF:

- Fungsi yang dibuat sendiri oleh pemrogram. Dapat diletakkan dimana saja dalam script PHP, tapi sebaiknya diletakkan paling atas agar memudahkan dalam proses debugging dalam tahap pembuatan dan pengembangan aplikasi.

- Deklarasi UDF:

```
Function nama_fungsi (parameter1, ..., n) {  
    Statement;  
}
```

- Pemanggilan UDF:

```
nama_fungsi (parameter1, ..., n);
```

# Fungsi

- Fungsi dengan parameter tanpa umpan balik.

```
<?php
function cetak_ganjil ($awal, $akhir) {
    for ($i=$awal; $i<$akhir; $i++) {
        if ($i%2 == 1) {
            echo "$i ";
        }
    }
}
//pemanggilan fungsi
$a = 10;
$b = 50;
echo "<b>Bilangan ganjil dari $a sampai $b : </b><br>";
cetak_ganjil($a, $b);
?>
```

# Fungsi

- Fungsi menggunakan parameter tanpa umpan balik

```
<?php
```

```
function luas($p, $l) {
```

```
    $luas = $p * $l;
```

```
    echo $luas;
```

```
}
```

```
luas(5,3) ;
```

```
?>
```

# Fungsi

- Fungsi dengan parameter dan Umpan Balik (return value)

```
<?php
```

```
function luas($p, $l) {
```

```
    $luas = $p * $l;
```

```
    return $luas;
```

```
}
```

```
echo luas(5,3) ;
```

```
?>
```

# Fungsi

- Fungsi dengan passing value

```
<?php
function tambah_string ($str) {
    $str = $str . ", Jakarta";
    return $str;
}
//
$str = "Universitas Budi Luhur";
echo "\$str = $str<br>";
echo tambah_string ($str). "<br>";
echo "\$str = $str<br>";
?>
```

# Modularisasi

- ▶ require
- ▶ include
- ▶ require\_once
- ▶ include\_once

# Modularisasi

- Modularisasi dalam pemrograman umum dilakukan dan sangat diperlukan untuk mempermudah debugging dan pengembangan program.
- Modularisasi berarti melakukan pembuatan program berdasarkan modul-modul .
- Setiap modul dikembangkan untuk tujuan atau fungsi khusus. Modul dibuat secara general (umum). Modul dapat berupa fungsi atau prosedur.



# Require

- ▶ Mekanisme untuk menyertakan file ke dalam sebuah file php
- ▶ Jika require gagal menyertakan file yang dimaksud maka akan terjadi *fatal error*
- ▶ Contoh :

```
require 'vars.php';
```

# Modularisasi

- Statement Include akan menyertakan isi suatu file tertentu.
- Include dapat diletakkan di dalam suatu looping misalkan dalam statement for atau while.
- Sintak penulisan: `include (namafile);`

# Include

- ▶ Mekanisme untuk menyertakan file ke dalam sebuah file php
- ▶ Jika *include* gagal menyertakan file yang dimaksud maka akan memunculkan *warning*
- ▶ Contoh :

```
include 'vars.php';
```

# Modularisasi

```
<?php  
echo("-----  
echo("Lab. Multimedia Broadcasting <br>");  
echo("-----  
echo("<br>");  
?>
```

```
<?php  
for ($b=1; $b<5; $b++)  
{  
include("modularisasi2.php");  
//include bisa dipanggil lbh dari 1x  
}  
?>
```

-----  
Lab. Multimedia Broadcasting  
-----

-----  
Lab. Multimedia Broadcasting  
-----

-----  
Lab. Multimedia Broadcasting  
-----

The background of the slide features a scenic landscape. In the foreground, there is a calm body of water, likely a lake, with subtle ripples. In the distance, a range of mountains is visible under a soft, hazy sky. A faint rainbow is visible in the upper right portion of the image, adding a touch of color to the scene. The overall tone is peaceful and natural.

**TERIMA KASIH**