1. Write a Java program to associate the specified value with the specified key in a HashMap. [Hint:Create HashMap and store few elements on it]

package colleection;

import java.util.HashMap;

public class HashmapExampleKeyValue

{

public static void main(String[] args)

{

// Create a HashMap

HashMap<String, Integer> hashMap = new HashMap<>();

// Add elements to the HashMap

hashMap.put("One", 1);

hashMap.put("Two", 2);

hashMap.put("Three", 3);

// Display the HashMap

System.out.println("HashMap: " + hashMap);

}

}

**Output**

HashMap: {One=1, Two=2, Three=3}

2. Write a program in Java to create a Map Interface where we can store the cricketer name in it along with his scores and search for the batsman name and display his score. [Hint:use containsKey() method to search batsman name]

package colleection;

import java.util.HashMap;

import java.util.Map;

public class CricketScoreKey

{

public static void main(String[] args)

{

// creating hashmap to store name and score of cricketer

Map<String, Integer> cricketerScores = new HashMap<>();

// adding values

cricketerScores.put("Virat Kohli", 72);

cricketerScores.put("Rohit Sharma", 56);

cricketerScores.put("Steve Smith", 45);

cricketerScores.put("Kane Williamson", 68);

// searching Batsman

String batsmanName = "Virat Kohli";

if (cricketerScores.containsKey(batsmanName))

{

System.out.println(batsmanName + "'s score: " + cricketerScores.get(batsmanName));

}

else

{

System.out.println("Batsman not found in the records.");

}

}

}

**Output**

Virat Kohli's score: 72

3.Write a Java program that demonstrates the functionality of this dictionary application using a TreeMap. Your program should include the following features: i)A TreeMap named dictionary to store word-definition pairs. ii)A way to input word-definition pairs and add them to the dictionary. iii)A way to retrieve and display the definition of a specific word. iv)An iteration through the dictionary to display all word-definition pairs in alphabetical order based on words

package colleection;

import java.util.\*;

public class DictionaryApplication

{

public static void main(String[] args)

{

TreeMap<String, String> dictionary = new TreeMap<>();

Scanner scanner = new Scanner(System.in);

// Adding word-definition pairs to the dictionary

dictionary.put("apple", "a round fruit with red or green skin and a whitish interior");

dictionary.put("book", "a written or printed work consisting of pages glued or sewn together along one side");

// Retrieving and displaying the definition of a specific word

System.out.print("Enter a word to get its definition: ");

String word = scanner.nextLine();

if (dictionary.containsKey(word)) {

System.out.println("Definition: " + dictionary.get(word));

} else {

System.out.println("Word not found in the dictionary.");

}

// Displaying all word-definition pairs in alphabetical order

System.out.println("\nAll word-definition pairs in the dictionary:");

for (String key : dictionary.keySet()) {

System.out.println(key + ": " + dictionary.get(key));

}

scanner.close();

}

}

**Output**

Enter a word to get its definition: apple

Definition: a round fruit with red or green skin and a whitish interior

All word-definition pairs in the dictionary:

apple: a round fruit with red or green skin and a whitish interior

book: a written or printed work consisting of pages glued or sewn together along one side