Q1. What is the purpose of Python's OOP?
To make the program more structured and systematic and to make the code reusable.

Q2. Where does an inheritance search look for an attribute?
The inheritance search in python starts searching for an attribute going from the instance object to the top searching in all the classes and the superclasses of that instance object. It starts searching from top to bottom in a left to right manner.

Q3. How do you distinguish between a class object and an instance object?
Classes are factories of instances. An instance is itself an individual entity containing data for a particular object.

Q4. What makes the first argument in a class's method function special?
The first argument in a class's method function special is "self". Suppose we create a class and define a method in that class and when we create an instance of that class and if that instance access the method we created. The instance will apply that method on itself with the help of this special first argument. It is a special argument because by using it we don't need to explicitly mention on which instance we want to apply a particular method.

Q5. What is the purpose of the __init__ method?
It is known as the constructor method. It simply assigns values to the data members of the class and is itself called when we create a new instance of the class.

Q6. What is the process for creating a class instance?

Firstly, we define a class

```
Class merds    #defining a class with classname merds
I = merds()     # I is the instance of class merds
```

Q7. What is the process for creating a class?
We can create a class by using the keyword "class" and then defining the class name.
Example: class <classname>

Q8. How would you define the superclasses of a class?
Superclasses of a class is defined in the following way;
Class C1(C2, C3)   # here C2 and C3 are the superclasses.