* What is **object-oriented programming** in general terms?

Object-oriented programming (OOP) is a computer programming model that organizes software design around data, or objects, rather than functions and logic. An object can be defined as a data field that has unique attributes and behavior.

* What is a **class**?

Classes are the blueprints of objects. One of the big differences between functions and classes is that a class contains both data (variables) and functions that form a package called an: ‘object’.

Class is a programmer-defined data type, which includes local methods and local variables.

Class is a collection of objects. Object has properties and behavior.

* What is an **object**?

An Object is an individual instance of the data structure defined by a class. We define a class once and then make many objects that belong to it. Objects are also known as instances.

* What is an **instance**?

An instance is an object that has been created from an existing class. Creating an object from an existing class is called instantiating the object. To create an object out of a class, the new keyword must be used. Classes should be defined prior to instantiation.

* What is a **property**?

Properties are variables. Classes can have variables within it. Those variables are called properties. A property is a normal PHP variable which is in any data type (integer, string, array, object, etc).

* What is a **method**?

In Object Oriented Programming in PHP, methods are functions inside classes. Their declaration and behavior are almost similar to normal functions, except their special uses inside the class.

* What is the difference between a **function** and a **method**?

Both are used interchangeably, but function is the terminology used in structural languages and method is the terminology used in Object Oriented Langauages. ... The difference is that a method is used to describe functions defined in classes that are used with instances of those classes.

* What is a **constructor**?

A constructor allows you to initialize an object's properties upon creation of the object. If you create a \_\_construct() function, PHP will automatically call this function when you create an object from a class.

* What is the difference between a **class, an** **object** and an **instance**?

A class is a blueprint which you use to create objects. An object is an instance of a class - it's a concrete 'thing' that you made using a specific class. So, 'object' and 'instance' are the same thing, but the word 'instance' indicates the relationship of an object to its class.

* What do we understand about the concept of **encapsulation**?

Encapsulation is one of the fundamentals of OOP (object-oriented programming). It refers to the bundling of data with the methods that operate on that data. Encapsulation is used to hide the values or state of a structured data object inside a class, preventing unauthorized parties' direct access to them.

* What do we understand about the concept of **abstraction**?

Abstract classes are the classes in which at least one method is abstract.In PHP are declared with the help of abstract keyword. Use of abstract classes are that all base classes implementing this class should give implementation of abstract methods declared in parent class.

* What do we understand about the concept of **inheritance**?

When a class derives from another class. The child class will inherit all the public and protected properties and methods from the parent class. An inherited class is defined by using the extends keyword.

* What do we understand about the concept of **polymorphism**?

The word polymorphism is used in various contexts and describes situations in which something occurs in several different forms. In computer science, it describes the concept that objects of different types can be accessed through the same interface. ... It is one of the core concepts of object-oriented programming

* What do we understand about the concept of **Overload**?

In PHP, overloading means that you are able to add object members at runtime, by implementing some of the \_\_magic methods, like \_\_get , \_\_set , \_\_call , \_\_callStatic . You load objects with new members. Overloading in PHP provides means to dynamically "create" properties and methods.

* What do we understand about the concept of **Override**?

In function overriding, both parent and child classes should have same function name with and number of arguments. It is used to replace parent method in child class. The purpose of overriding is to change the behavior of parent class method. The two methods with the same name and same parameter is called overriding.

* What differences exist between the concept of **Overload** and **Override**?

The most basic difference is that overloading is being done in the same class while for overriding base and child classes are required. Overriding is all about giving a specific implementation to the inherited method of parent class.

* What is a **static class**?

A static class in PHP is a type of class which is instantiated only once in a program. It must contain a static member (variable) or a static member function (method) or both. The variables and methods are accessed without the creation of an object, using the scope resolution operator(::)

* Look for 3 advantages over **object-oriented** **programming** compared to other programming paradigms

1. Modularity for easier troubleshooting
2. Reuse of code through inheritance
3. Flexibility through polymorphism

* Look for **disadvantages of this paradigm**.

1. Steep learning curve.
2. Larger program size.
3. **Slower programs.**