OOPS concepts – BY ROHIT

1) What is meant by Object Oriented Programming?

Procedural programming is about writing procedures or functions that perform operations on the data, while object-oriented programming is about creating objects that contain both data and functions.

Object-oriented programming has several advantages over procedural programming:

- OOP is faster and easier to execute
- OOP provides a clear structure for the programs
- OOP helps to keep the C++ code DRY "Don't Repeat Yourself", and makes the code easier to maintain, modify and debug
- OOP makes it possible to create full reusable applications with less code and shorter development time

Tip: The "Don't Repeat Yourself" (DRY) principle is about reducing the repetition of code.

When the individual objects are created, they inherit all the variables and functions from the class.

2) What is a Class?

Class is a template for a set of objects that share a common structure and a common behavior.

3) What is an Object?

Object is an instance of a class. It has state, behavior and identity. It is also called as an instance of a class.

4) What is an Instance?

An instance has state, behavior and identity. The structure and behavior of similar classes are defined in their common class. An instance is also called as an object.

5) What are the core OOP's concepts?

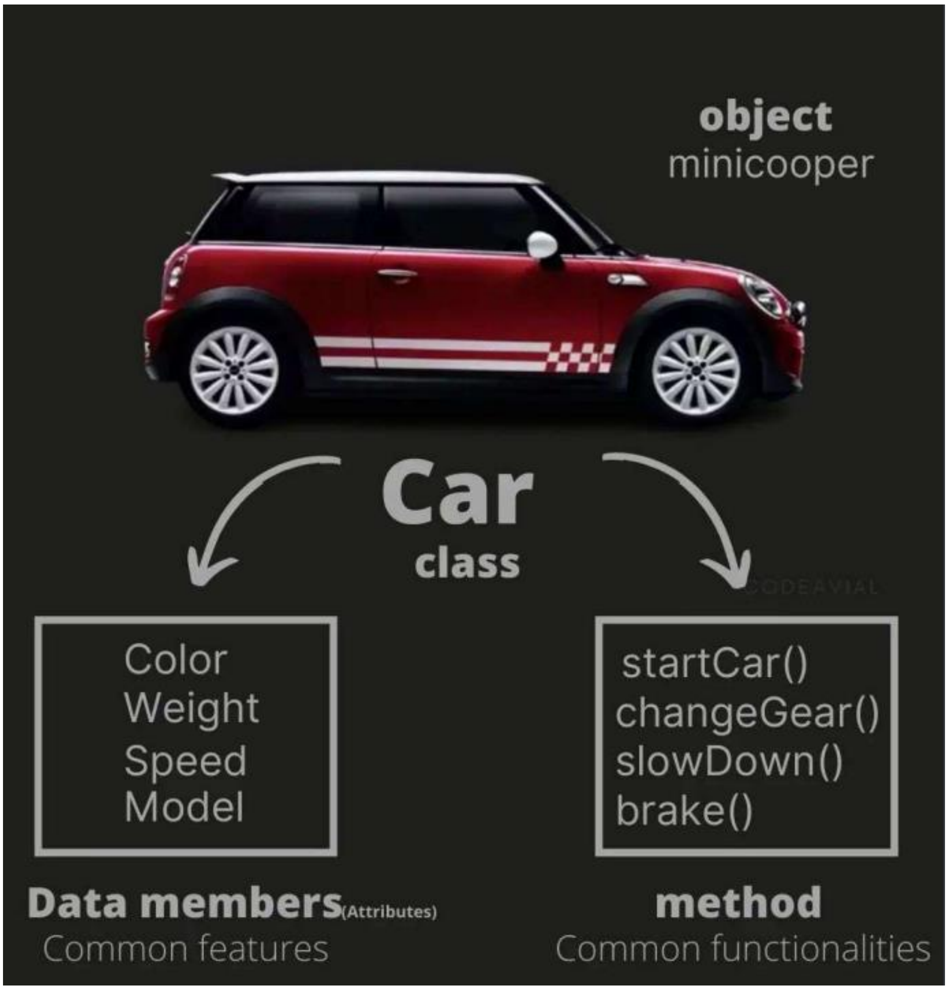
Abstraction, Encapsulation, Inheritance and Polymorphism are the core OOP's concept

6) What is meant by abstraction?

Abstraction is the process of hiding the internal details of an application from the outer world. Abstraction is used to describe things in simple terms. It's used to create a boundary between the application and the client programs.

There are two types of abstraction.

1. Data Abstraction
2. Process Abstraction



7) What is meant by Encapsulation?

Encapsulation is defined as the wrapping up of data under a single unit. It is the mechanism that binds together code and the data it manipulates. Another way to think about encapsulation is, that it is a protective shield that prevents the data from being accessed by the code outside this shield.

Binding denotes association of a name with a class.

8) What is meant by Inheritance?

In C++, it is possible to inherit attributes and methods from one class to another. We group the "inheritance concept" into two categories:

- **derived class** (child) - the class that inherits from another class
- **base class** (parent) - the class being inherited from

To inherit from a class, use the : symbol.

Inheritance is a relationship among classes, wherein one class shares the structure or behavior defined in another class. This is called Single Inheritance. If a class shares the structure or behavior from multiple classes, then it is called Multiple Inheritance. Inheritance defines “is-a” hierarchy among classes in which one subclass inherits from one or more generalized superclasses.

9) What is meant by Polymorphism?

The word “polymorphism” means having many forms. In simple words, we can define polymorphism as the ability of a message to be displayed in more than one form.

A real-life example of polymorphism is a person who at the same time can have different characteristics. A man at the same time is a father, a husband, and an employee.

10) What is a base class?

Base class is the most generalized class in a class structure. Most applications have such root classes. In Java, Object is the base class for all classes.

11) What is a subclass?

Subclass is a class that inherits from one or more classes

12) What is a superclass?

superclass is a class from which another class inherits.

13) What is meant by static binding?

Static binding is a binding in which the class association is made during compile time. This is also called as Early binding.

14) What is meant by Dynamic binding?

Dynamic binding is a binding in which the class association is not made until the object is created at execution time. It is also called as Late binding.

15) What is Overloading?

Adding a new method with the same name in same/derived class but with different number/types of parameters. It implements Polymorphism. What is Overriding A process of creating different implementation of a method having a same name as base class, in a derived class. It implements Inheritance.

16) What is Shadowing?

When the method is defined as Final/sealed in base class and not override able and we need to provide different implementation for the same. This process is known as shadowing, uses shadows/new keyword.

17) What is Inheritance?

It is a process of acquiring attributes and behaviors from another object (normally a class or interface).

3. What are some major Object Oriented Programming languages?

- *Java*
- *C++*
- *Javascript*
- *Python*
- *PHP*