**Collections:**

A collection is a framework, holder, or container designed to store items (objects). It is an interface and cannot create direct objects but can be implemented by child classes.

**Set:**

Does not allow duplicates and does not maintain order.

**HashSet:**

Similar to a set, stores data in hash values.

**Tree Set:**

Does not allow duplicates but maintains ascending order. It can contain only one type of object due to sorting.

**List:**

Allows duplicates and maintains order.

**Array List:**

Stores objects in a continuous sequence.

**LinkedList:**

Divides memory into nodes, with each node containing three segments: previous, data, and next. Allows forward and backward traversal.

**Stack:**

Follows Last In, First Out (LIFO) principle.

**Queue:**

Follows First In, First Out (FIFO) principle.

**Map/Dictionary:**

Follows key-value pair structure.

1.HashMap

2.TreeMap

**Vector:**

Similar to a list but is asynchronous.

**Exception Handling**

Exception handling is used to manage exceptions that occur during the execution of a program.

Try: Block containing code that may throw an exception. Must be followed by catch or finally. Can have multiple catch blocks.

Catch: Handles the exceptions thrown by the try block.

Finally: Code that will execute whether an exception is handled or not.

Throw: Used to explicitly throw an exception.

Throws: Indicates that a method can throw one or more exceptions.

There are two types of exceptions:

1.Checked exceptions (compile-time)

1. IOException
2. FileNotFoundException
3. SQLException

2.Unchecked exceptions (runtime)

1. ArithmeticException
2. NullPointerException
3. ArrayIndexOutOfBoundsException