Ans 1: A Map is a collection in Java that stores data as key-value pairs, where each key is unique.

Ans 2: The commonly used implementations of Map in Java are HashMap, TreeMap, LinkedHashMap, and

ConcurrentHashMap.

Ans 3: HashMap is an unordered collection that uses hashing to store the key-value pairs, while TreeMap is a

sorted collection that stores the key-value pairs in a sorted order based on the natural order of the keys or a

custom Comparator.

Ans 4: we can check if a key exists in a Map in Java using the containsKey() method or the get() method. The

containsKey() method returns a boolean value indicating whether the Map contains the specified key, while the

get() method returns the value associated with the specified key, or null if the key is not present in the Map.

Ans 5: Generics in Java are used to provide type safety and reduce code redundancy by allowing the use of

generic types. It allows classes, methods, and interfaces to be written generically, without specifying the type of

data being used.

Ans 6: The benefits of using Generics in Java are:

Type safety

Code reusability

Improved readability

Reduced code redundancy

Improved performance

Ans 7: A Generic Class in Java is a class that can work with different types of data. It is defined using a type

parameter enclosed in angle brackets, which represents the type of data being used.

Ans 8: A Type Parameter in Java Generics is a placeholder for the type of data that is used by a generic class or

method. It is defined using a single uppercase letter enclosed in angle brackets, such as <T> or <E>.

Ans 9: A Generic Method in Java is a method that can work with different types of data. It is defined using a type

parameter enclosed in angle brackets, which represents the type of data being used.

Ans 10: ArrayList is a non-generic class, while ArrayList<T> is a generic class. ArrayList<T> provides type safety, as it

can only store elements of the specified type, whereas ArrayList can store any type of element.