1.What does it mean for PHP to be an interpreted language? How does this differ from a compiled language?

PHP is an interpreted language, meaning the PHP code is executed line by line at runtime by the PHP interpreter. This differs from compiled languages where the entire source code is translated into machine code before execution.

2.Discuss the advantages and disadvantages of using an interpreted language like PHP over a compiled language.

Advantages:

Easier to develop and debug

Platform independence

Faster development cycle

Dynamic typing

Disadvantages:

Generally slower execution compared to compiled languages

Source code is exposed

Requires the interpreter to be present on the server

PHP Arrays

3.Explain the difference between indexed arrays and associative arrays in PHP. Provide examples of both.

Indexed arrays use numeric keys, while associative arrays use string keys.

// Indexed array $fruits = ['apple', 'banana', 'red'];

// Associative array $person = ['name' => 'Angelo', 'age' => 20, 'city' => 'New York'

4.How can you iterate through an associative array in PHP? Provide code demonstrating at least two different methods.

Method 1: foreach loop

foreach ($person as $key => $value)

```
{ echo "$key: $value\n"; }
```

Method 2: While loop

```
while (list($key, $value) = each($person))
{ echo "$key: $value\n"; }
```

5What will the following code output, and why?

```
$arr = [1 => 'apple', 2 => 'banana', 'key' => 'orange'];
echo $arr[1];  Outputs: apple
echo $arr['key']; Outputs: orange
```

PHP Classes

6.Define a class in PHP that represents a Car with properties like make, model, and year. Include methods for starting the car and stopping it.

```
class Car {
private $make;
private $model;
private $year; private
$isRunning = false;

public function __construct($make, $model, $year)
{ $this->make = $make;
$this->model = $model;
$this->year = $year; }

public function start() {
$this->isRunning = true;
echo "The car is starting.\n"; }

public function stop() {
$this->isRunning = false;
echo "The car is stopping.\n"; } }
```

7.What is the purpose of the __construct() method in PHP classes? How is it different from regular methods?

The __construct() method is a special method called automatically when a new object is created. It's used for initializing object properties. Unlike regular methods, it's called implicitly when using the new keyword.

PHP Variables

8.What are the different types of variables in PHP?

Scalar types: integer, float, string, Boolean. Compound types: array, object

Special types: NULL

9.What does the $this keyword refer to in PHP? Provide an example of how it is used within a class.

$this refers to the current object instance within a class method.

```
class Person { private $name;
public function setName($name)
 { $this->name = $name; }
public function getName()
{ return $this->name; } }
```

PHP If Statements
10.Write an if-else statement that checks whether a number is positive, negative, or zero, and prints the appropriate message.

```
$number = 5;
if ($number > 0)
{ echo "The number is positive."; }
elseif ($number < 0)
{ echo "The number is negative."; }
else { echo "The number is zero."; }
```

11.Explain the difference between == and === in PHP if statements. Provide an example where they would behave differently.

Difference between == and === in PHP: == checks for equality of value, while === checks for equality of both value and type.

```
$a = 5;
$b = "5";
if ($a == $b)
{ echo "Equal with ==\n"; // This will print }
if ($a === $b)
{ echo "Equal with ===\n"; // This will not print }
```

12.What is a ternary operator in PHP? Rewrite the following if-else statement using the ternary operator:
Copy code

```
if ($a > $b) {
   echo "a is greater";
} else {
  echo "b is greater";
}
```

```
echo ($a > $b) ? "a is greater" : "b is greater";
```

13.Write an if-else statement in PHP that checks whether a variable contains a string with more than 10 characters and prints a message accordingly.
instanceof Operator

```
$string = "Hello, World!";
if (strlen($string) > 10)
{ echo "The string has more than 10 characters."; }
else { echo "The string has 10 or fewer characters."; }
```

14.What does the instanceof operator do in PHP? Provide an example where it is used.

The instanceof operator is used to determine if an object is an instance of a specific class or interface.

```
class Animal {}
class Dog extends Animal {}
$dog = new Dog();
if ($dog instanceof Animal)
{ echo "The dog is an Animal."; }
```

15. Why might you use instanceof in an object-oriented PHP application? Give a scenario where this would be beneficial. It might be good for type checking. in a function that accepts different types of objects, you can use instanceof to determine the specific type and act accordingly.

16. What will the following code output, and why? Will return a bool (true) because The output is true because $dog is an instance of Dog, which extends Animal, so it is also considered an instance of Animal.

```
class Animal {}
22. class Dog extends Animal {}
23. $dog = new Dog();
24. var_dump($dog instanceof Animal)
```