Object Oriented Programming

1. What is OOP in general terms?
   OOP is a program paradigm or concept of which the app we are building is encapsulated into objects and functions. Each object contains functions (called methods) and data (called properties) and in fact is the base or a unit code of our future app. In that way we can reuse code, organising the app structure in a more cleaner and understandable way.

2. What is a class?
   A class in OOP is like a mould, or the base on which the object is built. It contains data and functions.

3. What is an object?
   Object is a way to gather code together and keep information that later can be reused over and over again.

4. What is an instance?
   An instance is a specific realisation of any object. In that way we can create a new object that has common properties and functions from a class.

5. What is property?
   As said before, property is some sort of data, put in variables for example that a class has.

6. What are methods?
   Methods are actions or functions that the class contains and later will be passed to the instance of the class.

7. What is the difference between a function and a method?
   Mostly semantical, while functions are more general, and can be declared in global scope, methods are class functions, being called inside the class.

8. What is a constructor?
   The constructor is a special method in the class that sets some defined properties once the class is instantiated.

9. What is the difference between a class, an object and an instance?
   Firstly, it's important to understand that all of the above are by definition objects. Now if we go deeper into the relation between them, that is when the beauty kicks in.
   As we said, a class in the skeleton or a blueprint for the later come object that for this purpose is an instance. We call it an instance to imply their relationship.

10. What are the four pillars of OOP?
    Encapsulation - put in simple words, is the ability to gather together common properties and functions into a unit and reuse them for every later object.

    Abstraction - is showing only the necessary properties or function, not exposing redundant or unimportant functionalities or data.

    Inheritance - the idea in which each instance inherits its abilities, properties and functionality from its parent class.

    Polymorphism - is the ability to perform the same actions in different instances of the same class and receive different outcomes thanks to the same interface the object has.

11. What is the concept of the override method?
    As we inherit the same properties and methods from a parent class to its daughter, we can set a property or a method, having the same name, but do something totally different. This is thanks to the ability to override these properties without harming the class structure.

12. What is the concept of overload methods?
    When multiple methods are defined with the same name, the concept is called overloading. In such cases we need to differentiate them by changing the parameters and using the magic method __call() instead of the method itself.

13. What is a static class?
    A static class allows us to execute methods and get data without creating an instance of the class but calling them directly on the class itself.

14. Advantages of OOP paradigm:
    - allowing to break the code into unit code size and reuse it.
    - it is more understandable when trying to pick up on a project and work together with other developers.
    - code maintenance.

15. Disadvantages of OOP:
    - app size might be bigger.
    - slow on performance.
    - learning curve and takes time to implement.