

1. Which of the following is NOT a valid PHP conditional statement?

- a) if
- b) unless
- c) switch
- d) elseif

Answer: b) unless

2. Which keyword is used to check multiple conditions in PHP?

- a) switch
- b) if
- c) for
- d) while

Answer: a) switch

3. What will be the output of the following code?

```
$x = 10;  
if ($x > 5) {  
    echo "Greater";  
} else {  
    echo "Smaller";  
}
```

- a) Greater
- b) Smaller
- c) Error
- d) No output

Answer: a) Greater

4. Which of the following is the correct syntax for an if statement in PHP?

- a) if x > 5 { }
- b) if (\$x > 5) { }
- c) if \$x > 5 then { }
- d) if (\$x > 5) then { }

Answer: b) `if ($x > 5) { }`

5. What will be the output of the following code?

```
$x = 10;  
if ($x == "10") {  
    echo "Equal";  
} else {  
    echo "Not Equal";  
}
```

- a) Equal
- b) Not Equal
- c) Error
- d) No output

Answer: a) Equal

6. What will be the output of the following code?

```
$x = 10;  
if ($x === "10") {  
    echo "Identical";  
} else {  
    echo "Not Identical";  
}
```

- a) Identical
- b) Not Identical
- c) Error
- d) No output

Answer: b) Not Identical

7. The `else` statement in PHP is executed when the condition in `if` is:

- a) True
- b) False
- c) Always
- d) Never

Answer: b) False

8. What is the correct syntax of an `elseif` statement in PHP?

- a) `else if ($x > 10) { }`
- b) `elseif ($x > 10) { }`
- c) `else if { $x > 10 }`
- d) `elseif { $x > 10 }`

Answer: b) `elseif ($x > 10) { }`

9. The `switch` statement in PHP is similar to:

- a) for loop
- b) while loop
- c) if-else
- d) function

Answer: c) if-else

10. Which keyword is used to exit a switch case?

- a) stop
- b) exit
- c) break
- d) continue

Answer: c) break

11. What will be the output of the following code?

```
$color = "blue";
switch ($color) {
    case "red":
        echo "Red";
        break;
    case "blue":
        echo "Blue";
        break;
    default:
        echo "No match";
}
```

- a) Red
- b) Blue
- c) No match
- d) Error

Answer: b) Blue

12. The `default` case in a switch statement is executed when:

- a) No case matches
- b) Always
- c) First case is executed
- d) When `break` is not used

Answer: a) No case matches

13. What will be the output of the following code?

```
$x = 5;  
if ($x > 10)  
    echo "Greater";  
elseif ($x == 5)  
    echo "Equal";  
else  
    echo "Smaller";
```

- a) Greater
- b) Equal
- c) Smaller
- d) Error

Answer: b) Equal

14. The ternary operator in PHP is represented by:

- a) ??
- b) :
- c) ? :
- d) ->

Answer: c) ? :

15. What is the output of the following ternary operation?

```
$x = 10;  
echo ($x > 5) ? "Yes" : "No";
```

- a) Yes
- b) No
- c) Error
- d) 10

Answer: a) Yes

16. How many cases can be used inside a switch statement?

- a) Only 2
- b) Only 3
- c) Unlimited
- d) None

Answer: c) Unlimited

17. Which of the following is the correct syntax for a switch statement?

a)

```
switch {  
    case 1: echo "One";  
}
```

b)

```
switch ($var) {  
    case 1: echo "One"; break;  
}
```

c)

```
switch ($var) {  
    1: echo "One"; break;  
}
```

d)

```
switch ($var) {  
    case 1 echo "One"; break;  
}
```

Answer: b)

18. Which statement is used to terminate the execution of a loop or switch?

- a) stop
- b) exit
- c) break
- d) return

Answer: c) break

19. What will be the output of the following code?

```
$age = 18;  
echo ($age >= 18) ? "Adult" : "Minor";
```

- a) Adult
- b) Minor
- c) Error
- d) No output

Answer: a) Adult

20. Can a switch statement be nested in PHP?

- a) Yes
- b) No

Answer: a) Yes

21. Can we use multiple `case` values in a switch statement?

- a) Yes
- b) No

Answer: a) Yes

22. What happens if a `break` statement is omitted in a switch case?

- a) It exits the switch statement
- b) It throws an error
- c) It executes the next case(s) until a break is found
- d) It restarts the switch statement

Answer: c) It executes the next case(s) until a break is found

23. What will be the output of the following code?

```
$x = 3;
switch ($x) {
    case 1:
        echo "One";
    case 3:
        echo "Three";
    default:
        echo "None";
}
```

- a) One
- b) Three
- c) ThreeNone
- d) OneThreeNone

Answer: c) ThreeNone

24. What is the correct syntax for the ternary operator?

- a) `(condition) ? (true_statement) : (false_statement);`
- b) `if (condition) { true_statement; } else { false_statement; }`
- c) `(true_statement) : (false_statement) ? (condition);`
- d) None of the above

Answer: a) `(condition) ? (true_statement) : (false_statement);`

25. What is the output of the following code?

```
$y = 0;  
$result = ($y > 0) ? "Positive" : ($y < 0 ? "Negative" : "Zero");  
echo $result;
```

- a) Positive
- b) Negative
- c) Zero
- d) Error

Answer: c) Zero

26. Can an `if` statement be written without curly braces `{}`?

- a) Yes, if it has a single statement
- b) No, curly braces are mandatory
- c) Only inside loops
- d) Only in switch statements

Answer: a) Yes, if it has a single statement

27. What will be the output of the following code?

```
if (0) {  
    echo "True";  
} else {  
    echo "False";  
}
```

- a) True
- b) False
- c) Error
- d) No output

Answer: b) False

28. What does `===` operator check in PHP?

- a) Value only
- b) Type only
- c) Both value and type
- d) None of the above

Answer: c) Both value and type

29. What will be the output of the following code?

```
if ("0") {  
    echo "True";  
} else {  
    echo "False";  
}
```

- a) True
- b) False
- c) Error
- d) No output

Answer: a) True

30. Which of the following evaluates to `false` in PHP?

- a) 0
- b) ""
- c) null
- d) All of the above

Answer: d) All of the above

31. The `default` case in a switch statement is:

- a) Optional
- b) Mandatory
- c) Executed first
- d) Must be placed at the beginning

Answer: a) Optional

32. Can the `switch` statement be used with strings in PHP?

- a) Yes
- b) No

Answer: a) Yes

33. What will be the output of the following code?

```
if (true == "1") {  
    echo "Equal";  
} else {  
    echo "Not Equal";  
}
```

- a) Equal
- b) Not Equal
- c) Error
- d) No output

Answer: a) Equal

34. What is the purpose of the `continue` statement in PHP?

- a) Terminates loop execution
- b) Skips current iteration and continues to next one
- c) Stops script execution
- d) Restarts the script

Answer: b) Skips current iteration and continues to next one

35. What is the output of the following code?

```
if (true && false) {  
    echo "Yes";  
} else {  
    echo "No";  
}
```

- a) Yes
- b) No
- c) Error
- d) No output

Answer: b) No

36. How many `else if` conditions can be used in PHP?

- a) Only one
- b) Only two
- c) No limit
- d) Depends on PHP version

Answer: c) No limit

37. The `switch` statement can replace which conditional statement?

- a) if-else
- b) while
- c) for
- d) do-while

Answer: a) if-else

38. What will be the output of the following code?

```
$val = NULL;  
if ($val) {  
    echo "True";  
} else {  
    echo "False";  
}
```

- a) True
- b) False
- c) Error
- d) No output

Answer: b) False

39. What will be the output of the following code?

```
$num = 0;  
echo ($num ? "Yes" : "No");
```

- a) Yes
- b) No

- c) Error
- d) No output

Answer: b) No

40. The `elseif` keyword is equivalent to:

- a) `if-else`
- b) `else if`
- c) `switch`
- d) `while`

Answer: b) `else if`

41. Which statement is used to immediately stop script execution in PHP?

- a) `break`
- b) `exit`
- c) `continue`
- d) `return`

Answer: b) `exit`

42. Can we use an `if` statement inside another `if` statement?

- a) Yes
- b) No

Answer: a) Yes

43. Which PHP keyword is used to return a value from a function?

- a) `return`
- b) `exit`
- c) `break`
- d) `continue`

Answer: a) return

44. What will be the output of the following code?

```
$val = "10";  
if ($val === 10) {  
    echo "True";  
} else {  
    echo "False";  
}
```

- a) True
- b) False
- c) Error
- d) No output

Answer: b) False

45. Which operator is used to check for inequality in PHP?

- a) !=
- b) <>
- c) !==
- d) All of the above

Answer: d) All of the above

46. What is the output of `var_dump((bool) "0")`?

- a) bool(true)
- b) bool(false)
- c) 0
- d) 1

Answer: b) bool(false)

47. Which statement is used to execute a block of code only when a condition is false?

- a) if
- b) else
- c) !if
- d) switch

Answer: b) else

48. How do you write an if statement that executes when \$x is NOT equal to 100?

- a) if (\$x <> 100) {}
- b) if (\$x != 100) {}
- c) if (\$x !== 100) {}
- d) All of the above

Answer: d) All of the above

49. What will be the output of `echo 5 > 3 ? "Yes" : "No";`?

- a) Yes
- b) No
- c) Error
- d) Nothing

Answer: a) Yes

50. Is the following syntax correct?

```
if ($x = 5) { echo "Yes"; }
```

- a) Yes, but it has an assignment mistake
- b) No, it's incorrect
- c) Only in PHP 8+
- d) Only in strict mode

Answer: a) Yes, but it has an assignment mistake

1. Which of the following is NOT a loop in PHP?

- a) for
- b) while

- c) switch
- d) foreach

Answer: c) switch

2. Which PHP loop is used when the number of iterations is known?

- a) while
- b) for
- c) do-while
- d) foreach

Answer: b) for

3. What is the correct syntax for a `for` loop in PHP?

- a) `for ($i = 0; $i < 10; $i++) { }`
- b) `for $i = 0 to 10 { }`
- c) `for $i = 0; $i < 10 { }`
- d) `for ($i < 10; $i++) { }`

Answer: a) `for ($i = 0; $i < 10; $i++) { }`

4. What is the correct syntax of a `while` loop?

- a) `while $x > 10 { }`
- b) `while ($x > 10) { }`
- c) `while x > 10 then { }`
- d) `while ($x > 10) then { }`

Answer: b) `while ($x > 10) { }`

5. Which loop always executes at least once?

- a) for
- b) while

- c) do-while
- d) foreach

Answer: c) do-while

6. What will be the output of the following code?

```
$x = 5;
while ($x > 0) {
    echo $x;
    $x--;
}
```

- a) 54321
- b) 12345
- c) Infinite loop
- d) Error

Answer: a) 54321

7. Which loop is used to iterate over an array in PHP?

- a) for
- b) while
- c) do-while
- d) foreach

Answer: d) foreach

8. What will be the output of the following code?

```
$i = 0;
do {
    echo $i;
    $i++;
} while ($i < 3);
```

- a) 012
- b) 123
- c) Infinite loop
- d) Error

Answer: a) 012

9. How do you exit a loop in PHP?

- a) exit
- b) stop
- c) break
- d) return

Answer: c) break

10. What will be the output of the following code?

```
for ($i = 0; $i < 3; $i++) {  
    if ($i == 2) break;  
    echo $i;  
}
```

- a) 01
- b) 012
- c) 02
- d) 1

Answer: a) 01

11. Which statement skips the current iteration of a loop and moves to the next iteration?

- a) break
- b) exit
- c) return
- d) continue

Answer: d) continue

12. What will be the output of the following code?

```
for ($i = 0; $i < 5; $i++) {
```

```
        if ($i == 3) continue;
        echo $i;
    }
```

- a) 0124
- b) 01234
- c) 1234
- d) 0

Answer: a) 0124

13. What will be the output of the following code?

```
$x = 5;
while ($x < 5) {
    echo "Hello";
}
```

- a) Hello
- b) Infinite loop
- c) No output
- d) Error

Answer: c) No output

14. How do you iterate over an associative array in PHP?

- a) Using `for` loop
- b) Using `foreach` loop
- c) Using `while` loop
- d) Both b and c

Answer: b) Using `foreach` loop

15. Which loop structure is best for iterating over an indexed array?

- a) `for`
- b) `while`
- c) `do-while`
- d) `switch`

Answer: a) for

16. Which loop is best for executing a block of code at least once?

- a) while
- b) do-while
- c) for
- d) foreach

Answer: b) do-while

17. What is the default step value in a `for` loop?

- a) 0
- b) 1
- c) -1
- d) 2

Answer: b) 1

18. Can a loop be nested inside another loop in PHP?

- a) Yes
- b) No

Answer: a) Yes

19. What is the output of the following code?

```
$arr = array("a", "b", "c");  
foreach ($arr as $value) {  
    echo $value;  
}
```

- a) abc
- b) a b c
- c) "a""b""c"
- d) Error

Answer: a) abc

20. What will be the output of the following code?

```
for ($i = 0; $i < 3; $i++) {  
    for ($j = 0; $j < 2; $j++) {  
        echo "*";  
    }  
}
```

- a) **
- b) ****
- c) *****
- d) ***

Answer: c) *****

21. What happens when a `break` statement is used in a nested loop?

- a) It exits both loops
- b) It exits only the innermost loop
- c) It throws an error
- d) None of the above

Answer: b) It exits only the innermost loop

22. What will be the output of the following code?

```
$count = 1;  
while ($count <= 3) {  
    echo $count;  
    $count++;  
}
```

- a) 123
- b) 0123
- c) 3
- d) Error

Answer: a) 123

23. What happens if the condition in a `while` loop is always `true`?

- a) Loop runs infinitely
- b) Loop executes only once
- c) Loop executes 10 times
- d) Syntax error

Answer: a) Loop runs infinitely

24. What will be the output of the following code?

```
for ($i = 0; $i < 3; $i++) {  
    echo "$i ";  
}
```

- a) 0 1 2
- b) 1 2 3
- c) 012
- d) 0 1 2 3

Answer: a) 0 1 2

25. Which loop structure is best when the number of iterations is unknown?

- a) for
- b) while
- c) do-while
- d) foreach

Answer: b) while

26. Can a loop be used inside a function in PHP?

- a) Yes
- b) No

Answer: a) Yes

27. What is an infinite loop?

- a) A loop that runs forever
- b) A loop with a fixed number of iterations
- c) A loop that never starts
- d) None of the above

Answer: a) A loop that runs forever

28. Can a `foreach` loop iterate over an object?

- a) Yes
- b) No

Answer: a) Yes

29. What is the output of `var_dump(5 > 3 ? "Yes" : "No");`?

- a) `string(3) "Yes"`
- b) `Yes`
- c) `No`
- d) Error

Answer: a) `string(3) "Yes"`

30. Can we use a `for` loop inside a `while` loop?

- a) Yes
- b) No

Answer: a) Yes

1. Which of the following loops executes at least once, even if the condition is false?

- a) `for`
- b) `while`
- c) `do-while`
- d) `foreach`

Answer: c) do-while

2. What is the correct syntax for a `while` loop?

- a) `while $x > 5 { }`
- b) `while ($x > 5) { }`
- c) `while x > 5 { }`
- d) `while ($x > 5) then { }`

Answer: b) `while ($x > 5) { }`

3. How many times does the following loop execute?

```
for ($i = 0; $i < 5; $i++) {  
    echo $i;  
}
```

- a) 4 times
- b) 5 times
- c) Infinite times
- d) 0 times

Answer: b) 5 times

4. Which loop is best when the number of iterations is known?

- a) `while`
- b) `for`
- c) `do-while`
- d) `foreach`

Answer: b) `for`

5. What will be the output of the following code?

```
$i = 5;  
while ($i < 5) {  
    echo "Hello";  
}
```

- a) Hello
- b) Infinite loop
- c) No output
- d) Error

Answer: c) No output

6. What is the output of the following code?

```
$i = 0;  
do {  
    echo $i;  
    $i++;  
} while ($i < 3);
```

- a) 012
- b) 123
- c) Infinite loop
- d) Error

Answer: a) 012

7. What will happen if the loop condition is always `true`?

- a) The loop runs infinitely
- b) The loop stops after 10 iterations
- c) The loop executes only once
- d) Syntax error

Answer: a) The loop runs infinitely

8. What is the output of the following code?

```
for ($i = 0; $i < 3; $i++) {  
    echo "$i ";  
}
```

- a) 0 1 2
- b) 1 2 3
- c) 012
- d) 0 1 2 3

Answer: a) 0 1 2

9. Which statement immediately terminates a loop?

- a) continue
- b) exit
- c) break
- d) return

Answer: c) break

10. What will be the output of the following code?

```
$i = 0;
while ($i < 3) {
    if ($i == 1) break;
    echo $i;
    $i++;
}
```

- a) 0
- b) 01
- c) 012
- d) 1

Answer: a) 0

11. What does the `continue` statement do?

- a) Exits the loop
- b) Skips the current iteration and continues to the next one
- c) Stops the script execution
- d) None of the above

Answer: b) Skips the current iteration and continues to the next one

12. How do you create an infinite loop using `while`?

- a) `while (true) { }`
- b) `while (1) { }`
- c) `while ("PHP") { }`
- d) All of the above

Answer: d) All of the above

13. Which loop structure is best for reading a file line by line?

- a) `for`
- b) `while`
- c) `do-while`
- d) `foreach`

Answer: b) `while`

14. What will be the output of the following code?

```
for ($i = 0; $i < 3; $i++) {  
    for ($j = 0; $j < 2; $j++) {  
        echo "*";  
    }  
}
```

- a) **
- b) ****
- c) *****
- d) ***

Answer: c) *****

15. What is the correct syntax for a `do-while` loop?

- a) `do { statements } while (condition);`
- b) `do while (condition) { statements };`
- c) `do { statements } while condition;`
- d) `do (statements) while (condition);`

Answer: a) `do { statements } while (condition);`

16. Can a `while` loop execute without a body?

- a) Yes
- b) No

Answer: a) Yes

17. Can you use multiple `for` loops inside each other?

- a) Yes
- b) No

Answer: a) Yes

18. Which loop is best suited for iterating over an array?

- a) `for`
- b) `while`
- c) `do-while`
- d) `foreach`

Answer: d) `foreach`

19. What happens if the loop condition in `for` is omitted?

- a) Compilation error
- b) Infinite loop
- c) No iterations
- d) Executes once

Answer: b) Infinite loop

20. How many times does this loop execute?

```
for ($i = 0; $i < 3; $i--) {  
    echo $i;  
}
```

```
}
```

- a) 3 times
- b) Infinite loop
- c) No output
- d) Syntax error

Answer: b) Infinite loop

21. Which loop structure can iterate through database records?

- a) `while`
- b) `for`
- c) `foreach`
- d) All of the above

Answer: d) All of the above

22. What will be the output of `var_dump(5 > 3 ? "Yes" : "No");?`

- a) `string(3) "Yes"`
- b) Yes
- c) No
- d) Error

Answer: a) `string(3) "Yes"`

23. Can a `for` loop be used with a decrementing counter?

- a) Yes
- b) No

Answer: a) Yes

24. What will be the output of `echo 5 > 3 ? "Yes" : "No";?`

- a) Yes
- b) No

- c) Error
- d) Nothing

Answer: a) Yes

25. Which loop runs faster in PHP?

- a) `for`
- b) `while`
- c) `do-while`
- d) It depends on the scenario

Answer: d) It depends on the scenario

26. Can we use a `for` loop inside a `while` loop?

- a) Yes
- b) No

Answer: a) Yes

27. Can a `do-while` loop execute zero times?

- a) Yes
- b) No

Answer: b) No

28. What is the output of the following code?

```
$i = 0;
while ($i++ < 3) {
    echo $i;
}
```

- a) 012
- b) 123

- c) 234
- d) Error

Answer: b) 123

29. Can a loop be written inside a function?

- a) Yes
- b) No

Answer: a) Yes

30. What is the default step value in a `for` loop?

- a) 0
- b) 1
- c) -1
- d) 2

Answer: b) 1

1. What is the purpose of the `break` statement in PHP?

- a) To exit a loop
- b) To continue to the next iteration of a loop
- c) To terminate the script
- d) To pause execution

Answer: a) To exit a loop

2. Which statement is used to skip the current iteration and move to the next one?

- a) `exit`
- b) `break`
- c) `continue`
- d) `return`

Answer: c) `continue`

3. What does the `exit()` function do?

- a) Terminates the current script
- b) Breaks out of a loop
- c) Skips the current iteration
- d) Restarts the script

Answer: a) Terminates the current script

4. What will be the output of the following code?

```
for ($i = 0; $i < 5; $i++) {  
    if ($i == 3) break;  
    echo $i;  
}
```

- a) 012
- b) 01234
- c) 1234
- d) 0123

Answer: a) 012

5. What happens when `break` is used inside a nested loop?

- a) It exits all loops
- b) It only exits the innermost loop
- c) It exits the outermost loop
- d) It throws an error

Answer: b) It only exits the innermost loop

6. What will be the output of the following code?

```
for ($i = 1; $i <= 5; $i++) {  
    if ($i == 3) continue;  
    echo $i;  
}
```

- a) 12345
- b) 1245
- c) 1345
- d) 012

Answer: b) 1245

7. What will happen if `exit()` ; is used in a loop?

- a) The loop will stop execution immediately
- b) The loop will continue to the next iteration
- c) The script will terminate completely
- d) The loop will break and continue after it

Answer: c) The script will terminate completely

8. What is the difference between `break` and `exit`?

- a) `break` stops execution of a script, `exit` stops execution of a loop
- b) `break` stops execution of a loop, `exit` stops execution of the script
- c) `break` and `exit` both stop script execution
- d) There is no difference

Answer: b) `break` stops execution of a loop, `exit` stops execution of the script

9. What will be the output of the following code?

```
$i = 1;
while ($i <= 5) {
    if ($i == 3) break;
    echo $i;
    $i++;
}
```

- a) 12345
- b) 12
- c) 1345
- d) 1245

Answer: b) 12

10. Can `continue` be used outside a loop?

- a) Yes
- b) No

Answer: b) No

11. What will happen if `break` is used outside a loop?

- a) It will break the program
- b) It will work normally
- c) It will throw an error
- d) It will do nothing

Answer: c) It will throw an error

12. What will be the output of the following code?

```
for ($i = 1; $i <= 5; $i++) {  
    if ($i == 4) exit;  
    echo $i;  
}
```

- a) 12345
- b) 1234
- c) 123
- d) No output

Answer: c) 123

13. What is the return type of the `exit()` function?

- a) String
- b) Integer
- c) Void
- d) Boolean

Answer: c) Void

14. What will be the output of the following code?

```
for ($i = 1; $i <= 5; $i++) {  
    echo $i;  
    exit;  
}
```

- a) 12345
- b) 1
- c) 5
- d) No output

Answer: b) 1

15. How can `exit()` be used to display a message?

- a) `exit("Error occurred!");`
- b) `exit = "Error occurred!";`
- c) `exit; "Error occurred!";`
- d) `exit -> "Error occurred!";`

Answer: a) `exit("Error occurred!");`

16. Which of the following is true about `exit()`?

- a) It can accept an optional status code
- b) It can accept a string message
- c) It stops script execution immediately
- d) All of the above

Answer: d) All of the above

17. What will be the output of the following code?

```
$i = 0;  
while ($i < 5) {  
    if ($i == 2) continue;  
    echo $i;  
    $i++;  
}
```

}

- a) 01234
- b) 0134
- c) 024
- d) Infinite loop

Answer: d) Infinite loop (since `$i` is not incremented when `$i == 2`)

18. Can `exit()` be used inside a function?

- a) Yes
- b) No

Answer: a) Yes

19. What is the difference between `die()` and `exit()`?

- a) `die()` can only stop a loop
- b) `exit()` can only be used in functions
- c) They are identical
- d) `die()` does not stop script execution

Answer: c) They are identical

20. Can `break` be used inside a switch case?

- a) Yes
- b) No

Answer: a) Yes

21. Can `continue` be used inside a `foreach` loop?

- a) Yes
- b) No

Answer: a) Yes

22. Can `break` be used inside a function?

- a) Yes
- b) No

Answer: b) No

23. Can `exit` take a numeric argument?

- a) Yes
- b) No

Answer: a) Yes

24. What happens when `exit(0) ;` is used?

- a) It stops execution normally
- b) It stops execution with an error
- c) It breaks a loop
- d) It does nothing

Answer: a) It stops execution normally

25. What will be the output of the following code?

```
for ($i = 1; $i <= 5; $i++) {  
    if ($i == 3) continue;  
    echo $i;  
    if ($i == 4) break;  
}
```

- a) 1245
- b) 1234
- c) 124
- d) 134

Answer: c) 124

26. Can `exit` be used in an `if` condition?

- a) Yes
- b) No

Answer: a) Yes

27. What happens if `exit(1)` ; is used?

- a) It stops execution with a failure status
- b) It continues execution
- c) It breaks a loop
- d) It does nothing

Answer: a) It stops execution with a failure status

28. Which statement is used to terminate the script?

- a) `break`
- b) `continue`
- c) `exit`
- d) `skip`

Answer: c) `exit`

29. What will `exit("Goodbye")` ; output?

- a) Goodbye
- b) 0
- c) Error
- d) Nothing

Answer: a) Goodbye

30. Can `exit()` be used without parentheses?

- a) Yes
- b) No

Answer: a) Yes

1. What is a function in PHP?

- a) A predefined code block
- b) A variable
- c) A database query
- d) A PHP class

Answer: a) A predefined code block

2. How do you declare a function in PHP?

- a) `define myFunction() { }`
- b) `function myFunction() { }`
- c) `void myFunction() { }`
- d) `new function myFunction() { }`

Answer: b) `function myFunction() { }`

3. How do you call a function in PHP?

- a) `call myFunction();`
- b) `invoke myFunction();`
- c) `myFunction();`
- d) `execute myFunction();`

Answer: c) `myFunction();`

4. What will be the output of the following code?

```
function sayHello() {  
    echo "Hello, World!";  
}  
sayHello();
```

- a) Nothing
- b) Hello, World!
- c) Error
- d) 0

Answer: b) Hello, World!

5. Which keyword is used to return a value from a function?

- a) `exit`
- b) `return`
- c) `output`
- d) `send`

Answer: b) `return`

6. What will be the output of this code?

```
function sum($a, $b) {  
    return $a + $b;  
}  
echo sum(3, 5);
```

- a) 8
- b) 3
- c) 5
- d) Error

Answer: a) 8

7. Can a function return multiple values in PHP?

- a) No
- b) Yes, using an array
- c) Yes, using multiple `return` statements
- d) Yes, using multiple `echo` statements

Answer: b) Yes, using an array

8. What is the default return type of a PHP function?

- a) int
- b) string
- c) void
- d) mixed

Answer: d) mixed

9. How can a function accept default parameters?

- a) `function myFunction($a = 5) { }`
- b) `function myFunction(default $a = 5) { }`
- c) `function myFunction($a -> default 5) { }`
- d) `function myFunction($a: 5) { }`

Answer: a) `function myFunction($a = 5) { }`

10. What will be the output of this code?

```
function greet($name = "Guest") {  
    echo "Hello, $name";  
}  
greet();
```

- a) Error
- b) Hello, Guest
- c) Hello,
- d) Guest

Answer: b) Hello, Guest

11. Which function is used to check if a function exists?

- a) `function_exists()`
- b) `is_function()`
- c) `exists_function()`
- d) `isset()`

Answer: a) `function_exists()`

12. What will be the output of the following?

```
function example() {  
    static $x = 0;  
    $x++;  
    echo $x;  
}  
example();  
example();  
example();
```

- a) 111
- b) 123
- c) 012
- d) Error

Answer: b) 123

13. What is the purpose of `static` variables in functions?

- a) They keep their value between function calls
- b) They are reset after function execution
- c) They are global variables
- d) They act as constants

Answer: a) They keep their value between function calls

14. How do you pass arguments by reference?

- a) `function test(&$var) { }`
- b) `function test(@$var) { }`
- c) `function test(*$var) { }`
- d) `function test(#$var) { }`

Answer: a) `function test(&$var) { }`

15. What will be the output?

```
function add(&$a) {
```

```
$a += 5;  
}  
$x = 10;  
add($x);  
echo $x;
```

- a) 10
- b) 5
- c) 15
- d) Error

Answer: c) 15

16. Which PHP function is used to find the length of a string?

- a) strlen()
- b) count()
- c) str_length()
- d) length()

Answer: a) strlen()

17. How do you define a function with variable arguments?

- a) function myFunc(...\$args) { }
- b) function myFunc(\$args...) { }
- c) function myFunc(\$*args) { }
- d) function myFunc(\$args) { }

Answer: a) function myFunc(...\$args) { }

18. What does `array_map()` do?

- a) Applies a function to each element of an array
- b) Joins two arrays
- c) Removes duplicates from an array
- d) Creates a new array

Answer: a) Applies a function to each element of an array

19. How do you stop a function before it reaches the end?

- a) break
- b) exit
- c) return
- d) stop

Answer: c) return

20. What is the output?

```
function check() {  
    return true;  
}  
echo check() ? "Yes" : "No";
```

- a) Yes
- b) No
- c) Error
- d) 1

Answer: a) Yes

21. What does `str_replace()` do?

- a) Replaces a substring in a string
- b) Removes spaces from a string
- c) Converts a string to uppercase
- d) Splits a string into an array

Answer: a) Replaces a substring in a string

22. What is the purpose of `array_filter()`?

- a) Filters array elements based on a callback function
- b) Removes empty values
- c) Sorts an array
- d) Combines two arrays

Answer: a) Filters array elements based on a callback function

23. What does `is_callable()` do?

- a) Checks if a function can be called
- b) Calls a function
- c) Defines a function
- d) Declares a function

Answer: a) Checks if a function can be called

24. What does `max(10, 20, 30)` return?

- a) 10
- b) 20
- c) 30
- d) Error

Answer: c) 30

25. Which function is used to include a file?

- a) `include()`
- b) `require()`
- c) Both a and b
- d) None

Answer: c) Both a and b

26. What does `implode()` do?

- a) Joins array elements into a string
- b) Splits a string into an array
- c) Replaces array elements
- d) Sorts an array

Answer: a) Joins array elements into a string

1. How are function arguments passed in PHP by default?

- a) By reference
- b) By value
- c) By pointer
- d) By copy

Answer: b) By value

2. What is "call by value" in PHP?

- a) The original variable is modified
- b) A copy of the variable is passed
- c) A reference to the original variable is passed
- d) The function directly accesses global variables

Answer: b) A copy of the variable is passed

3. What is "call by reference" in PHP?

- a) A copy of the variable is passed
- b) A reference to the original variable is passed
- c) The function creates a new variable
- d) The function does not receive any value

Answer: b) A reference to the original variable is passed

4. How do you pass an argument by reference in PHP?

- a) `function example($var&)`
- b) `function example(ref $var)`
- c) `function example(&$var)`
- d) `function example(*$var)`

Answer: c) `function example(&$var)`

5. What happens when a function modifies a variable passed by value?

- a) The original variable remains unchanged
- b) The original variable is modified
- c) The function throws an error
- d) The function does not execute

Answer: a) The original variable remains unchanged

6. What happens when a function modifies a variable passed by reference?

- a) The original variable remains unchanged
- b) The original variable is modified
- c) A new variable is created
- d) The function throws an error

Answer: b) The original variable is modified

7. What will be the output of the following code?

```
function test($x) {  
    $x = $x * 2;  
}  
$val = 5;  
test($val);  
echo $val;
```

- a) 10
- b) 5
- c) Error
- d) 0

Answer: b) 5

8. What will be the output of the following code?

```
function test(&$x) {  
    $x = $x * 2;  
}  
$val = 5;  
test($val);  
echo $val;
```

- a) 10
- b) 5
- c) Error
- d) 0

Answer: a) 10

9. What is the correct syntax for passing multiple arguments by reference?

- a) `function example(&$x, &$y)`
- b) `function example(*$x, *$y)`
- c) `function example($x&, $y&)`
- d) `function example(ref $x, ref $y)`

Answer: a) `function example(&$x, &$y)`

10. Can a function mix call by value and call by reference parameters?

- a) Yes
- b) No

Answer: a) Yes

11. What will be the output of the following?

```
function modify($num) {  
    $num += 10;  
}  
$a = 5;  
modify($a);  
echo $a;
```

- a) 5
- b) 10
- c) 15
- d) Error

Answer: a) 5

12. What will be the output of the following?

```
function modify(&$num) {  
    $num += 10;  
}  
$a = 5;  
modify($a);  
echo $a;
```

- a) 5
- b) 10
- c) 15
- d) Error

Answer: c) 15

13. What does & signify in function parameters?

- a) It indicates a global variable
- b) It passes the argument by reference
- c) It copies the value of the variable
- d) It converts the variable to a string

Answer: b) It passes the argument by reference

14. Which of the following function calls is correct for call by reference?

- a) myFunction(\$var&)
- b) myFunction(ref \$var)
- c) myFunction(&\$var)
- d) myFunction(*\$var)

Answer: c) myFunction(&\$var)

15. What will be the output of the following?

```
function change(&$x) {  
    $x = "PHP";  
}  
$y = "Java";  
change($y);  
echo $y;
```


- a) PHP
- b) Java
- c) Error
- d) Nothing

Answer: a) PHP

16. How does passing by reference affect performance?

- a) It makes function execution slower
- b) It improves performance by avoiding unnecessary copies
- c) It has no impact on performance
- d) It causes memory leaks

Answer: b) It improves performance by avoiding unnecessary copies

17. What will be the output of the following?

```
function increment(&$num) {  
    $num++;  
}  
$value = 10;  
increment($value);  
echo $value;
```

- a) 10
- b) 11
- c) 9
- d) Error

Answer: b) 11

18. Can an array be passed by reference in PHP?

- a) Yes
- b) No

Answer: a) Yes

19. What will be the output of the following?

```
php
CopyEdit
function modifyArray(&$arr) {
    $arr[0] = 100;
}
$numbers = [1, 2, 3];
modifyArray($numbers);
echo $numbers[0];
```

- a) 1
- b) 100
- c) 2
- d) Error

Answer: b) 100

20. What happens if a reference is passed to a function but not used as a reference?

- a) The function modifies the original variable
- b) The function works normally but does not modify the original variable
- c) The function throws an error
- d) The function does not execute

Answer: b) The function works normally but does not modify the original variable

21. Can an object be passed by reference in PHP?

- a) Yes
- b) No

Answer: a) Yes

22. Which is faster, call by value or call by reference?

- a) Call by value
- b) Call by reference
- c) Both are the same

Answer: b) Call by reference

23. What happens if you pass a constant by reference?

- a) It throws an error
- b) The function works normally
- c) The constant gets modified
- d) The function ignores it

Answer: a) It throws an error

24. Which function returns a reference to a global variable?

- a) global
- b) &\$GLOBALS
- c) return &\$var;
- d) global_return()

Answer: c) return &\$var;

25. What does the following function return?

```
php
CopyEdit
function &getValue() {
    static $val = 10;
    return $val;
}
$a = &getValue();
$a = 20;
echo getValue();
```

- a) 10
- b) 20
- c) 30
- d) Error

Answer: b) 20

String Manipulation Functions in PHP

1. Which function is used to get the length of a string in PHP?

- a) `strlen()`
- b) `length()`
- c) `strlenlength()`
- d) `count()`

Answer: a) `strlen()`

2. What will `strtolower("HELLO")` return?

- a) hello
- b) HELLO
- c) Hello
- d) Error

Answer: a) hello

3. Which function is used to convert the first character of a string to uppercase?

- a) `strtoupper()`
- b) `ucfirst()`
- c) `capitalize()`
- d) `strcapitalize()`

Answer: b) `ucfirst()`

4. What is the output of `substr("abcdef", 2, 3)`?

- a) abc
- b) cde
- c) cdef
- d) def

Answer: b) cde

5. Which function is used to replace a substring within a string?

- a) `replace()`
- b) `str_replace()`
- c) `substr_replace()`
- d) `change()`

Answer: b) `str_replace()`

6. What will `strrev("PHP")` return?

- a) PHP
- b) HPP
- c) PPH
- d) HPP

Answer: c) PPH

7. Which function removes whitespace from both sides of a string?

- a) `remove_spaces()`
- b) `trim()`
- c) `strim()`
- d) `strip()`

Answer: b) `trim()`

8. What does `explode(" ", "Hello World")` return?

- a) `["Hello", "World"]`
- b) `"Hello World"`
- c) `["HelloWorld"]`
- d) Error

Answer: a) `["Hello", "World"]`

9. What is the output of `strcmp("apple", "banana")`?

- a) -1
- b) 0
- c) 1
- d) Error

Answer: a) -1

10. Which function joins an array into a string?

- a) `implode()`
- b) `join()`
- c) `concat()`
- d) `merge()`

Answer: a) `implode()`

Mathematical Functions in PHP

11. Which function returns the absolute value of a number?

- a) `abs()`
- b) `absolute()`
- c) `ceil()`
- d) `floor()`

Answer: a) `abs()`

12. What will `pow(2, 3)` return?

- a) 6
- b) 8
- c) 9
- d) 16

Answer: b) 8

13. What does `sqrt(16)` return?

- a) 2
- b) 4
- c) 8
- d) 16

Answer: b) 4

14. What is the result of `round(4.7)` ?

- a) 4
- b) 5
- c) 4.7
- d) Error

Answer: b) 5

15. Which function generates a random number?

- a) `random()`
- b) `rand()`
- c) `randomize()`
- d) `generate_rand()`

Answer: b) `rand()`

16. What does `ceil(4.3)` return?

- a) 4
- b) 5
- c) 4.3
- d) Error

Answer: b) 5

17. What is the output of `floor(4.9)`?

- a) 4
- b) 5
- c) 4.9
- d) Error

Answer: a) 4

18. Which function returns the sine of an angle in radians?

- a) `sin()`
- b) `sine()`
- c) `cos()`
- d) `tan()`

Answer: a) `sin()`

19. What is the result of `log(1)`?

- a) 1
- b) 0
- c) -1
- d) Error

Answer: b) 0

20. What does `min(3, 5, 7, 1, 9)` return?

- a) 1
- b) 3
- c) 5
- d) 9

Answer: a) 1

Date and Time Functions in PHP

21. Which function returns the current timestamp?

- a) `date()`
- b) `time()`
- c) `now()`
- d) `timestamp()`

Answer: b) `time()`

22. What will `date("Y")` return?

- a) Full date
- b) Current year
- c) Current time
- d) Error

Answer: b) Current year

23. Which function formats a timestamp into a human-readable date?

- a) `format_date()`
- b) `strtotime()`
- c) `date()`
- d) `timestamp_format()`

Answer: c) `date()`

24. What does `mktime(0, 0, 0, 1, 1, 2024)` return?

- a) Timestamp for January 1, 2024
- b) 0

- c) Error
- d) January 1, 1970

Answer: a) Timestamp for January 1, 2024

25. What will `strtotime("next Sunday")` return?

- a) The timestamp of the next Sunday
- b) "Next Sunday"
- c) The first Sunday of the year
- d) Error

Answer: a) The timestamp of the next Sunday

26. Which function returns the number of days in a given month?

- a) `days_in_month()`
- b) `date_days()`
- c) `cal_days_in_month()`
- d) `month_days()`

Answer: c) `cal_days_in_month()`

27. What is the result of `date("d-m-Y", time())`?

- a) Current date in dd-mm-yyyy format
- b) Current timestamp
- c) Current day
- d) Error

Answer: a) Current date in dd-mm-yyyy format

28. What does `checkdate(2, 30, 2024)` return?

- a) true
- b) false

Answer: b) false

29. Which function adds a time interval to a timestamp?

- a) `strtotime("+1 day")`
- b) `add_time("+1 day")`
- c) `time_add("+1 day")`
- d) `date_modify("+1 day")`

Answer: a) `strtotime("+1 day")`

30. Which function returns the name of the day for a given date?

- a) `date("l")`
- b) `date("d")`
- c) `day_name()`
- d) `get_day()`

Answer: a) `date("l")`

31. What does `strtoupper("php")` return?

- a) php
- b) PHP
- c) PhP
- d) pHp

Answer: b) PHP

32. What does `strpos("Hello World", "World")` return?

- a) 5
- b) 6
- c) 7
- d) Error

Answer: b) 6

33. What does `str_repeat("abc", 3)` return?

- a) abcabc
- b) abcabcabc
- c) abc
- d) Error

Answer: b) abcabcabc

34. Which function converts a string to an array?

- a) `split()`
- b) `explode()`
- c) `string_to_array()`
- d) `str_split()`

Answer: d) `str_split()`

35. What will `substr("abcdef", -3)` return?

- a) abc
- b) def
- c) cdef
- d) Error

Answer: b) def

Mathematical Functions in PHP

36. What is the result of `round(3.5)`?

- a) 3
- b) 4
- c) 3.5
- d) Error

Answer: b) 4

37. What does `fmod(10, 3)` return?

- a) 1
- b) 2
- c) 3
- d) 0

Answer: b) 1

38. What is the output of `max(5, 10, 15, 3, 8)`?

- a) 3
- b) 5
- c) 10
- d) 15

Answer: d) 15

39. What is the return type of `is_nan(sqrt(-1))`?

- a) true
- b) false
- c) 1
- d) 0

Answer: a) true

40. What does `rand(1, 10)` return?

- a) A random number between 1 and 10
- b) A fixed number
- c) Always 10
- d) Always 1

Answer: a) A random number between 1 and 10

Date and Time Functions in PHP

41. What does `date("H:i:s")` return?

- a) Current date
- b) Current time in HH:MM:SS format
- c) Unix timestamp
- d) Time in YYYY-MM-DD format

Answer: b) Current time in HH:MM:SS format

42. What is the default time zone in PHP if not set?

- a) UTC
- b) GMT
- c) EST
- d) PST

Answer: a) UTC

43. What does `strtotime("last Monday")` return?

- a) The timestamp of the last Monday
- b) The next Monday's timestamp
- c) Today's timestamp
- d) Error

Answer: a) The timestamp of the last Monday

44. What does `date("t")` return?

- a) The current timestamp
- b) The number of days in the current month
- c) The current day
- d) The current month

Answer: b) The number of days in the current month

45. What is the output of `date("z")` ?

- a) The current hour
- b) The current day of the year (0 to 365)
- c) The current minute
- d) The current second

Answer: b) The current day of the year (0 to 365)

46. What does `gmdate("Y-m-d")` return?

- a) The current date in the UTC timezone
- b) The current date in local time
- c) The current timestamp
- d) An error

Answer: a) The current date in the UTC timezone

47. What is the purpose of `set_time_limit()` function?

- a) Sets the maximum execution time for a script
- b) Sets the time zone
- c) Sets the system clock
- d) Sets the date format

Answer: a) Sets the maximum execution time for a script

48. What will `date_default_timezone_set("Asia/Kolkata")` do?

- a) Set the time zone to Kolkata
- b) Set the date format to Indian standard
- c) Set the system time
- d) Error

Answer: a) Set the time zone to Kolkata

49. What does `date("w")` return?

- a) The current day of the year
- b) The current week number of the year
- c) The current month number
- d) The current hour

Answer: b) The current week number of the year

50. What does `getdate() ["month"]` return?

- a) The current month name
- b) The current month number
- c) The current timestamp
- d) The current year

Answer: a) The current month name