Boolean result = hashmap.containsKey(1);

Code: LP_code1.java

Output

```
Value of Map is: {1=Piyush, 2=Athar, 3=Ajay, 4=Anil}
Value of Map is: {1=Rajesh, 2=Athar, 3=Ajay, 4=Anil}
Value of key = 3 is: Ajay
Value of Map is: {1=Rajesh, 3=Ajay, 4=Anil}
Key: 1 Value: Rajesh
Key: 3 Value: Ajay
Key: 4 Value: Anil
```

Collision in Hashmap

What is the Hash function?

A hash function is basically used to map a data of large size into a data of fixed size. The values returned by a hash function are called hash values.
The hash values are used to store keys in the map.

For eg a simple hash function could be

```
Integer hashFunction(Integer key) {
        return key%10;
}
```

Now if you pass any sort of data to this hash function it will return modulus 10. So all the hash values will be in order of 0-9.

What is collision

When a hash function starts returning same hash value for more than one key, then it results in collision. for eg for the above hash function, if we pass 10, 20 to the hash function it will return as 0 in both cases. Now we won't be able to store value for different keys as both are being mapped to same hash key.So it's very necessary to have a strong hash function.

How to handle collision

- **Separate Chaining.**
  a. Separate Chaining with linked lists.
- **Open addressing**
  a. Linear Probing
  b. Quadratic probing
  c. Double hashing.

Separate Chaining

Separate chaining is used when for the same hash value we start storing the keys in a chain. The chain can be of any type. Generally Linked list is preferred as the type of chain.

Now for eg. for they following keys 31,33,12, 13 ,10,20,30 the hashing would look like

{
0 -> [10]->[20]->[30]
1 -> [31]
2 -> [12]
3 -> [33,]->[13]

}

Now the time complexity would be to first find the hash key and then linearly search the list for that particular hash key.

Open Addressing

- **Linear probing**

In linear probing,

- When collision occurs, we linearly probe for the next bucket.
- We keep probing until an empty bucket is found.

Advantage-

- It is easy to compute.

Disadvantage-

- The main problem with linear probing is clustering.
- Many consecutive elements form groups.
- Then, it takes time to search an element or to find an empty bucket.

- **Quadratic Probing-**

In quadratic probing,

- When collision occurs, we probe for i2'th bucket in ith iteration.
- We keep probing until an empty bucket is found.

- **Double Hashing.**

As the name suggests, to store a hash, we use a second hash function which acts as an offset and updates the current hash function.

The new hash function would somewhat look like

h(i,k) = (h1(k) + i* h2(k))%SOME_CONSTANT

So now we have hash function h1 and hash function h2 which acts as an offset and updates the key for the total hash.