

LAPORAN TUGAS PENDAHULUAN PRAKTIKUM
PENGEMBANGAN APLIKASI WEB
TPP MODUL 1



Disusun Oleh:

Nama: Safri Iwanussuyuf

NIM: 240411100170

Kelas: IF 3A

Dosen Pengampu:

Nama: Moh. Kautsar Sophan, S.Kom., M.MT

NIP: 197707132002121004

Asisten Praktikum:

Nama: M Sultan Abdurrahman Al Zudas

NIM: 230411100191

PRODI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS TRUNOJOYO MADURA
2025

SOAL

1. Apa yang anda ketahui tentang PHP, mulai dari asal usulnya sampai dengan kelebihan dan kekurangan penggunaan pemrograman dengan PHP?
2. Bagaimana aturan-aturan dalam:
 - a. Skrip PHP dan cara pemodelannya
 - b. Nama variabel PHP
3. Jelaskan yang anda ketahui tentang:
 - a. Variabel global
 - b. Variabel Session
 - c. Operator
 - d. Function
 - e. File dan tahapan pengaksesan file
4. Jelaskan apa kegunaan dari:
 - a. Fungsi String
 - b. Fungsi Date
 - c. Fungsi Mail
 - d. Fungsi rekursif

JAWABAN

1. A. Asal-usul PHP:

Diciptakan oleh Rasmus Lerdorf (1994–1995), awalnya sebagai kumpulan skrip CGI (Common Gateway Interface) dalam bahasa C untuk melacak kunjungan ke situs pribadinya.

Nama awalnya “Personal Home Page/Forms Interpreter (PHP/FI)”.

Pada 1997, dikembangkan lebih serius oleh tim programmer, berganti nama menjadi PHP: Hypertext Preprocessor (nama rekursif).

Sejak itu PHP berkembang sebagai bahasa scripting open-source yang banyak digunakan untuk pengembangan web dinamis.

B. Pengertian PHP

PHP adalah bahasa server-side scripting yang diproses di server, hasilnya dikirim ke browser dalam bentuk HTML.

Umumnya digunakan untuk membuat website dinamis, sistem manajemen konten (CMS), e-commerce, hingga API backend.

C. Kelebihan PHP

Open Source & Gratis → bisa digunakan siapa saja tanpa biaya lisensi.

Mudah Dipelajari → sintaks mirip C, Java, dan Perl, sehingga mudah dipahami pemula.

Kompatibel & Portabel → dapat dijalankan di berbagai sistem operasi (Windows, Linux, macOS) dan mendukung banyak web server (Apache, Nginx, IIS).

Integrasi Mudah → terintegrasi dengan berbagai database (MySQL, PostgreSQL, MariaDB, Oracle, dsb.).

Komunitas Besar → banyak dokumentasi, forum, dan framework populer (Laravel, CodeIgniter, Symfony).

Performa Tinggi (versi terbaru) → sejak PHP 7+, performa jauh lebih cepat dibanding versi lama.

Banyak Framework & CMS → mendukung WordPress, Joomla, Drupal, dll.

D. Kekurangan PHP

Keamanan → jika tidak ditulis dengan baik, rawan celah (SQL Injection, XSS, dsb.).

Konsistensi Sintaks → beberapa fungsi PHP tidak konsisten dalam penamaan (misalnya `str_replace()` vs `strpos()`).

Kurang Modern di Versi Lama → sebelum PHP 7, performanya kalah jauh dibanding bahasa lain (misalnya Node.js, Python).

Tidak Cocok untuk Semua Aplikasi → kuat untuk web, tapi kurang optimal untuk aplikasi desktop atau machine learning.

Legacy Code yang Banyak → masih ada kode lama yang sulit di-maintain.

2. A. Aturan Penulisan Skrip PHP

1. **Open Tag PHP:**

Semua kode PHP harus diapit oleh tag pembuka `<?php` dan tag penutup `?>` agar interpreter PHP dapat mengenalinya.

2. **Ekstensi File:**

File yang berisi kode PHP harus disimpan dengan ekstensi `.php` agar web server dapat memprosesnya.

3. **Deklarasi Variabel:**

Variabel dimulai dengan tanda dolar (\$) diikuti nama variabel, misalnya `$nama_pengguna`.

4. **Sifat Case-Sensitive :**

PHP membedakan huruf besar dan kecil, sehingga `$nama` berbeda dengan `$Nama`.

5. **Akhir Statement :**

Setiap pernyataan atau perintah dalam PHP diakhiri dengan titik koma (;).

6. **Pemisahan Kode:**

Spasi, tab, dan enter pada dasarnya diabaikan oleh PHP, yang memungkinkan Anda memformat kode agar lebih mudah dibaca.

Cara Pemodelan:

1. **Pemisahan Logika dan Presentasi:**

Gunakan open tag `<?php` untuk memisahkan kode PHP dari HTML. Kode PHP ditempatkan di dalam tag tersebut, sedangkan HTML berada di luarnya.

2. **Penggunaan Tags XML:**

Metode penulisan open tag yang paling direkomendasikan adalah `<?php` dan `?>` (XML Style), yang memudahkan server memproses kode.

3. **Struktur MVC (Model-View-Controller):**

Untuk proyek yang lebih kompleks, gunakan pola MVC untuk memisahkan data (Model), tampilan (View), dan logika aplikasi (Controller), sehingga kode lebih terstruktur.

4. **Penggunaan Konstanta:**

Gunakan `define()` untuk mendefinisikan konstanta yang nilainya tidak akan berubah selama program berjalan, misalnya `define("PHAR", 3.14);`.

5. **Pemformatan Kode:**

Selalu format kode dengan rapi dan konsisten, baik untuk diri sendiri maupun orang lain yang mungkin akan membacanya.

B. Aturan Penulisan Variabel PHP

1. **Menggunakan Tanda \$**

Selalu gunakan tanda \$ sebelum menuliskan variabel dalam php

Contoh: \$usia

Jadi diatas merupakan variabel bernama usia

2. **Harus Diawali huruf atau underscore (“_”)**

Contoh:\$nama atau \$_nama

Jadi diatas ada variabel yang sesuai dengan ketentuan

3. **Tidak diawali dengan angka**

Jadi variabel itu tidak boleh diawali dengan angka tetapi boleh mengandung angka

Contoh:\$1siswa variabel ini tidak sesuai ketentuan nantinya akan error

Contoh:\$siswa1 variabel ini tidak akan error Karena tidak diawali dengan angka meskipun mengandung angka didalamnya

4. **Hanya boleh berisi angka, huruf, dan underscore(0-9, A-z, dan _)**

Jadi hanya angka huruf dan juga underscore yang boleh menjadi nama variabel.

Contoh:\$Nama_siswa1

Contoh di atas merupakan contoh penamaan variabel yang benar,

5. **Variabel case-sensitive**

Jadi variabel pada php merupakan case-sensitive artinya akan beda hasilnya jika menggunakan huruf besar dengan menggunakan huruf kecil

Contoh:\$usia dengan \$USIA

usia dengan USIA akan berbeda hasilnya dan akan terdeteksi sebagai variabel yang berbeda.

3. A. Variabel Global

variabel yang didefinisikan di luar fungsi, sehingga bisa diakses dari seluruh bagian program (namun di dalam fungsi harus menggunakan keyword global agar bisa dipakai).

PHP juga punya Superglobals, yaitu variabel global bawaan, seperti \$_GET, \$_POST, \$_SESSION, \$_COOKIE, \$_SERVER, dll.

B. Variabel Session

Variabel yang digunakan untuk menyimpan informasi user selama sesi browsing berlangsung. Data tersimpan di server, bukan di browser.

Digunakan untuk login, shopping cart, dsb.

Untuk menggunakan session harus diawali dengan session_start().

Data session bisa diakses di halaman lain selama sesi aktif.

C. Operator dalam PHP

Operator digunakan untuk operasi pada variabel dan nilai.

Aritmatika

+ (tambah), - (kurang), * (kali), / (bagi), % (modulus).

\$a = 10; \$b = 3;

```
echo $a + $b; // 13
```

```
echo $a % $b; // 1
```

Penugasan

=, +=, -=, *=, /=, .= (penggabungan string).

Perbandingan

==, !=, >, <, >=, <=, === (identik).

Logika

&& (dan), || (atau), ! (not).

Increment/Decrement

++\$a, \$a++, --\$a, \$a--.

String

. (penggabungan string).

```
$nama = "Andi";
```

```
echo "Halo " . $nama;
```

D. Function

Sekumpulan perintah yang dikemas agar bisa dipanggil berulang kali.

Ada fungsi bawaan (built-in) (misalnya strlen(), date(), round()) dan fungsi buatan sendiri.

Contoh fungsi buatan:

```
<?php
```

```
function sapa($nama) {
```

```
return "Halo, $nama!";
```

```
}
```

```
echo sapa("Budi");
```

E. File dan Tahapan Pengaksesan File

PHP menyediakan fungsi untuk membaca, menulis, membuka, dan menutup file.

Tahapan akses file:

Membuka file → `fopen("namafile.txt", "mode")`

Mode:

"r" → baca

"w" → tulis (hapus isi lama)

"a" → tambah data

"r+", "w+", "a+" → baca + tulis

Membaca/Menulis file

`fread()`, `fgets()`, `fwrite()`.

Menutup file → `fclose()`.

4. A. Fungsi String dalam PHP

Kegunaan → untuk memanipulasi teks (string).

PHP punya banyak built-in function untuk string, misalnya:

`strlen($str)` → menghitung panjang string.

`strtoupper($str)` / `strtolower($str)` → ubah huruf besar/kecil.

`substr($str, start, length)` → ambil potongan string.

`str_replace("cari", "ganti", $str)` → ganti kata.

`explode(" ", $str)` → memecah string jadi array.

Contoh:

```
<?php  
$teks = "Halo Dunia";  
echo strlen($teks); // 10  
echo strtoupper($teks); // HALO DUNIA  
echo str_replace("Dunia", "PHP", $teks); // Halo PHP  
?>
```

B. Fungsi Date dalam PHP

Kegunaan → menampilkan atau memanipulasi tanggal & waktu.

Format date("format"), contoh:

date("Y") → tahun (2025)

date("m") → bulan (09)

date("d") → hari (24)

date("H:i:s") → jam:menit:detik

Contoh:

```
<?php  
echo date("l, d F Y"); // Selasa, 24 September 2025  
echo date("H:i:s"); // 11:30:45  
?>
```

C. Fungsi Mail dalam PHP

Kegunaan → untuk mengirim email langsung dari server menggunakan mail().

Sintaks:

```
mail(to, subject, message, headers, parameters);
```

Umumnya digunakan untuk form kontak, verifikasi akun, notifikasi.

Contoh sederhana:

```
<?php  
$to = "user@example.com";  
$subject = "Tes Email PHP";  
$message = "Halo, ini email dari PHP!";  
$headers = "From: admin@domain.com";  
mail($to, $subject, $message, $headers);  
?>
```

Catatan: Fungsi ini hanya bekerja jika server sudah dikonfigurasi dengan SMTP/Sendmail.

D. Fungsi Rekursif

Definisi → fungsi yang memanggil dirinya sendiri untuk menyelesaikan masalah.

Digunakan untuk masalah yang berulang, seperti faktorial, perhitungan Fibonacci, menelusuri struktur data pohon, dsb.