

PHP Full Notes (Hinglish Explanation)

◆ 1. PHP Kya Hai?

PHP ka full form hai "Hypertext Preprocessor".

Yeh ek server-side scripting language hai.

Matlab:

- Jab aap website kholte ho (index.html), woh browser me run hoti hai (client-side).
- Lekin PHP ka code server par run hota hai, aur result (HTML form me) browser ko bhejta hai.

👉 Example:

Agar aap likhte ho:

```
<?php echo "Hello World"; ?>
```

Toh browser me sirf Hello World dikhai dega — PHP ka code nahi dikhega.

Server ne us code ko execute karke result bhej diya.

◆ 2. PHP Kyu Use Karte Hain?

PHP ka use mainly dynamic websites banane ke liye hota hai.

Examples:

- Login/Register system
- Contact form
- To-Do list (CRUD)
- E-commerce site (add to cart, payment, etc.)
- Admin panel
- CMS (WordPress bhi PHP me bana hai)

◆ 3. PHP Kaise Kaam Karta Hai?

Simple process samjho 📌

1. User apne browser me request bhejta hai — index.php file ke liye.
2. Server (like XAMPP / WAMP / localhost / hosting server) PHP code ko run karta hai.
3. PHP file ka result HTML me convert hota hai.
4. Browser user ko final page dikhata hai.

🧠 Server ke bina PHP code browser me run nahi karega!

Isliye hamesha localhost par run karo — jaise:

<http://localhost/myproject/index.php>

◆ 4. PHP Syntax (Basic Rules)

PHP code hamesha `<?php ... ?>` ke beech me likha jata hai.

```
<?php
```

```
echo "Hello World!";
```

```
?>
```

Important Points:

- Every PHP statement ends with ; (semicolon).
- PHP case-insensitive hai (for keywords). Example: echo aur ECHO dono chalega.
- Comments likhne ke liye:
 - // Single line comment
 - /* Multi-line comment */

◆ 5. PHP Variables

Variables data store karne ke liye use hote hain.

PHP me variable hamesha \$ se start hota hai.

```
$name = "Talha";
```

```
$age = 22;
```

```
echo "My name is $name and I am $age years old.";
```

🧠 Rules:

- Variable \$ se start hota hai.
- Space nahi hoti variable name me.
- Case-sensitive hote hain (\$name ≠ \$Name).

◆ 6. PHP Data Types

PHP me mainly ye data types hote hain:

Type	Example	Description
String	"Hello"	Text
Integer	25	Number without decimal
Float	3.14	Decimal number
Boolean	true / false	True ya False
Array	["apple", "banana"]	Multiple values
Object	Custom class data	

NULL	NULL	Empty value
-------------	-------------	------------------------

Example:

```
$name = "Talha";
```

```
$age = 22;
```

```
$price = 99.99;
```

```
$isAdmin = true;
```

◆ 7. PHP Operators

Operators woh symbols hote hain jinke through hum variables me calculation ya comparison karte hain.

Arithmetic Operators

```
$a + $b // Addition
```

```
$a - $b // Subtraction
```

```
$a * $b // Multiplication
```

```
$a / $b // Division
```

```
$a % $b // Modulus
```

Comparison Operators

```
$a == $b // Equal to
```

```
$a != $b // Not equal
```

```
$a > $b // Greater than
```

```
$a < $b // Less than
```

Logical Operators

```
$a && $b // AND
```

```
$a || $b // OR
```

!\$a // NOT

◆ 8. Conditional Statements

Ye program me decision lene ke liye use hoti hain.

```
$age = 18;
```

```
if($age >= 18) {  
    echo "You can vote";  
}  
else {  
    echo "You are under 18";  
}
```

Agar ek se zyada condition check karni ho:

```
if($marks >= 90) {  
    echo "Grade A";  
}  
elseif($marks >= 60) {  
    echo "Grade B";  
}  
else {  
    echo "Grade C";  
}
```

◆ 9. Loops in PHP

Loop ka matlab hai — repeat karna jab tak condition true hai.

For Loop

```
for($i=1; $i<=5; $i++) {  
    echo "Number: $i <br>";  
}
```

While Loop

```
$i = 1;

while($i <= 5) {

    echo "$i <br>";

    $i++;

}
```

Foreach Loop (for arrays)

```
$fruits = ["apple", "banana", "mango"];

foreach($fruits as $f) {

    echo "$f <br>";

}
```

◆ **10. Arrays in PHP**

Array ek variable hota hai jisme multiple values store hoti hain.

```
$colors = ["red", "green", "blue"];

echo $colors[0]; // red
```

Associative Array

```
$person = ["name" => "Talha", "age" => 22];

echo $person["name"]; // Talha
```

◆ **11. Functions**

Function ek chhota code block hota hai jo reuse hota hai.

```
function greet($name) {

    echo "Hello, $name!";

}
```

```
greet("Talha"); // Output: Hello, Talha!
```

◆ **12. Forms and \$_POST / \$_GET**

PHP forms ke through data receive karta hai.

```
<form method="POST" action="welcome.php">
```

```
<input type="text" name="username">
```

```
<button type="submit">Submit</button>
```

```
</form>
```

welcome.php

```
<?php
```

```
$name = $_POST['username'];
```

```
echo "Welcome, $name!";
```

```
?>
```

\$_GET tab use hota hai jab data URL ke sath jata hai, jaise:

welcome.php?name=Talha

◆ 13. Include / Require

Code reuse karne ke liye use hota hai.

```
include 'header.php';
```

```
require 'db.php';
```

include agar file missing ho to warning deta hai (program continue hota hai).

require missing ho to program stop kar deta hai.

◆ 14. PHP with MySQL

Database se connect hone ke liye:

```
$conn = mysqli_connect("localhost", "root", "", "mydb");
```

```
if(!$conn) die("Connection failed");
```

Query run karne ke liye:

```
$sql = "SELECT * FROM users";
```

```
$result = mysqli_query($conn, $sql);
```

◆ 15. CRUD Concept (Create, Read, Update, Delete)

Operation	Description
Create	Insert data in database
Read	Display data from database
Update	Modify existing data
Delete	Remove data

Example query:

INSERT INTO todos (task) VALUES ('Buy milk');

SELECT * FROM todos;

UPDATE todos SET task='Buy eggs' WHERE id=1;

DELETE FROM todos WHERE id=2;

◆ 16. Super Global Variables

Variable	Use
\$_GET	Data from URL
\$_POST	Data from form
\$_SESSION	Store user data temporarily
\$_COOKIE	Store small info in browser
\$_SERVER	Server details

◆ 17. Sessions in PHP

Session user ke liye temporary data store karta hai (e.g. login info).

session_start();

```
$_SESSION['username'] = "Talha";
```

```
echo $_SESSION['username'];
```

Logout ke time:

```
session_start();
```

```
session_destroy();
```

◆ 18. File Handling

PHP files read/write kar sakta hai.

```
$file = fopen("data.txt", "w");
```

```
fwrite($file, "Hello PHP!");
```

```
fclose($file);
```

◆ 19. Error Handling

```
try {
```

```
    // code
```

```
} catch (Exception $e) {
```

```
    echo "Error: " . $e->getMessage();
```

```
}
```

◆ 20. PHP Advantages

- ✓ Easy to learn
- ✓ Open-source
- ✓ Fast and lightweight
- ✓ Works with all databases
- ✓ Best for backend and CMS (WordPress, Laravel)