Q1. Which two operator overloading methods can you use in your classes to support iteration?

Ans :

* Lists.
* Dictionary.
* Tuples.
* Class.

Q2. In what contexts do the two operator overloading methods manage printing?

Ans : In unary operator function, no arguments should be passed. It works only with one class objects. It is a overloading of an operator operating on a single operand.  
...

* Overloading unary operator.
* Overloading binary operator.
* Overloading binary operator using a friend function.

Q3. In a class, how do you intercept slice operations?

Ans : \_\_getitem\_\_ method can be implement in a class, and the behavior of slicing can be defined inside it. Parameter: slice() : constructor to create slice object. start: An integer number specifying start index.It is optional and default is 0.

Q4. In a class, how do you capture in-place addition?

Ans : Python provides the operator x += y to add two objects in-place by calculating the sum x + y and assigning the result to the first operands variable name x . You can set up the in-place addition behavior for your own class by overriding the magic “dunder” method \_\_iadd\_\_(self, other) in your class definition.

Q5. When is it appropriate to use operator overloading?

Ans : Now, if the user wants to make the operator “+” to add two class objects, the user has to redefine the meaning of the “+” operator such that it adds two class objects. This is done by using the concept “Operator overloading”