**Java Collections and its benefits**

Collections aka containers are used to group objects into single entity. Once the objects are stored in a collection you can then perform operations on single object or multiple objects (bulk operations). You can insert, update, delete, and retrieve objects into entity. Also we can query the entity for size and other things.

Java Collections framework is widely used and it is definitely asked in interview. There can be numerous interview questions on Java Collections framework.

Let us first understand what java collections are.

Java collections are made up of following.

* **Interfaces** 
  + Interfaces are abstract data types which represents the collections. Interfaces allow the collection classes to be manipulated independently given that the implementing class adheres to the contract of the interface.
* **Implementations**
  + Implementations are class that implements the collection interfaces. There are abstract as well as concrete implementations of collection interfaces. This implementations are **ready to use, highly efficient, well tested and reusable** data structures.
* **Algorithms**
  + On the core of the implementation of the interfaces there reside the powerful algorithm implementations done on object. This includes searching sorting, shuffling, frequency and others. These algorithms are finely tuned and suit the needs of many applications as per the data size.

MOST IMPORTANT PART OF JAVA COLLECTION FRAMEWORK IS TO SELECT WHICH CLASSES TO USE THAT BETTER SUITS THE NEEDS OF APPLICATIONS.

Benefits of Java Collections

* **Tuned algorithms and quality**
  + There are several algorithms implemented in collections framework. We don’t know them but we use them daily. For example Arrays.sort(int[]).
  + These algorithms are finely tuned and suit the better needs of applications.
  + Because of these implementations developers are freed to design their own data structures and test them for errors or memory leaks.
* **Reduced Programming effort**
  + As the algorithms and data structures are implemented in collections developer can focus on other important parts of application.
* **Reduced learning effort**
  + It is very easy to learn collections because of its API names and documentation.
* **Reusability**
  + Collections can be reused. Its interfaces and implementations are flexible enough to be reused given that it adheres contract of interface.