Q. 52 Write a program in Java to create a class to calculate Area of circle with one data

member to store the radius and another to store area value.

Create method members

1. init - to input radius from user
2. calc - to calculate area
3. display - to display area.

```
import java.util.Scanner;
class Circle {
   private double radius;
   private double area;
   // Method to input radius from user
   public void init() {
        Scanner scanner = new Scanner(System.in);
        System.out.println(" Vipashyana Wagh, 0873CS231135");
        System.out.print("Enter the radius of the circle: ");
        radius = scanner.nextDouble();
        scanner.close();
   }
   // Method to calculate area
   public void calc() {
        area = 3.14159 * radius * radius;
   }
   // Method to display area
   public void display() {
        System.out.println("The radius of the circle is: " + radius);
        System.out.println("The area of the circle is: " + area);
   }
   public static void main(String[] args) {
        Circle circle = new Circle();
        circle.init();
        circle.calc();
```

```
        circle.display();

    }

}
```

Q 53 Find the greater value between a,b and c .

```
public class Main {
    public static void main(String[] args) {
        //TIP Press <shortcut actionId="ShowIntentionActions"/> with your
caret at the highlighted text
        // to see how IntelliJ IDEA suggests fixing it.
        int a = 50;
        int b = 100;
        int c = 150;
        if(a >= b && a>=c){
            System.out.print("a is greater than b and c");
        }else if(b>a && b>c){
            System.out.println("b is greater than a and c");
        }else if (c>a && c>b){
            System.out.println("c is greater than a and b ");
        }
    }
}
Output :-
/Library/Java/JavaVirtualMachines/jdk-24.jdk/Contents/Home/bin/java
-javaagent:/Applications/IntelliJ IDEA CE.app/Contents/lib/idea_rt.jar=49416
-Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8
-classpath /Users/vipashyanawagh/IdeaProjects/Java Workshop/Java
Workshop/out/production/Java Workshop Main

c is greater than a and b
```

```
Q .54  write a java program  to insert a specific element at a given position
in a linked list.

class Node {
    int data;
    Node next;

    public Node(int data) {
        this.data = data;
        this.next = null;
    }
}

class LinkedList {
    Node head;

    public void insertAtPosition(int data, int position) {
```

```
        Node newNode = new Node(data);
        if (position == 0) {
            newNode.next = head;
            head = newNode;
            return;
        }

        Node temp = head;
        for (int i = 0; i < position - 1 && temp != null; i++) {
            temp = temp.next;
        }

        if (temp == null) {
            System.out.println("Position exceeds the length of the list");
            return;
        }

        newNode.next = temp.next;
        temp.next = newNode;
    }

    public void printList() {
        Node temp = head;
        while (temp != null) {
            System.out.print(temp.data + " ");
            temp = temp.next;
        }
        System.out.println();
    }
}

public class Main {
    public static void main(String[] args) {
        LinkedList list = new LinkedList();
        list.insertAtPosition(10, 0);
        list.insertAtPosition(20, 1);
        list.insertAtPosition(30, 2);
        list.insertAtPosition(40, 3);
        list.insertAtPosition(50, 4);

        System.out.println("Linked List:");
        list.printList();

        list.insertAtPosition(25, 2);

        System.out.println("Linked List after insertion:");
        list.printList();
    }
}
```

```
/Library/Java/JavaVirtualMachines/jdk-24.jdk/Contents/Home/bin/java
-javaagent:/Applications/IntelliJ IDEA CE.app/Contents/lib/idea_rt.jar=49638
-Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8
```

-classpath /Users/vipashyanawagh/IdeaProjects/Java Workshop/Java Workshop/out/production/Java Workshop Main
Linked List:
10 20 30 40 50
Linked List after insertion:
10 20 25 30 40 50

```
55. Write a Java program to clone an array list to another array list.
 import java.util.ArrayList;

public class Main {
   public static void main(String[] args) {
       System.out.println("Name : Vipashyana Wagh, Roll no. 0873CS231135");
       // Create an ArrayList
       ArrayList<String> originalList = new ArrayList<>();
       originalList.add("Apple");
       originalList.add("Banana");
       originalList.add("Cherry");

       // Print the original ArrayList
       System.out.println("Original List: " + originalList);

       // Clone the ArrayList
       ArrayList<String> clonedList = new ArrayList<>(originalList);

       // Print the cloned ArrayList
       System.out.println("Cloned List: " + clonedList);

       // Modify the cloned ArrayList
       clonedList.add("Date");

       // Print the modified cloned ArrayList
       System.out.println("Modified Cloned List: " + clonedList);

       // Print the original ArrayList to verify that it's unchanged
       System.out.println("Original List after modification: " +
originalList);
   }
}
```

Output : - /Library/Java/JavaVirtualMachines/jdk-24.jdk/Contents/Home/bin/java -javaagent:/Applications/IntelliJ IDEA CE.app/Contents/lib/idea_rt.jar=49524 -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath /Users/vipashyanawagh/IdeaProjects/Java Workshop/Java Workshop/out/production/Java Workshop Main
Name : Vipashyana Wagh, Roll no. 0873CS231135
Original List: [Apple, Banana, Cherry]
Cloned List: [Apple, Banana, Cherry]
Modified Cloned List: [Apple, Banana, Cherry, Date]
Original List after modification: [Apple, Banana, Cherry]

```
56. Write a Java program for trimming the capacity of an array list.
import java.util.ArrayList;

public class Main {
   public static void main(String[] args) {
       System.out.println("Name : Vipashyana Wagh , Roll no . 0873CS231135");
       // Create an ArrayList
       ArrayList<String> list = new ArrayList<>(10);
       list.add("Apple");
       list.add("Banana");
       list.add("Cherry");

       // Print the original ArrayList
       System.out.println("Original List: " + list);

       // Print the capacity of the ArrayList
       System.out.println("Capacity before trimming: " + getCapacity(list));

       // Trim the capacity of the ArrayList
       list.trimToSize();

       // Print the capacity of the ArrayList after trimming
       System.out.println("Capacity after trimming: " + getCapacity(list));
   }

   // Method to get the capacity of an ArrayList
   public static int getCapacity(ArrayList<?> list) {
       try {
           java.lang.reflect.Field field =
ArrayList.class.getDeclaredField("elementData");
           field.setAccessible(true);
           Object[] array = (Object[]) field.get(list);
           return array.length;
       } catch (Exception e) {
           return -1; // Unable to get capacity
       }
   }
}
```

Output :- /Library/Java/JavaVirtualMachines/jdk-24.jdk/Contents/Home/bin/java -javaagent:/Applications/IntelliJ IDEA CE.app/Contents/lib/idea_rt.jar=49532 -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath /Users/vipashyanawagh/IdeaProjects/Java Workshop/Java Workshop/out/production/Java Workshop Main
Name : Vipashyana Wagh , Roll no . 0873CS231135
Original List: [Apple, Banana, Cherry]
Capacity before trimming: -1
Capacity after trimming: -1

```
57. Write a Java program for trimming the capacity of an array list.
import java.util.ArrayList;
```

```java
public class Main {
    public static void main(String[] args) {
        // Create an ArrayList with initial capacity 10
        ArrayList<String> list = new ArrayList<>(10);

        // Add elements to the list
        list.add("Apple");
        list.add("Banana");
        list.add("Cherry");

        // Print the original list
        System.out.println("Original List: " + list);

        // Print the size of the list
        System.out.println("Size of the list: " + list.size());

        // Trim the capacity of the list to its current size
        list.trimToSize();

        // Print a message indicating that the capacity has been trimmed
        System.out.println("Capacity trimmed to size: " + list.size());
    }
}
```

output : /Library/Java/JavaVirtualMachines/jdk-24.jdk/Contents/Home/bin/java -javaagent:/Applications/IntelliJ IDEA CE.app/Contents/lib/idea_rt.jar=49534 -Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8 -classpath /Users/vipashyanawagh/IdeaProjects/Java Workshop/Java Workshop/out/production/Java Workshop Main
Original List: [Apple, Banana, Cherry]
Size of the list: 3
Capacity trimmed to size: 3

Process finished with exit code 0

Q55. Write a Java program to join two array lists.

.import java.util.ArrayList;

public class Main {

    public static void main(String[] args) {

        // Create two ArrayLists

        ArrayList<String> list1 = new ArrayList<>();

        list1.add("Apple");

```
        list1.add("Banana");
        list1.add("Cherry");

        ArrayList<String> list2 = new ArrayList<>();
        list2.add("Date");
        list2.add("Elderberry");
        list2.add("Fig");

        // Print the original ArrayLists
        System.out.println("List 1: " + list1);
        System.out.println("List 2: " + list2);

        // Join the two ArrayLists
        list1.addAll(list2);

        // Print the joined ArrayList
        System.out.println("Joined List: " + list1);
    }
}
```

Output:-

```
/Library/Java/JavaVirtualMachines/jdk-24.jdk/Contents/Home/bin/java
-javaagent:/Applications/IntelliJ IDEA CE.app/Contents/lib/idea_rt.jar=49506
-Dfile.encoding=UTF-8 -Dsun.stdout.encoding=UTF-8 -Dsun.stderr.encoding=UTF-8
-classpath /Users/vipashyanawagh/IdeaProjects/Java Workshop/Java
Workshop/out/production/Java Workshop Main
List 1: [Apple, Banana, Cherry]
List 2: [Date, Elderberry, Fig]
Joined List: [Apple, Banana, Cherry, Date, Elderberry, Fig]
```

Q 58. Write a Java program to insert elements at the first and last positions of a linked list.

```
class Node {
    int data;
    Node next;

    public Node(int data) {
        this.data = data;
        this.next = null;
    }
}

// LinkedList class with methods to insert at first and last positions
class LinkedList {
    Node head;

    // Method to insert a new node at the beginning of the linked list
    public void insertAtFirst(int data) {
        Node newNode = new Node(data);
        if (head == null) {
            head = newNode;
        } else {
            newNode.next = head;
            head = newNode;
        }
    }

    // Method to insert a new node at the end of the linked list
```

```
    public void insertAtLast(int data) {
        Node newNode = new Node(data);
        if (head == null) {
            head = newNode;
        } else {
            Node temp = head;
            while (temp.next != null) {
                temp = temp.next;
            }
            temp.next = newNode;
        }
    }

    // Method to print the linked list
    public void printList() {
        Node temp = head;
        while (temp != null) {
            System.out.print(temp.data + " ");
            temp = temp.next;
        }
        System.out.println();
    }
}

public class Main {
    public static void main(String[] args) {
        LinkedList list = new LinkedList();
System.out.println("Name : Vipashyana Wagh ,0873CS231135");
```

```
        // Insert elements at the last position
        list.insertAtLast(10);
        list.insertAtLast(20);
        list.insertAtLast(30);

        System.out.println("Linked List after inserting at last:");
        list.printList();

        // Insert elements at the first position
        list.insertAtFirst(5);

        System.out.println("Linked List after inserting at first:");
        list.printList();

        // Insert another element at the last position
        list.insertAtLast(40);

        System.out.println("Linked List after inserting at last again:");
        list.printList();
    }
}
```

Output:

Name :- Vipashyana Wagh

Roll no. 0873CS231135

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java"

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java\" ; if ($?) { javac Main.java } ; if ($?) { java Main }

Linked List after inserting at last:

10 20 30

Linked List after inserting at first:

5 10 20 30

Linked List after inserting at last again:

5 10 20 30 40

PS C:\Users\HP\Desktop\java>

Q 59 : Write a Java program to add all elements from one TreeSet to another TreeSet

```
import java.util.TreeSet;

public class Main {
    public static void main(String[] args) {
        // Create the first TreeSet
System.out.println(“Name : Vipashyana Wagh , Roll no.0873CS231135”);
        TreeSet<String> treeSet1 = new TreeSet<>();
        treeSet1.add("Apple");
        treeSet1.add("Banana");
        treeSet1.add("Cherry");

        // Create the second TreeSet
        TreeSet<String> treeSet2 = new TreeSet<>();
        treeSet2.add("Date");
        treeSet2.add("Elderberry");
        treeSet2.add("Fig");

        // Print the original TreeSets
        System.out.println("TreeSet 1: " + treeSet1);
```

```
        System.out.println("TreeSet 2: " + treeSet2);

        // Add all elements from treeSet1 to treeSet2
        treeSet2.addAll(treeSet1);

        // Print the updated TreeSet2
        System.out.println("TreeSet 2 after adding all elements from TreeSet 1: " + treeSet2);
    }
}
```

Output:

Name :- Vipashyana Wagh

Roll no. 0873CS231135

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java"

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java\" ; if ($?) { javac Main.java } ; if ($?) { java Main }

TreeSet 1: [Apple, Banana, Cherry]

TreeSet 2: [Date, Elderberry, Fig]

TreeSet 2 after adding all elements from TreeSet 1: [Apple, Banana, Cherry, Date, Elderberry, Fig]

PS C:\Users\HP\Desktop\java>

Q 60. Write a Java program to display the elements of a TreeSet in reverse order.

```
import java.util.TreeSet;

public class Main {
    public static void main(String[] args) {
        // Create a TreeSet
System.out.println(“Name : Vipashyana Wagh , Roll no. 0873CS231135”);
```

```
        TreeSet<String> treeSet = new TreeSet<>();
        treeSet.add("Apple");
        treeSet.add("Banana");
        treeSet.add("Cherry");
        treeSet.add("Date");
        treeSet.add("Elderberry");

        // Print the original TreeSet
        System.out.println("Original TreeSet: " + treeSet);

        // Display the elements of the TreeSet in reverse order
        System.out.println("TreeSet in reverse order:");
        for (String element : treeSet.descendingSet()) {
            System.out.println(element);
        }
    }
}
```

Output:

Name :- Vipashyana Wagh

Roll no . 0873CS231135

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java"

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java\" ; if ($?) { javac Main.java } ; if ($?) { java Main }

Original TreeSet: [Apple, Banana, Cherry, Date, Elderberry]

TreeSet in reverse order:

Elderberry

Date

Cherry

Banana

Apple

PS C:\Users\HP\Desktop\java>

Q 61 . Write a Java program to retrieve the first and last elements from a TreeSet.

```
import java.util.TreeSet;

public class Main {
    public static void main(String[] args) {
        // Create a TreeSet
System.out.println(“Name : Vipashyana Wagh , Roll no. 0873CS231135”);

        TreeSet<String> treeSet = new TreeSet<>();
        treeSet.add("Apple");
        treeSet.add("Banana");
        treeSet.add("Cherry");
        treeSet.add("Date");
        treeSet.add("Elderberry");

        // Print the original TreeSet
        System.out.println("Original TreeSet: " + treeSet);

        // Retrieve the first element
        String firstElement = treeSet.first();
        System.out.println("First element: " + firstElement);

        // Retrieve the last element
        String lastElement = treeSet.last();
```

```
        System.out.println("Last element: " + lastElement);
    }
}
```

Output:

Name :- Vipashyana Wagh

Roll No. 0873CS231135

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java"

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java\" ; if ($?) { javac Main.java } ; if ($?) { java Main }

Original TreeSet: [Apple, Banana, Cherry, Date, Elderberry]

First element: Apple

Last element: Elderberry

PS C:\Users\HP\Desktop\java>

Q 62. Write a Java program to clone a TreeSet into another TreeSet.

```
import java.util.TreeSet;

public class Main {
    public static void main(String[] args) {
        // Create the first TreeSet
System.out.println(“Name : Vipashyana Wagh , Roll no. 0873CS231135”);

        TreeSet<String> treeSet1 = new TreeSet<>();
        treeSet1.add("Apple");
        treeSet1.add("Banana");
        treeSet1.add("Cherry");
        treeSet1.add("Date");
        treeSet1.add("Elderberry");
```

```
        // Print the original TreeSet
        System.out.println("Original TreeSet: " + treeSet1);

        // Clone the TreeSet
        TreeSet<String> treeSet2 = (TreeSet<String>) treeSet1.clone();

        // Print the cloned TreeSet
        System.out.println("Cloned TreeSet: " + treeSet2);
    }
}
```

Output:

Name :- Vipashyana Wagh

Roll No. 0873CS231135

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java"

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java\" ; if ($?) { javac Main.java } ; if ($?) { java Main }

Note: Main.java uses unchecked or unsafe operations.

Note: Recompile with -Xlint:unchecked for details.

Original TreeSet: [Apple, Banana, Cherry, Date, Elderberry]

Cloned TreeSet: [Apple, Banana, Cherry, Date, Elderberry]

PS C:\Users\HP\Desktop\java>

Q 63. Write a Java program to count the number of elements in a TreeSet

```
import java.util.TreeSet;

public class Main {
```

```
    public static void main(String[] args) {
        // Create a TreeSet
System.out.println(“Name : Vipashyana Wagh , Roll no. 0873CS231135”);

        TreeSet<String> treeSet = new TreeSet<>();
        treeSet.add("Apple");
        treeSet.add("Banana");
        treeSet.add("Cherry");
        treeSet.add("Date");
        treeSet.add("Elderberry");

        // Print the original TreeSet
        System.out.println("Original TreeSet: " + treeSet);

        // Count the number of elements
        int count = treeSet.size();

        // Print the count
        System.out.println("Number of elements: " + count);
    }
}
```

Output:

Name: Vipashyana Wagh

Roll no. 0873CS231135

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java"

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java\" ; if ($?) { javac Main.java } ; if ($?) { java Main }

Original TreeSet: [Apple, Banana, Cherry, Date, Elderberry]

```
Number of elements: 5
PS C:\Users\HP\Desktop\java>
```

Q 64 . Write a Java program to compare two TreeSets.

```
import java.util.TreeSet;

public class Main {
    public static void main(String[] args) {
        // Create the first TreeSet
System.out.println(“Name : Vipashyana Wagh , Roll no. 0873CS231135”);
  TreeSet<String> treeSet1 = new TreeSet<>();
        treeSet1.add("Apple");
        treeSet1.add("Banana");
        treeSet1.add("Cherry");

        // Create the second TreeSet
        TreeSet<String> treeSet2 = new TreeSet<>();
        treeSet2.add("Apple");
        treeSet2.add("Banana");
        treeSet2.add("Cherry");

        // Create the third TreeSet
        TreeSet<String> treeSet3 = new TreeSet<>();
        treeSet3.add("Apple");
        treeSet3.add("Banana");
        treeSet3.add("Date");

        // Compare the TreeSets
```

```
        System.out.println("TreeSet 1: " + treeSet1);
        System.out.println("TreeSet 2: " + treeSet2);
        System.out.println("TreeSet 3: " + treeSet3);

        System.out.println("TreeSet 1 equals TreeSet 2: " + treeSet1.equals(treeSet2));
        System.out.println("TreeSet 1 equals TreeSet 3: " + treeSet1.equals(treeSet3));
    }
}
```

Output:

Name :- Vipashyana Wagh

Roll no. 0873CS231135

```
PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java"
PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java\" ; if ($?) { javac Main.java } ; if ($?) { java Main }
TreeSet 1: [Apple, Banana, Cherry]
TreeSet 2: [Apple, Banana, Cherry]
TreeSet 3: [Apple, Banana, Date]
TreeSet 1 equals TreeSet 2: true
TreeSet 1 equals TreeSet 3: false
PS C:\Users\HP\Desktop\java>
```

Q 65. Write a Java program to clone one HashSet into another

```
import java.util.HashSet;
```

```
public class Main {
    public static void main(String[] args) {
        // Create the first HashSet
System.out.println(“Name : Vipashyana Wagh, 0873CS231135”);
        HashSet<String> hashSet1 = new HashSet<>();
        hashSet1.add("Apple");
        hashSet1.add("Banana");
        hashSet1.add("Cherry");

        // Print the original HashSet
        System.out.println("Original HashSet: " + hashSet1);

        // Clone the HashSet
        HashSet<String> hashSet2 = (HashSet<String>) hashSet1.clone();

        // Print the cloned HashSet
        System.out.println("Cloned HashSet: " + hashSet2);
    }
}
```

Output:

Name :- Vipashyana Wagh

Roll no 0873CS231135

S C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java"

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java\" ; if ($?) { javac Main.java } ; if ($?) { java Main }

Note: Main.java uses unchecked or unsafe operations.

Note: Recompile with -Xlint:unchecked for details.

Original HashSet: [Apple, Cherry, Banana]

```
Cloned HashSet: [Cherry, Apple, Banana]
PS C:\Users\HP\Desktop\java>
```

Q 66 . Write a Java program to convert a HashSet into an array.

```
import java.util.HashSet;

public class Main {
    public static void main(String[] args) {
        // Create a HashSet
System.out.println(“Name : Vipashyana Wagh, Roll. No 0873CS231135”);
        HashSet<String> hashSet = new HashSet<>();
        hashSet.add("Apple");
        hashSet.add("Banana");
        hashSet.add("Cherry");

        // Print the original HashSet
        System.out.println("Original HashSet: " + hashSet);

        // Convert the HashSet into an array
        String[] array = hashSet.toArray(new String[0]);

        // Print the array
        System.out.println("Array:");
        for (String element : array) {
            System.out.println(element);
        }
    }
}
```

Output:

Name Vipashyana Wagh

Roll no . 0873CS231135

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java"

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java\" ; if ($?) { javac Main.java } ; if ($?) { java Main }

Original HashSet: [Apple, Cherry, Banana]

Array:

Apple

Cherry

Banana

PS C:\Users\HP\Desktop\java>

Q 67. Write a Java program to convert a HashSet into a TreeS

```
import java.util.HashSet;
import java.util.TreeSet;


public class Main {
    public static void main(String[] args) {
        // Create a HashSet
System.out.println(“Name : Vipashyana Wagh, Roll no. 0873CS231135”);
        HashSet<String> hashSet = new HashSet<>();
        hashSet.add("Banana");
        hashSet.add("Apple");
        hashSet.add("Cherry");


        // Print the original HashSet
        System.out.println("Original HashSet: " + hashSet);
```

```
        // Convert the HashSet into a TreeSet
        TreeSet<String> treeSet = new TreeSet<>(hashSet);

        // Print the TreeSet
        System.out.println("TreeSet: " + treeSet);
    }
}
```

Output:

Name Vipashyana Wagh

Roll no . 0873CS231135

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java"

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java\" ; if ($?) { javac Main.java } ; if ($?) { java Main }

Original HashSet: [Apple, Cherry, Banana]

TreeSet: [Apple, Banana, Cherry]

PS C:\Users\HP\Desktop\java>

Q 68. Write a Java program to find numbers less than 7 in a TreeSet.

```
import java.util.TreeSet;

public class Main {
    public static void main(String[] args) {
        // Create a TreeSet
System.out.println(“Name : Vipashyana Wagh, Roll no. 0873CS231135”);
 TreeSet<Integer> treeSet = new TreeSet<>();
        treeSet.add(3);
        treeSet.add(5);
        treeSet.add(7);
```

```
        treeSet.add(9);
        treeSet.add(1);

        // Print the original TreeSet
        System.out.println("Original TreeSet: " + treeSet);

        // Find numbers less than 7
        System.out.println("Numbers less than 7:");
        for (Integer num : treeSet.headSet(7)) {
            System.out.println(num);
        }
    }
}
```

Output:

Name Vipashyana Wagh

Roll no . 0873CS231135

```
 PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java"
PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java\" ; if ($?) { javac Main.java } ; if ($?) { java Main }
Original TreeSet: [1, 3, 5, 7, 9]
Numbers less than 7:
1
3
5
PS C:\Users\HP\Desktop\java>
```

Q 69. Write a Java program to compare two HashSets.

```
import java.util.HashSet;
```

```
public class Main {
    public static void main(String[] args) {
        // Create the first HashSet
System.out.println(“Name : Vipashyana Wagh, Roll no. 0873CS231135”);
 HashSet<String> hashSet1 = new HashSet<>();
        hashSet1.add("Apple");
        hashSet1.add("Banana");
        hashSet1.add("Cherry");

        // Create the second HashSet
        HashSet<String> hashSet2 = new HashSet<>();
        hashSet2.add("Apple");
        hashSet2.add("Banana");
        hashSet2.add("Cherry");

        // Create the third HashSet
        HashSet<String> hashSet3 = new HashSet<>();
        hashSet3.add("Apple");
        hashSet3.add("Banana");
        hashSet3.add("Date");

        // Compare the HashSets
        System.out.println("HashSet 1: " + hashSet1);
        System.out.println("HashSet 2: " + hashSet2);
        System.out.println("HashSet 3: " + hashSet3);

        System.out.println("HashSet 1 equals HashSet 2: " + hashSet1.equals(hashSet2));
```

```
        System.out.println("HashSet 1 equals HashSet 3: " + hashSet1.equals(hashSet3));
    }
}
```

Output:

Name Vipashyana Wagh

Roll no. 0873CS231135

Are sets equal? True

Q 70. Write a Java program to retain common elements from two sets.

```
import java.util.HashSet;

public class Main {
    public static void main(String[] args) {
        // Create the first HashSet
System.out.println(“Name : Vipashyana Wagh, Roll no. 0873CS231135”);
 HashSet<String> hashSet1 = new HashSet<>();
        hashSet1.add("Apple");
        hashSet1.add("Banana");
        hashSet1.add("Cherry");
        hashSet1.add("Date");

        // Create the second HashSet
        HashSet<String> hashSet2 = new HashSet<>();
        hashSet2.add("Apple");
        hashSet2.add("Banana");
        hashSet2.add("Elderberry");
        hashSet2.add("Fig");

        // Print the original sets
```

```
        System.out.println("HashSet 1: " + hashSet1);
        System.out.println("HashSet 2: " + hashSet2);

        // Retain common elements
        hashSet1.retainAll(hashSet2);

        // Print the common elements
        System.out.println("Common elements: " + hashSet1);
    }
}
```

Output:

Name Vipashyana Wagh

Roll no . 0873CS231135

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java"

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java\" ; if ($?) { javac Main.java } ; if ($?) { java Main }

HashSet 1: [Apple, Cherry, Date, Banana]

HashSet 2: [Apple, Fig, Elderberry, Banana]

Common elements: [Apple, Banana]

PS C:\Users\HP\Desktop\java>

Q 71. Write a Java program to remove all elements from a HashSet

```
 import java.util.HashSet;
  public class Main {
     public static void main(String[] args) {
System.out.println(“Name : Vipashyana Wagh, Roll no. 0873CS231135”);
// Create a HashSet
```

```
        HashSet<String> hashSet = new HashSet<>();
         hashSet.add("Apple");
          hashSet.add("Banana");
          hashSet.add("Cherry");

      // Print the original HashSet
      System.out.println("Original HashSet: " + hashSet);

      // Remove all elements
      hashSet.clear();

      // Print the HashSet after removal
      System.out.println("HashSet after removal: " + hashSet);
   }
}
```

```
PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java"
PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java\" ; if ($?) { javac Main.java } ; if ($?) { java Main }
Name: Vipashyana Wagh
Enrollement:0873CS231135
Original HashSet: [Apple, Cherry, Banana]
HashSet after removal: []
PS C:\Users\HP\Desktop\java>
```

Q 72. Write a Java program to copy all mappings from one map to another.

```
import java.util.HashMap;
import java.util.Map;

public class Main {
```

```
    public static void main(String[] args) {
System.out.println(“Name : Vipashyana Wagh, Roll no. 0873CS231135”);
  // Create the first map
        Map<String, String> map1 = new HashMap<>();
        map1.put("Apple", "Fruit");
        map1.put("Carrot", "Vegetable");
        map1.put("Potato", "Vegetable");

        // Create the second map
        Map<String, String> map2 = new HashMap<>();

        // Print the original maps
        System.out.println("Map 1: " + map1);
        System.out.println("Map 2: " + map2);

        // Copy all mappings from map1 to map2
        map2.putAll(map1);

        // Print the maps after copying
        System.out.println("Map 1 after copying: " + map1);
        System.out.println("Map 2 after copying: " + map2);
    }
}
```

Output :-

Name Vipashyana Wagh

Roll no . 0873CS231135

C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java\" ; if ($?) { javac Main.java } ; if ($?) { java Main }

```
Map 1: {Potato=Vegetable, Apple=Fruit, Carrot=Vegetable}
Map 2: {}
Map 1 after copying: {Potato=Vegetable, Apple=Fruit, Carrot=Vegetable}
Map 2 after copying: {Potato=Vegetable, Apple=Fruit, Carrot=Vegetable}
PS C:\Users\HP\Desktop\java>
```

Q. 73.. Write a Java program to remove all key-value pairs from a map.

```
import java.util.HashMap;
import java.util.Map;

public class Main {
    public static void main(String[] args) {
System.out.println("Name : Vipashyana Wagh, Roll no. 0873CS231135");
// Create a map
        Map<String, String> map = new HashMap<>();
        map.put("Apple", "Fruit");
        map.put("Carrot", "Vegetable");
        map.put("Potato", "Vegetable");

        // Print the original map
        System.out.println("Original Map: " + map);

        // Remove all key-value pairs
        map.clear();

        // Print the map after removal
        System.out.println("Map after removal: " + map);
    }
```

```
}
```

Output :-

```
Name Vipashyana Wagh
Roll no. 0873CS231135
PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java"
PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java\" ; if ($?) { javac Main.java } ; if ($?) { java Main }
Original Map: {Potato=Vegetable, Apple=Fruit, Carrot=Vegetable}
Map after removal: {}
PS C:\Users\HP\Desktop\java>
```

Q 74. Write a Java program to check if a map is empty or contains key-value mappings.

```
import java.util.HashMap;
import java.util.Map;

public class Main {
    public static void main(String[] args) {
System.out.println(“Name : Vipashyana Wagh, Roll no. 0873CS231135”);

        // Create a map
        Map<String, String> map = new HashMap<>();
        map.put("Apple", "Fruit");
        map.put("Carrot", "Vegetable");
        map.put("Potato", "Vegetable");

        // Check if map is empty
        System.out.println("Is map empty? " + isMapEmpty(map));
```

```
        // Remove all key-value pairs
        map.clear();

        // Check if map is empty
        System.out.println("Is map empty after clearing? " + isMapEmpty(map));
    }

    public static boolean isMapEmpty(Map<?, ?> map) {
        return map.isEmpty();
    }
}
```

Output :-

Name Vipashyana Wagh

Roll no 0873CS231135

```
C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java"
PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java\" ; if ($?) { javac Main.java } ; if ($?) { java Main }
Is map empty? false
Is map empty after clearing? true
PS C:\Users\HP\Desktop\java>
```

Q 75. Write a Java program to create a shallow copy of a HashMap instance.

```
import java.util.HashMap;
import java.util.Map;

public class Main {
    public static void main(String[] args) {
System.out.println(“Name : Vipashyana Wagh, Roll no. 0873CS231135”);
```

```
        // Create a map
        Map<String, String> originalMap = new HashMap<>();
        originalMap.put("Apple", "Fruit");
        originalMap.put("Carrot", "Vegetable");
        originalMap.put("Potato", "Vegetable");

        // Create a shallow copy of the map
        Map<String, String> copiedMap = new HashMap<>(originalMap);

        // Print the original and copied maps
        System.out.println("Original Map: " + originalMap);
        System.out.println("Copied Map: " + copiedMap);

        // Modify the original map
        originalMap.put("Tomato", "Fruit");

        // Print the original and copied maps after modification
        System.out.println("Original Map after modification: " + originalMap);
        Sy
stem.out.println("Copied Map after modification: " + copiedMap);
    }
}
```

Output :-

Name Vipashyana Wagh

Roll no . 0873CS231135

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java"

```
PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java\" ; if ($?) { javac Main.java } ; if ($?) { java Main }
Original Map: {Potato=Vegetable, Apple=Fruit, Carrot=Vegetable}
Copied Map: {Potato=Vegetable, Apple=Fruit, Carrot=Vegetable}
Original Map after modification: {Potato=Vegetable, Apple=Fruit, Carrot=Vegetable, Tomato=Fruit}
Copied Map after modification: {Potato=Vegetable, Apple=Fruit, Carrot=Vegetable}
PS C:\Users\HP\Desktop\java>
```

Q 76 Write a Java program to test whether a specified key exists in the map.

```
import java.util.HashMap;
import java.util.Map;

public class Main {
    public static void main(String[] args) {
System.out.println(“Name : Vipashyana Wagh, Roll no. 0873CS231135”);

        // Create a map
        Map<String, String> map = new HashMap<>();
        map.put("Apple", "Fruit");
        map.put("Carrot", "Vegetable");
        map.put("Potato", "Vegetable");

        // Test if a key exists
        String key = "Apple";
        if (map.containsKey(key)) {
```

```
        System.out.println("The key '" + key + "' exists in the map with value: " +
map.get(key));
    } else {
        System.out.println("The key '" + key + "' does not exist in the map");
    }

    // Test if a key does not exist
    key = "Tomato";
    if (map.containsKey(key)) {
        System.out.println("The key '" + key + "' exists in the map with value: " +
map.get(key));
    } else {
        System.out.println("The key '" + key + "' does not exist in the map");
    }
  }
}
```

Output :-

Name Vipashyana Wagh

Roll no . 0873CS231135

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java"

PS C:\Users\HP\Desktop\java> cd "c:\Users\HP\Desktop\java\" ; if ($?) { javac Main.java } ; if ($?) { java Main }

The key 'Apple' exists in the map with value: Fruit

PS C:\Users\HP\Desktop\java>

Q 77 . Create a table Item_dtls (Electronics)
Try to insert at least 10 records in the above table
Try to insert at least 2 records with null value

```
CREATE TABLE Item_dtls (
Item_id INT PRIMARY KEY,
Item_name VARCHAR(50),
Category VARCHAR(30),
```

Price DECIMAL(10, 2)
);
INSERT INTO Item_dtls VALUES (1, ‘Smartphone’, ‘Mobile’, 25000);
INSERT INTO Item_dtls VALUES (2, ‘Laptop’, ‘Computers’, 60000);
INSERT INTO Item_dtls VALUES (3, ‘TV’, ‘Home Appliances’, 30000);
INSERT INTO Item_dtls VALUES (4, ‘Headphones’, ‘Accessories’, 2000);
INSERT INTO Item_dtls VALUES (5, ‘Camera’, ‘Photography’, 35000);
INSERT INTO Item_dtls VALUES (6, ‘Smartwatch’, ‘Wearable’, 10000);
INSERT INTO Item_dtls VALUES (7, ‘Microwave’, ‘Kitchen’, 8000);
INSERT INTO Item_dtls VALUES (8, ‘Printer’, ‘Office’, 5000);
INSERT INTO Item_dtls VALUES (9, NULL, ‘Gaming’, 40000);
INSERT INTO Item_dtls VALUES (10, ‘Speaker’, NULL, 3000);

Q 78. Create a table Sales_dtls
Try to insert at least 10 records in the above table
Try to insert at least 2 records with null valueCREATE TABLE Sales_dtls (
Sale_id INT PRIMARY KEY,
Item_id INT,
Quantity INT,
Sale_date DATE
);
INSERT INTO Sales_dtls VALUES (1, 1, 2, ‘2025-01-10’);
INSERT INTO Sales_dtls VALUES (2, 2, 1, ‘2025-01-11’);
INSERT INTO Sales_dtls VALUES (3, 3, 1, ‘2025-01-12’);
INSERT INTO Sales_dtls VALUES (4, 4, 5, ‘2025-01-13’);
INSERT INTO Sales_dtls VALUES (5, 5, 3, ‘2025-01-14’);
INSERT INTO Sales_dtls VALUES (6, 6, 2, ‘2025-01-15’);
INSERT INTO Sales_dtls VALUES (7, 7, 1, ‘2025-01-16’);
INSERT INTO Sales_dtls VALUES (8, 8, 4, ‘2025-01-17’);
INSERT INTO Sales_dtls VALUES (9, 9, NULL, ‘2025-01-18’);
INSERT INTO Sales_dtls VALUES (10, 10, 2, NULL);
85.create a table manufacturers
Try to insert at least 10 records in the above table
Try to insert at least 2 records with null value
Consider the below tables with estimated columns and then practise below questions.
CUST DTLS
CUST Act DTLS
ACT_TYPES_INFO
PROD_DTLSEMP
DEPT
CREATE TABLE manufacturers (

```
Mfg_id INT PRIMARY KEY,
Mfg_name VARCHAR(50),
Country VARCHAR(50)
);
INSERT INTO manufacturers VALUES (1, ‘Samsung’, ‘South Korea’);
INSERT INTO manufacturers VALUES (2, ‘Apple’, ‘USA’);
INSERT INTO manufacturers VALUES (3, ‘Sony’, ‘Japan’);
INSERT INTO manufacturers VALUES (4, ‘LG’, ‘South Korea’);
INSERT INTO manufacturers VALUES (5, ‘Dell’, ‘USA’);
INSERT INTO manufacturers VALUES (6, ‘HP’, ‘USA’);
INSERT INTO manufacturers VALUES (7, ‘Panasonic’, ‘Japan’);
INSERT INTO manufacturers VALUES (8, ‘Xiaomi’, ‘China’);
INSERT INTO manufacturers VALUES (9, NULL, ‘China’);
INSERT INTO manufacturers VALUES (10, ‘Realme’, NULL);
```