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

**Ans. \_\_iter\_\_(self) & \_\_next\_\_(self).**

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

**Ans. It depends upon their behaviors which have been defined in python.**

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

**Ans. Using \_\_setslice\_\_ and \_\_delslice\_\_**

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

**Ans. import operator # importing operator to handle operator operations**

**x = operator.iadd(2, 3); # using iadd() to add and assign value**

Q5. When is it appropriate to use operator overloading?

**Ans. When you have a lot of demand for that functionality and where this feature will have** **a major impact on your code then only use operator overloading.**