Q1. What is the purpose of Python's OOP?

Object oriented programming is mainly used to implement the objects , class, inheritance, polymorphisms, encapsulation, in the programs

Q2. Where does an inheritance search look for an attribute?

Inheritance search looks for an attribute first instance in object then in the class instant and all other higher superclasses by default it goes from left to right .

Q3. How do you distinguish between a class object and an instance object?

Class is template for creating objects in the program and object is an instance of class.Class does not allocate memory space but objects allocate memory space.

Q4. What makes the first argument in a class’s method function special?

The first argument in a class method function is object itself

Q5. What is the purpose of the init method?

Init method is mainly used to initialize the object state

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

When creating the object we are creating the instance of the class so instantiating the class .The new operator requires the single , postfix argument :call the constructor , the name of the constructor is the name of the class the instantiate . The constructor initializes the new object .

Q7. What is the process for creating a class?

Classes are created in a new local namespace and all the attributes are defined . Attributes may be data or function

Q8. How would you define the superclasses of a class?

The class whose properties inheritance with another class is known as superclasses

Q9. What is the relationship between classes and modules?

Modules is collection of method and constant they cannot generate instances , class can create more instances and have pre instance state

Q10. How do you make instances and classes?

When creating the object we are creating the instance of the class so instantiating the class .The new operator requires the single , postfix argument :call the constructor , the name of the constructor is the name of the class the instantiate . The constructor initializes the new object

Q11. Where and how should be class attributes created?

A class attributes is shared by all the instances class To define the class placed the outside the _int _method() .class attributes for storing class contants, track data across all instances, and setting default values for all instances of the class.

Q12. Where and how are instance attributes created?

Instance attributes are defined in the constructor . Defined directly inside a class.

Q13. What does the term "self" in a Python class mean?

The self parameters is refer to the current instances of object and is used to access parameter

Q14. How does a Python class handle operator overloading?

The operator overloading in python means its provide extend the meaning beyond the predefined operational meaning for eg if we use + operator used for merging list we are achieve with + operator is overloaded by int class and str class

Q15. When do you consider allowing operator overloading of your classes?

If we suppose + operator is used to joining list or merge list here + operator overloaded the int class and str class

Q16. What is the most popular form of operator overloading?

The most popular form of operator overloading is + operator it can performed addition and concentration

Q17. What are the two most important concepts to grasp in order to comprehend Python OOP code?

Inheritance and polymorphism are key ingredients for designing ,robust, easy to maintain software

Q18. Describe three applications for exception processing.

Syntax error is similar to grammar because if any symbols are missing the code does not execute

Exception may occurs the syntactically correct code block run time if eg divided by zero its common exception occurs in program

Logical error its may occurs when logic are not correct

Q19. What happens if you don't do something extra to treat an exception?

If don't treat an exception the program will forced to terminate abruptly

Q20. What are your options for recovering from an exception in your script?

You can also provide a generic except clause, which handles any exception.The else-block is a good place for code that does not need the try: block's protection

Q21. Describe two methods for triggering exceptions in your script.

Try this method catches the exception raised by program
Raise this method trigger the exception manually using custom exceptions

Q22. Identify two methods for specifying actions to be executed at termination time, regardless of whether or not an exception exists.

Try method and final method

Q23. What is the purpose of the try statement?

Try statements uses to tested the block of code for errors

Q24. What are the two most popular try statement variations?

The variations are try except , try except else , try except final

Q25. What is the purpose of the raise statement?

Raise statement is to stop the normal exception and transfer the control to exception handlers

Q26. What does the assert statement do, and what other statement is it like?

Assert statement has a condition if the condition is false the program halts and give the assertion error , the other method of assert statement the have an optional message if the condition is false the program stop and give the assertion error along with optional error message

Q27. What is the purpose of the with/as argument, and what other statement is it like?

In python with argument replace the try-catch, block , in concise shorthand and mostly it ensure resources closing right after the processing them

Q28. What are *args, **kwargs?

args in function definitions in python is used to pass a variable number of arguments to a function. It is used to pass a non-key worded, variable-length argument list.

***kwargs* in function definitions in python is used to pass a keyworded, variable-length argument list. We use the name *kwargs* with the double star. The reason is that the double star allows us to pass through keyword arguments

Q29. How can I pass optional or keyword parameters from one function to another?

Ether user can pass the values or pretend the function where the values are specified and also it can be done passing the operational parameters or by pass the required parameter

Q30. What are Lambda Functions?

Lambda functions are inline function or anonymous function , this lambda functions can be used without a defined name

Lambda functions syntax lambda arguments : expressions

Q31. Explain Inheritance in Python with an example?

Inheritance is the capability of one class to derive or inherit the properties from another class.

Q32. Suppose class C inherits from classes A and B as class C(A,B).Classes A and B both have their own versions of method func(). If we call func() from an object of class C, which version gets invoked?

Q33. Which methods/functions do we use to determine the type of instance and inheritance?

Use isinstance() to check an instance's type: isinstance(obj, int) will be True only if obj.__class__ is int or some class derived from int .

Use issubclass() to check class inheritance: issubclass(bool, int) is True since bool is a subclass of int .

Q34.Explain the use of the 'nonlocal' keyword in Python.

The nonlocal keyword is used to work with variables inside nested functions, where the variable should not belong to the inner function.

Q35. What is the global keyword?

Global keyword allows us to modify variables outside the current loop and it is used to create global variables and make changes in the local variable with the local context.