Q1. The two operator overloading methods that can be used in classes to support iteration are \_\_iter\_\_() and \_\_next\_\_().

Q2. The two operator overloading methods that manage printing are \_\_str\_\_() and \_\_repr\_\_(). \_\_str\_\_() returns a string representation of an object, while \_\_repr\_\_() returns an unambiguous string representation of an object.

Q3. To intercept slice operations, a class needs to define the \_\_getitem\_\_() method. This method takes the start, stop, and step values of the slice as arguments, and returns the requested elements.

Q4. To capture in-place addition, a class needs to define the \_\_iadd\_\_() method. This method modifies the object in place and returns it.

Q5. Operator overloading should be used when it provides a natural and intuitive way to work with objects of a class. For example, it can be used to make objects behave like numbers or containers. However, it should be used sparingly, as it can make code harder to read and understand if used excessively.