HashMap :->

1) The underlined data structure for HashMap is HashTable.
2) insertion order is not preserved and it is based on hashcode of keys.
3) Duplicate keys are not allowed but values can be duplicated.
4) Heterogenious objects are allowed for both key and value.
5) Null is allowed for key (only once).
6) Null is allowed for values (any no. of times)
7) It implements Serializable, and Cloneable interface but not Random Access.
8) HashMap is the best choice if our frequent opn is search.

Constructors :-

1) HashMap m = new HashMap ();
   creates an empty hashMap object with default initial capacity 16 and Default feel ratio is 0.75.

2) HashMap m = new HashMap (int initialCapacity);
   creates an empty HashMap objects with specified initial capacity and default feel ratio is 0.75.

3) HashMap m = new HashMap (int initialCapacity, float fileRatio)

4) HashMap m = new HashMap (Map m);

Ex:-

```
import java.util.*;
public class Test {
    p s v m (String[] args)
{
    HashMap m = new HashMap();
    m.put("chiranjivi", 800); m.put("balaiah", 700);
    m.put("venkatesh", 200); m.put("nagarjuna", 500);
    S.O.p(m) => { K=V, K=V, ... }.
```

```
s.o.p(m.put("chiranjivi", 1000));
Set s = m.keySet();
s.o.p(s); ⇒ [k, k, k ----]
Collection c = m.values();
    s.o.p(c);
Set s1 = m.entrySet();
s.o.p(s1); →[K=V, K=V, ----]
Iterator itr = s1.iterator();
while(itr.hasNext())
{
Map.Entry m1 = (Map.Entry) itr.next();
    s.o.p(m1.getKey()+"--"+
          m1.getValue());
if(m1.getKey().equals("nagarjuna"))
    {
      m1.setValue(10000);
    }
    s.o.p(m);
}
}
```

⇒ Diff b/w HashMap & HashTable :→

HashMap	HashTable
① methods are non synchronized	① Methods are Synchronized
② Not Thread safe	② Thread safe
③ performance is high	③ performance is low.
④ Null is allowed as key and value.	④ Null is not allowed as key and value CE :- NPE
⑤ Not Legacy (1.2 V)	⑤ Legacy (1.0V)

⇒ How to get Synchronized version of HashMap Object ?.

Soln :- By default, HashMap is non synchronized but we can get Synchronized version of HashMap by using SynchronizedMap() method of Collections class.