



(DIGITAL ASSIGNMENT - 5)

SYNCHRONIZATION, INTER THREADING, FILES, COLLECTION, JAVAFX AND JDBC

CSE1007(JAVA PROGRAMMING)LAB:L31-L32



APRIL 18, 2022
ANIRUDH VADERA
20BCE2940

ACTIVITY – 7(INTER THREADING)

QUESTION 1:

1. Demonstrate multithreading by creating two threads, one for printing the odd numbers and the other for printing even numbers with in a given range of your choice.

CODE:

```
package activity7;

class Array {
    static int[] a = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 };

    public static synchronized void print(String odd_even) {
        if (odd_even.equals("ODD")) {
            System.out.println("Printing Odd Numbers");
            System.out.print("[ ");
            for (int i = 0; i < a.length; i++) {
                if (a[i] % 2 != 0) {
                    System.out.print(a[i] + " ");
                }
            }
            System.out.println("]");
        } else {
            System.out.println("Printing Even Numbers");
            System.out.print("[ ");
            for (int i = 0; i < a.length; i++) {
                if (a[i] % 2 == 0) {
                    System.out.print(a[i] + " ");
                }
            }
            System.out.println("]");
        }
    }
}

public class activity7q1 {
    public static void main(String[] args) throws InterruptedException {
        System.out.println("ANIRUDH VADERA(20BCE2940)");
        Array a = new Array();
        Thread t1 = new Thread(() -> {
            System.out.println("Thread 1 - ODD");
            System.out.println(Thread.currentThread().getName());
            Array.print("ODD");
        });
        t1.start();
        Thread t2 = new Thread(() -> {
            System.out.println("Thread 2 - EVEN");
            System.out.println(Thread.currentThread().getName());
            Array.print("EVEN");
        });
        t2.start();
    }
}
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
});  
Thread t2 = new Thread(() -> {  
    System.out.println("Thread 2 - EVEN");  
    System.out.println(Thread.currentThread().getName());  
    Array.print("EVEN");  
});  
t1.start();  
t2.start();  
t1.join();  
t2.join();  
System.out.println("Thread 2 - MAIN");  
System.out.println(Thread.currentThread().getName());  
}  
}  
}
```

OUTPUT:

```
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes> c;; cd 'c:\Users\Anirudh\OneDrive\Desktop\java_codes';  
& 'C:\Liberica-JDK-17\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Anirudh\AppData\Roaming\Code\User\workspaceStorage\30032ff6b908f564bf959529c93335e9\redhat.java\jdt_ws\java_codes_72  
fe121c\bin' 'activity7.activity7q1'  
ANIRUDH VADERA(20BCE2940)  
Thread 1 - ODD  
Thread-0  
Thread 2 - EVEN  
Thread-1  
Printing Odd Numbers  
[ 1 3 5 7 9 ]  
Printing Even Numbers  
[ 2 4 6 8 10 ]  
Thread 2 - MAIN  
main  
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes> []
```

ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC

QUESTION 2:

2. Write a java program to create thread that stores a set of unique numbers from 100 to 200, whose sum of digits is a factor of the same number. If the input string is not in the range, raise a user defined exception. Another thread that stores the set of unique numbers from 1 to 100 whose sum of digits is prime number. Both the threads should execute in equal intervals of 10 numbers.

CODE:

```
package activity7;

import java.util.Scanner;

class invalidInputNotRange1 extends Exception {
    public invalidInputNotRange1(String message) {
        super(message);
    }
}

class invalidInputNotRange2 extends Exception {
    public invalidInputNotRange2(String message) {
        super(message);
    }
}

class Thread1 implements Runnable {
    int[] number;
    int range1_low;
    int range1_high;
    int sum = 0;
    int[] array_factor = new int[100];
    int factor_elements = 0;
    int[] unique = new int[100];
    int unique_elements = 0;
    int start;
    int end;

    Thread1(int[] number, int start, int end, int range1_low, int range1_high)
    {
        this.number = number;
        this.range1_high = range1_high;
        this.range1_low = range1_low;
        this.start = start;
        this.end = end;
    }

    @Override
    public void run() {
        for (int i = start; i <= end; i++) {
            sum = 0;
            for (int j = i; j > 0; j /= 10) {
                sum += j % 10;
            }
            if (sum == 1 || sum == 2 || sum == 3 || sum == 5 || sum == 7) {
                unique[unique_elements] = i;
                unique_elements++;
            }
            if (i % 10 == 0) {
                array_factor[factor_elements] = i;
                factor_elements++;
            }
        }
    }
}
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
}

void validate1(int input) throws invalidInputNotRange1 {
    if (input >= range1_low && input <= range1_high) {

    } else {
        throw new invalidInputNotRange1(
            "The Input (" + input + ") is not in the Specified Range
of " + "[" +
                range1_low + "," +
                range1_high + "]");
    }
}

public void generate() {
    for (int number = range1_low; number < (range1_high + 1); number++) {
        sum = 0;
        int temp = number;
        while (temp != 0) {
            int c = temp % 10;
            sum = sum + c;
            temp = temp / 10;
        }
        if (number % sum == 0) {
            array_factor[factor_elements++] = number;
        }
    }
}

public void run() {
    generate();
    System.out.println("Thread 1 - Numbers whose sum of digits is a factor
of number itself");
    for (int j = start; j < end; j++) {
        try {
            validate1(number[j]);
            int flag = 0;
            for (int i = 0; i < factor_elements; i++) {
                if (number[j] == array_factor[i]) {
                    flag = 1;
                    break;
                }
            }
            if (flag == 1) {
                unique[unique_elements++] = number[j];
            }
        } catch (invalidInputNotRange1 e) {
```

ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC

```
        System.out.println(e);
    }
}
print();
}

public void print() {
    System.out.println("The Numbers that satisfies the conditions are :");
    System.out.println("Printing Numbers whose sum of digits is a factor
of number itself");
    System.out.print("[ ");
    for (int i = 0; i < unique_elements; i++) {
        System.out.print(unique[i] + " ");
    }
    System.out.println("]");
}

}

class Thread2 implements Runnable {
    int[] number;
    int range2_low;
    int range2_high;
    int sum = 0;
    int[] array_factor = new int[100];
    int factor_elements = 0;
    int[] unique = new int[100];
    int unique_elements = 0;
    int start;
    int end;

    Thread2(int[] number, int start, int end, int range2_low, int range2_high)
{
    this.number = number;
    this.range2_high = range2_high;
    this.range2_low = range2_low;
    this.start = start;
    this.end = end;
}

void validate2(int input) throws invalidInputNotRange2 {
    if (input >= range2_low && input <= range2_high) {

    } else {
        throw new invalidInputNotRange2(

```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
        "The Input (" + input + ") is not in the Specified Range
of " + "[" + range2_low + "," + range2_high
                + "]");
    }
}

public void generate() {
    for (int number = range2_low; number < (range2_high + 1); number++) {
        sum = 0;
        int temp = number;
        while (temp != 0) {
            int c = temp % 10;
            sum = sum + c;
            temp = temp / 10;
        }
        int flag = 0;
        for (int i = 2; i < sum; i++) {
            if (sum % i == 0) {
                flag = 1;
                break;
            }
        }
        if (flag == 0) {
            array_factor[factor_elements++] = number;
        }
    }
}

public void run() {
    generate();
    System.out.println("Thread 2 - Numbers whose sum of digits is a
PrimeNumber");
    for (int j = start; j < end; j++) {
        try {
            validate2(number[j]);
            int flag = 0;
            for (int i = 0; i < factor_elements; i++) {
                if (number[j] == array_factor[i]) {
                    flag = 1;
                    break;
                }
            }
            if (flag == 1) {
                unique[unique_elements++] = number[j];
            }
        } catch (invalidInputNotRange2 e) {
            System.out.println(e);
        }
    }
}
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
        }
    }
    print();
}

public void print() {
    System.out.println("The Numbers that satisfies the conditions are :");
    System.out.println("Printing Numbers whose sum of digits is a Prime Number");
    System.out.print("[ ");
    for (int i = 0; i < unique_elements; i++) {
        System.out.print(unique[i] + " ");
    }
    System.out.println("]");
}
}

public class activity7q2 {

    static int range1_low;
    static int range1_high;
    static int range2_low;
    static int range2_high;

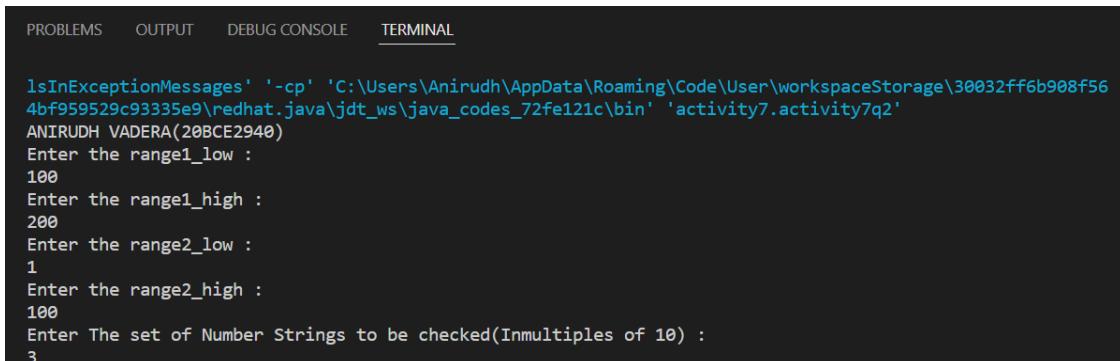
    public static void main(String[] args) throws InterruptedException {
        System.out.println("ANIRUDH VADERA(20BCE2940)");
        Scanner in = new Scanner(System.in);
        System.out.println("Enter the range1_low : ");
        activity7q2.range1_low = in.nextInt();
        System.out.println("Enter the range1_high : ");
        activity7q2.range1_high = in.nextInt();
        System.out.println("Enter the range2_low : ");
        activity7q2.range2_low = in.nextInt();
        System.out.println("Enter the range2_high : ");
        activity7q2.range2_high = in.nextInt();
        System.out.println("Enter The set of Number Strings to be checked(Inmultiples of 10) : ");
        int n = in.nextInt();
        n = n * 10;
        int[] set = new int[n];
        for (int i = 0; i < (n); i++) {
            set[i] = (i + 1) * 10;
        }

        for (int i = 0; i < n / 10; i++) {
            int start = (i * 10);
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
        int end = (i * 10) + 10;
        Thread1 factor = new Thread1(set, start, end, range1_low,
range1_high);
        Thread t1 = new Thread(factor);
        Thread2 prime = new Thread2(set, start, end,
activity7q2.range2_low,
                range2_high);
        Thread t2 = new Thread(prime);
        t1.start();
        t1.join();
        t2.start();
        t2.join();
    }
    System.out.println("Thread 2 - MAIN");
    System.out.println(Thread.currentThread().getName());
    in.close();
}
}
```

OUTPUT:



PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
lsInExceptionMessages' '-cp' 'C:\Users\Anirudh\AppData\Roaming\Code\User\workspaceStorage\30032ff6b908f56
4bf959529c93335e9\redhat.java\jdt_ws\java_codes_72fe121c\bin' 'activity7.activity7q2'
ANIRUDH VADERA(20BCE2940)
Enter the range1_low :
100
Enter the range1_high :
200
Enter the range2_low :
1
Enter the range2_high :
100
Enter The set of Number Strings to be checked(Inmultiples of 10) :
3
```

FOR NUMBER 1-100

OUTPUT FOR THREAD 1 AND 2:

```
Thread 1 - Numbers whose sum of digits is a factor of number itself
activity7.invalidInputNotRange1: The Input (10) is not in the Specified Range of [100,200]
activity7.invalidInputNotRange1: The Input (20) is not in the Specified Range of [100,200]
activity7.invalidInputNotRange1: The Input (30) is not in the Specified Range of [100,200]
activity7.invalidInputNotRange1: The Input (40) is not in the Specified Range of [100,200]
activity7.invalidInputNotRange1: The Input (50) is not in the Specified Range of [100,200]
activity7.invalidInputNotRange1: The Input (60) is not in the Specified Range of [100,200]
activity7.invalidInputNotRange1: The Input (70) is not in the Specified Range of [100,200]
activity7.invalidInputNotRange1: The Input (80) is not in the Specified Range of [100,200]
activity7.invalidInputNotRange1: The Input (90) is not in the Specified Range of [100,200]
The Numbers that satisfies the conditions are :
Printing Numbers whose sum of digits is a factor of number itself
[ 100 ]
Thread 2 - Numbers whose sum of digits is a PrimeNumber
The Numbers that satisfies the conditions are :
Printing Numbers whose sum of digits is a Prime Number
[ 10 20 30 50 70 100 ]
```

ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC

FOR NUMBER 101-200

OUTPUT FOR THREAD 1 AND 2:

```
Thread 1 - Numbers whose sum of digits is a factor of number itself
The Numbers that satisfies the conditions are :
Printing Numbers whose sum of digits is a factor of number itself
[ 110 120 140 150 180 190 200 ]
Thread 2 - Numbers whose sum of digits is a PrimeNumber
activity7.invalidInputNotRange2: The Input (110) is not in the Specified Range of [1,100]
activity7.invalidInputNotRange2: The Input (120) is not in the Specified Range of [1,100]
activity7.invalidInputNotRange2: The Input (130) is not in the Specified Range of [1,100]
activity7.invalidInputNotRange2: The Input (140) is not in the Specified Range of [1,100]
activity7.invalidInputNotRange2: The Input (150) is not in the Specified Range of [1,100]
activity7.invalidInputNotRange2: The Input (160) is not in the Specified Range of [1,100]
activity7.invalidInputNotRange2: The Input (170) is not in the Specified Range of [1,100]
activity7.invalidInputNotRange2: The Input (180) is not in the Specified Range of [1,100]
activity7.invalidInputNotRange2: The Input (190) is not in the Specified Range of [1,100]
activity7.invalidInputNotRange2: The Input (200) is not in the Specified Range of [1,100]
The Numbers that satisfies the conditions are :
Printing Numbers whose sum of digits is a Prime Number
[ ]
```

FOR NUMBER 201-300

OUTPUT FOR THREAD 1 AND 2:

```
Thread 1 - Numbers whose sum of digits is a factor of number itself
activity7.invalidInputNotRange1: The Input (210) is not in the Specified Range of [100,200]
activity7.invalidInputNotRange1: The Input (220) is not in the Specified Range of [100,200]
activity7.invalidInputNotRange1: The Input (230) is not in the Specified Range of [100,200]
activity7.invalidInputNotRange1: The Input (240) is not in the Specified Range of [100,200]
activity7.invalidInputNotRange1: The Input (250) is not in the Specified Range of [100,200]
activity7.invalidInputNotRange1: The Input (260) is not in the Specified Range of [100,200]
activity7.invalidInputNotRange1: The Input (270) is not in the Specified Range of [100,200]
activity7.invalidInputNotRange1: The Input (280) is not in the Specified Range of [100,200]
activity7.invalidInputNotRange1: The Input (290) is not in the Specified Range of [100,200]
activity7.invalidInputNotRange1: The Input (300) is not in the Specified Range of [100,200]
The Numbers that satisfies the conditions are :
Printing Numbers whose sum of digits is a factor of number itself
[ ]
Thread 2 - Numbers whose sum of digits is a PrimeNumber
activity7.invalidInputNotRange2: The Input (210) is not in the Specified Range of [1,100]
activity7.invalidInputNotRange2: The Input (220) is not in the Specified Range of [1,100]
activity7.invalidInputNotRange2: The Input (230) is not in the Specified Range of [1,100]
activity7.invalidInputNotRange2: The Input (240) is not in the Specified Range of [1,100]
activity7.invalidInputNotRange2: The Input (250) is not in the Specified Range of [1,100]
activity7.invalidInputNotRange2: The Input (260) is not in the Specified Range of [1,100]
activity7.invalidInputNotRange2: The Input (270) is not in the Specified Range of [1,100]
activity7.invalidInputNotRange2: The Input (280) is not in the Specified Range of [1,100]
activity7.invalidInputNotRange2: The Input (290) is not in the Specified Range of [1,100]
activity7.invalidInputNotRange2: The Input (300) is not in the Specified Range of [1,100]
The Numbers that satisfies the conditions are :
Printing Numbers whose sum of digits is a Prime Number
[ ]
Thread 2 - MAIN
main
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes> █
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

ACTIVITY – 8(SYNCHRONIZATION)

- Given two array that holds the information about various of stationary items and its cost.

Write a java code that uses three threads, in which the first thread performs the task of removing the details of a particular item which is taken as input from the user and the second thread that performs the task of removing the amount of the given stationary item. The third thread displays the price of the item given the item name.

Assume that the 2 arrays has the following items.

Array 1

Pen	Pencil	Eraser	Sketch	Paper
-----	--------	--------	--------	-------

Array 2

15	5	6	30	10
----	---	---	----	----

QUESTION 1:

CODE:

```
package activity8;

import java.util.Scanner;

class notInRange extends Exception {
    public notInRange(String message) {
        super(message);
    }
}

class Items {
    String[] stationary_items;
    int[] price;
    String[] itemsPriceRemoved = new String[100];
    int counter_removed = 0;

    public void validate(String input) throws notInRange {
        int flag = 0;
        for (int i = 0; i < itemsPriceRemoved.length; i++) {
            if (input.equals(itemsPriceRemoved[i])) {
                flag = 1;
                break;
            }
        }
        if (flag == 1) {
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
        throw new notInRange("This Item price is not available in the
price List : ");
    }
}

Items(String[] stationary_items, int[] price, int[] link, int end) {
    this.stationary_items = stationary_items;
    this.price = price;
}

public synchronized void removeDetails(String itemName) {
    int to_remove = 0;
    for (int i = 0; i < stationary_items.length; i++) {
        if (stationary_items[i].equals(itemName)) {
            to_remove = i;
            break;
        }
    }
    stationary_items[to_remove] = "NULL";
    price[to_remove] = -1;
}

public synchronized void removePrice(String itemName) {
    int to_remove = 0;
    for (int i = 0; i < stationary_items.length; i++) {
        if (stationary_items[i].equals(itemName)) {
            to_remove = i;
            break;
        }
    }
    itemsPriceRemoved[counter_removed++] = stationary_items[to_remove];
    price[to_remove] = -1;
}

public synchronized void Display(String itemName) {
    int to_display = 0;
    int flag_t = 0;
    for (int i = 0; i < stationary_items.length; i++) {
        if (stationary_items[i].equals(itemName)) {
            to_display = i;
            flag_t = 1;
            break;
        }
    }
    if (flag_t == 0) {
        System.out.println("The item doesnt exist or is deleted from the
list : ");
    }
}
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
        } else {
            System.out.println("The Item Name is : " +
stationary_items[to_display]);
            try {
                validate(stationary_items[to_display]);
                System.out.println("The Item Price is : " +
price[to_display]);
            } catch (notInRange e) {
                e.printStackTrace();
            }
        }
    }

}

public class activity8q1 {

    public static void main(String[] args) throws InterruptedException {
        Scanner in = new Scanner(System.in);
        System.out.println("ANIRUDH VADERA(20BCE2940)");
        System.out.println("Enter the Number of Items : ");
        int n = in.nextInt();
        String[] stationary_items = new String[n];
        int[] price = new int[n];
        int[] link = new int[n];
        for (int i = 0; i < n; i++) {
            link[i] = i;
        }
        System.out.println("Enter the initial Item Details : ");
        for (int i = 0; i < n; i++) {
            System.out.println("Item : " + (i + 1));
            System.out.print("Item Name : ");
            in.nextLine();
            stationary_items[i] = in.nextLine();
            System.out.print("Item Price : ");
            price[i] = in.nextInt();
        }
        in.nextLine();
        System.out.print("Item Name to be removed : ");
        String item1 = in.nextLine();
        Items items = new Items(stationary_items, price, link, n);
        Thread t1 = new Thread(() -> {
            items.removeDetails(item1);
        });
        System.out.print("Item Price to be removed : ");
        String item2 = in.nextLine();
        Thread t2 = new Thread(() -> {
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
        items.removePrice(item2);
    });
System.out.print("Item to be Displayed : ");
String item3 = in.nextLine();
Thread t3 = new Thread(() -> {
    try {
        Thread.sleep(500);
    } catch (InterruptedException e) {
        System.out.println(e);
    }
    items.Display(item3);
});
t1.start();
t2.start();
t3.start();
t1.join();
t2.join();
t3.join();
in.close();
}
}
```

ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC

OUTPUT:

When item name is available:

```
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes> c;; cd 'c:\Users\Anirudh\OneDrive\Desktop\java_codes'; & 'C:\Liberica-JDK-17\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Anirudh\AppData\Roaming\Code\User\workspaceStorage\30032ff6b908f564bf959529c93335e9\redhat.java\jdt_ws\java_codes_72fe121c\bin' 'activity8.activity8q1'
ANIRUDH VADERA (20BCE2940)
Enter the Number of Items :
5
Enter the initial Item Details :
Item : 1
Item Name : Pen
Item Price : 15
Item : 2
Item Name : Pencil
Item Price : 5
Item : 3
Item Name : Eraser
Item Price : 6
Item : 4
Item Name : Sketch
Item Price : 30
Item : 5
Item Name : Paper
Item Price : 10
Item Name to be removed : Sketch
Item Price to be removed : 10
Item to be Displayed : Eraser
The Item Name is : Eraser
The Item Price is : 6
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes>
```

When item name is not available:

```
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes> c;; cd 'c:\Users\Anirudh\OneDrive\Desktop\java_codes'; & 'C:\Liberica-JDK-17\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Anirudh\AppData\Roaming\Code\User\workspaceStorage\30032ff6b908f564bf959529c93335e9\redhat.java\jdt_ws\java_codes_72fe121c\bin' 'activity8.activity8q1'
ANIRUDH VADERA (20BCE2940)
Enter the Number of Items :
5
Enter the initial Item Details :
Item : 1
Item Name : Pen
Item Price : 15
Item : 2
Item Name : Pencil
Item Price : 5
Item : 3
Item Name : Eraser
Item Price : 6
Item : 4
Item Name : Sketch
Item Price : 30
Item : 5
Item Name : Paper
Item Price : 10
Item Name to be removed : Sketch
Item Price to be removed : 10
Item to be Displayed : Sketch
The item doesn't exist or is deleted from the list :
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes>
```

ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC

When item price is not available:

```
Data\Roaming\Code\User\workspaceStorage\30032ff6b908f564bf959529c93335e9\redhat.java\jdt_ws\java_codes_72
fe121c\bin' 'activity8.activity8q1'
ANIRUDH VADERA (20BCE2940)
Enter the Number of Items :
5
Enter the initial Item Details :
Item : 1
Item Name : Pen
Item Price : 15
Item : 2
Item Name : Pencil
Item Price : 5
Item : 3
Item Name : Eraser
Item Price : 6
Item : 4
Item Name : Sketch
Item Price : 30
Item : 5
Item Name : Paper
Item Price : 10
Item Name to be removed : Sketch
Item Price to be removed : Paper
Item to be Displayed : Paper
The Item Name is : Paper
lenght = Paper
activity8.notInRange: This Item price is not available in the price List :
    at activity8.Items.validate(activity8q1.java:27)
    at activity8.Items.Display(activity8q1.java:75)
    at activity8.activity8q1.lambda$2(activity8q1.java:127)
    at java.base/java.lang.Thread.run(Thread.java:833)
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes> █
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

QUESTION 2:

2. Consider a Bus Ticket Registration class given below that has the following data members

- BusName-String
- Number of Seats-Integer

The class has a parameterized constructor to initialize an object and the following methods (

The object is initialized with the given values [Bus Name-Red Bus, Number of seats-0].

Register_Seat- This method will deduct 1 seat from the Number of Seats.

Allot_Seat(int Seats)- This method will get the number of seats to be allocated and add those seats to NumberOfSeats.

Create the following threads and demonstrate registration and allocation of seats.

Thread 1-Will try and register a seat for the Red Bus

Thread 2-Will try and register a seat for the Red Bus

Thread 3-Will allocate 60 seats to the Bus

Thread 4- Print the total remaining seats after registration is complete

Thread 1 and Thread2 will be in wait if the value for Number of Seats is 0.

Thread 3 will notify other threads after allocating the seats

CODE:

```
package activity8;

class BusTicketRegistration {
    String BusName;
    int noOfSeats;

    BusTicketRegistration(String BusName, int noOfSeats) {
        this.BusName = BusName;
        this.noOfSeats = noOfSeats;
    }

    public synchronized void Register_Seat() {
        while (noOfSeats <= 0) {
            System.out.println("Waiting.... " +
Thread.currentThread().getId());
            try {
                wait();
            }
        }
    }
}
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
    noOfSeats = noOfSeats - 1;
}

public synchronized void Allot_Seat(int seatsToAllocate) {
    System.out.println("ALLOCATING SEATS.....");
    noOfSeats = noOfSeats + seatsToAllocate;
    notifyAll();
}
}

public class activity8q2 {
    public static void main(String[] args) throws InterruptedException {
        System.out.println("ANIRUDH VADERA (20BCE2940)");
        BusTicketRegistration registration = new BusTicketRegistration("Red
Bus", 0);
        Thread t1 = new Thread(() -> {
            registration.Register_Seat();
        });
        Thread t2 = new Thread(() -> {
            registration.Register_Seat();
        });
        Thread t3 = new Thread(() -> {
            registration.Allot_Seat(60);
        });
        Thread t4 = new Thread(() -> {

            try {
                Thread.sleep(500);
            } catch (InterruptedException e) {
                System.out.println(e);
            }
            System.out.println("The total Seats after registration is : " +
registration.noOfSeats);
        });
        t1.start();
        t2.start();
        t3.start();
        t4.start();
        t1.join();
        t2.join();
        t3.join();
        t4.join();
    }
}
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
        System.out.println("Main Thread :");  
    }  
}
```

OUTPUT:

First the two threads wait for allocation of the seats and then they execute.

```
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes> c:; cd 'c:\Users\Anirudh\OneDrive\Desktop\java_codes';  
& 'C:\Libérica-JDK-17\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Anirudh\App  
fe121c\bin' 'activity8.activity8q2'  
ANIRUDH VADERA (20BCE2940)  
Waiting.... 23  
Waiting.... 24  
ALLOCATING SEATS.....  
The total Seats after registration is : 58  
Main Thread :  
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes>
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

QUESTION 3:

3. A famous bakery in Vellore prepares cakes for their customers. The shop makes cake and stacks it up in a vessel, and the customer buys them. The max capacity of the vessel is 10. If the cakes are emptied, customers wait for the baker to prepare new cakes. Develop a program for the given scenario using inter thread communication concept in Java programming.

CODE:

```
package activity8;

class Cake {
    int buffer;
    int capicity;

    Cake(int capicity, int buffer) {
        this.capicity = capicity;
        this.buffer = buffer;
    }

    public synchronized void produceCake(int piece) {

        while ((piece + buffer) > capicity) {
            try {
                wait();
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
            System.out.println("Allowing Producer to consume more cakes : ");
        }
        System.out.println("Producing....Number : " + piece);
        buffer = buffer + piece;
        notifyAll();
    };

    public synchronized void consumeCake(int piece) {
        while (piece > buffer) {
            System.out.println("Waiting...To Produce Cakes...");
            notifyAll();
            try {
                wait();
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    }
}
```

ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC

```
        }
    }
    buffer = buffer - piece;
    System.out.println("Consuming.....Number : " + piece);
};

}

public class activity8q3 {
    public static void main(String[] args) throws InterruptedException {
        System.out.println("ANIRUDH VADERA (20BCE2940)");
        Cake c = new Cake(10, 8);
        Thread t1 = new Thread(() -> {
            for (int i = 0; i < 20; i++) {
                c.consumeCake(1);
            }
        });
        Thread t2 = new Thread(() -> {
            for (int i = 0; i < 10; i++) {
                c.produceCake(2);
            }
        });
        t1.start();
        t2.start();
    }
}
```

ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC

OUTPUT:

First the consumer thread consumes the cakes

Then after the buffer is 0 it waits for more cakes to be produced

After enough cakes has been produced it starts consuming again

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL

Data\Roaming\Code\User\workspaceStorage\30032ff6b908f564bf959529c93335e9\redhat.java\jdt_ws\java_codes_72
fe121c\bin' 'activity8.activity8q3'
ANIRUDH VADERA (20BCE2940)
Consuming.....Number : 1
Waiting...To Produce Cakes...
Producing....Number : 2
Consuming.....Number : 1
Waiting...To Produce Cakes...
Allowing Producer to consume more cakes :
Producing....Number : 2
Consuming.....Number : 1
Consuming.....Number : 1
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes> █
```

ACTIVITY – 9:(FILES)

QUESTION 1:

1. Write a Java Program to Reverse the Contents of a File and Print it.(Use FileInputStream and FileOutputStream)

CODE:

```
package activity9;

import java.io.FileInputStream;
import java.io.FileOutputStream;

public class activity9q1 {
    public static void main(String[] args) {
        System.out.println("ANIRUDH VADERA (20BCE2940)");
        String Content = new String("");
        int i;
        int[] Content_in_integer = new int[10000];
        int end = 0;

        // Reading the initial Content

        try {
            FileInputStream fin = new FileInputStream(
                "C:/Users/Anirudh/OneDrive/Desktop/java_codes/activity9/activity9q1.txt");
            while ((i = fin.read()) != -1) {
                Content_in_integer[end++] = i;
                Content = Content + ((char) i);
            }
            System.out.println("The Contents of the File is : " + Content);
            fin.close();
        } catch (Exception e) {
            e.printStackTrace();
        }

        // Writing the reversed Content in the file

        try {
            FileOutputStream fout = new FileOutputStream(
                "C:/Users/Anirudh/OneDrive/Desktop/java_codes/activity9/activity9q1.txt");
            for (int j = (end - 1); j >= 0; j--) {
                fout.write(Content_in_integer[j]);
            }
        }
    }
}
```

ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC

```
        }
        fout.close();
        System.out.println("Success... Written the reversed Content : ");
    } catch (Exception e) {
        System.out.println(e);
    }

    // Reading the Final Content

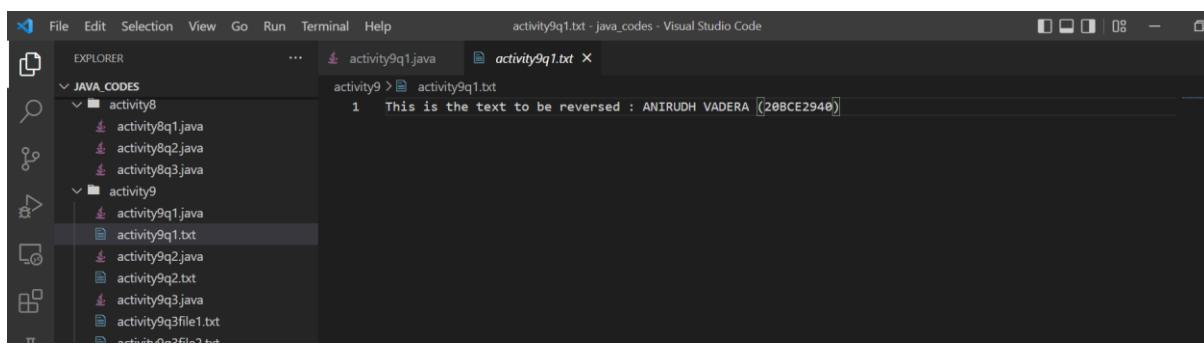
    try {
        FileInputStream fin = new FileInputStream(
            "C:/Users/Anirudh/OneDrive/Desktop/java_codes/activity9/ac
tivity9q1.txt");
        Content = "";
        end = 0;
        while ((i = fin.read()) != -1) {
            Content_in_integer[end++] = i;
            Content = Content + ((char) i);
        }
        System.out.println("The Contents of the File after reversing is :
" + Content);
        fin.close();
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC

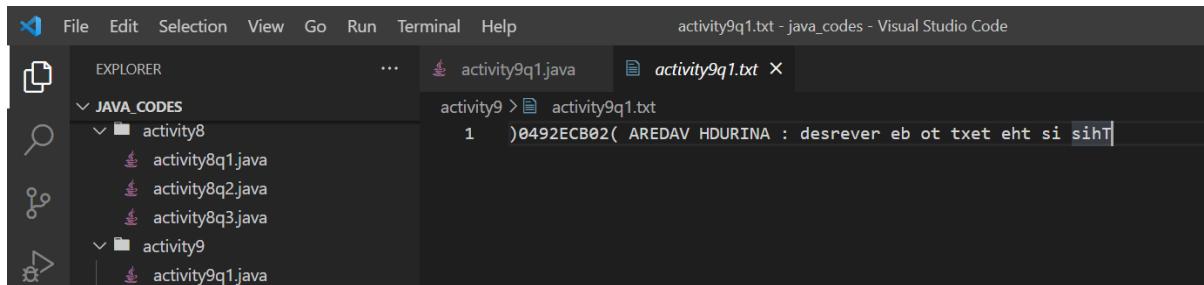
OUTPUT:

```
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes> c:; cd 'c:\Users\Anirudh\OneDrive\Desktop\java_codes'; & 'C:\Liberica-JDK-17\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Anirudh\AppData\Roaming\Code\User\workspaceStorage\30032ff6b908f564bf959529c93335e9\redhat.java\jdt_ws\java_codes_72fe121c\bin' 'activity9.activity9q1'
ANIRUDH VADERA (20BCE2940)
The Contents of the File is : This is the text to be reversed : ANIRUDH VADERA (20BCE2940)
Success... Written the reversed Content :
The Contents of the File after reversing is : )0492ECB02( AREDAV HDURINA : desrever eb ot txet eht si sihT
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes>
```

Text file previously:



Text file after reversing:



ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC

QUESTION 2:

2. Write a Java Program to read and write English alphabets(A-Z) into a file and read it from the console(Use BufferedReader and Writer)

CODE:

```
package activity9;

import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.FileWriter;
import java.io.InputStreamReader;

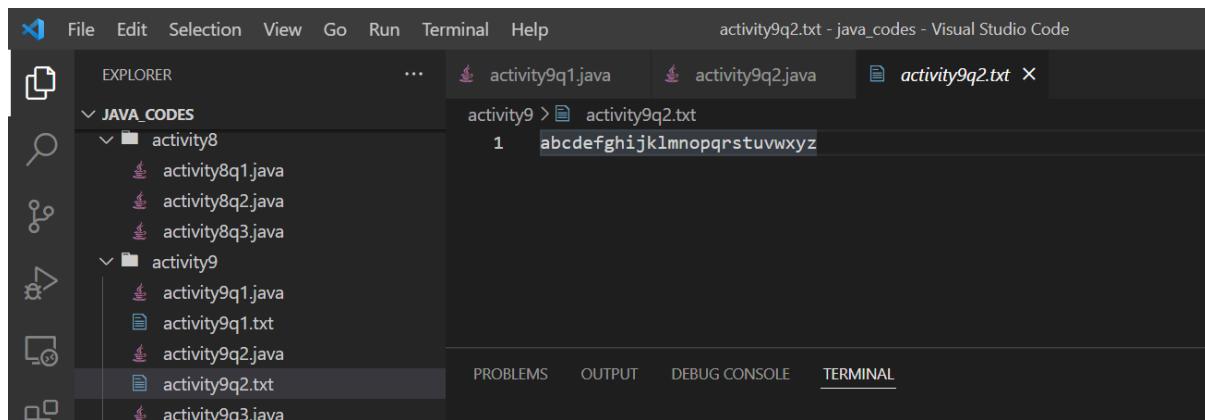
public class activity9q2 {
    public static void main(String[] args) {
        System.out.println("ANIRUDH VADERA (20BCE2940)");
        try (FileWriter writer = new FileWriter(
            "C:/Users/Anirudh/OneDrive/Desktop/java_codes/activity9/activity9q2.txt")) {
            BufferedWriter buffer = new BufferedWriter(writer);
            BufferedReader reader = new BufferedReader(new
InputStreamReader(System.in)));
            String towrite = reader.readLine();
            buffer.write(towrite);
            System.out.println("Success in writing the alphabets...");
            buffer.flush();
            buffer.close();
        } catch (
            Exception e) {
            e.printStackTrace();
        }
    }
}
```

ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC

OUTPUT:

```
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes> & 'C:\Liberica-JDK-17\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Anirudh\AppData\Roaming\Code\User\workspaceStorage\30032ff6b908f564bf959529c93335e9\redhat.java\jdt_ws\java_codes_72fe121c\bin' 'activity9.activity9q2'
ANIRUDH VADERA (20BCE2940)
abcdefghijklmnopqrstuvwxyz
Success in writing the alphabets...
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes>
```

The output File:



ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

QUESTION 3:

1. Create Threads that will perform the following:

Thread 1 – Writes all prime numbers between 1 and 100 to the file file1.txt

Thread2- Writes all prime numbers between 101 and 200 to the file2.txt

(Write operations to the files should be paused for 1 second after each write.)

Thread3- Reads and displays prime numbers from file1.txt

Thread4- Reads and displays prime numbers from file2.txt

Thread3 and Thread4 should start the read operations after the write operations to the file is complete. (Use DataOutputStream and DataInputStream for the file write and read operations)

CODE:

```
package activity9;

import java.io.DataInputStream;
import java.io.DataOutputStream;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;

class WriteThread implements Runnable {

    String filePath;
    int flag = 0;
    DataOutputStream dout = null;
    int start;
    int end;
    int file;

    WriteThread(String filePath, int start, int end, int file) {
        this.filePath = filePath;
        this.start = start;
        this.end = end;
        this.file = file;
    }

    public void run() {

        synchronized (WriteThread.class) {
            try {
                dout = new DataOutputStream(new FileOutputStream(filePath));
            }
        }
    }
}
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
        } catch (FileNotFoundException e) {
            e.printStackTrace();
        }
        for (int i = start; i < end; i++) {
            flag = 0;
            for (int j = 2; j < i; j++) {
                if (i % j == 0) {
                    flag = 1;
                    break;
                }
            }
            if (flag == 0) {
                if (i != 1) {
                    try {
                        dout.writeInt(i);
                        dout.flush();
                        System.out.println("Waiting 1 sec after
writing.....prime : " + i);
                        try {
                            Thread.sleep(1000);
                        } catch (InterruptedException e) {
                            System.out.println(e);
                        }
                    } catch (IOException e) {
                        e.printStackTrace();
                    }
                }
            }
        }
    }
    try {
        dout.close();
    } catch (IOException e) {
        e.printStackTrace();
    }

    if (flag == 1) {
        System.out.println("Successfully written prime numbers from 1
to 100 in file1...");
    } else {
        System.out.println("Successfully written prime numbers from
101 to 201 in file2...");
    }
}
}

class ReadThread implements Runnable {
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
String filePath;
DataInputStream din = null;
int primes[] = new int[100];
int end = 0;
int i;
int file; // 1 - file1 2 - file2

ReadThread(String filePath, int file) {
    this.filePath = filePath;
    this.file = file;
}

public void run() {
    // Waiting for Write operations to complete before reading

    synchronized (ReadThread.class) {
        try {
            Thread.sleep(1000);
        } catch (InterruptedException e) {
            System.out.println(e);
        }
        try {
            din = new DataInputStream(new FileInputStream(filePath));
        } catch (FileNotFoundException e) {
            e.printStackTrace();
        }
        try {
            while ((din.available() > 0)) {
                i = din.readInt();
                primes[end++] = i;
            }
            din.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
        if (file == 1) {
            System.out.println("The Primes are (From File1) : ");
            System.out.print("[ ");
            for (int i = 0; i < end; i++) {
                System.out.print(primes[i] + " ");
            }
            System.out.println("]");
        } else {
            System.out.println("The Primes are (From File2) : ");
            System.out.print("[ ");
            for (int i = 0; i < end; i++) {
```

ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC

```
        System.out.print(primes[i] + " ");
    }
    System.out.println("]");
}
}

public class activity9q3 {
    public static void main(String[] args) throws InterruptedException {
        System.out.println("ANIRUDH VADERA (20BCE2940)");
        WriteThread writeFile1 = new WriteThread(
            "C:/Users/Anirudh/OneDrive/Desktop/java_codes/activity9/activity9q3file1.txt", 1, 101, 1);
        Thread t1 = new Thread(writeFile1);

        WriteThread writeFile2 = new WriteThread(
            "C:/Users/Anirudh/OneDrive/Desktop/java_codes/activity9/activity9q3file2.txt", 101, 201, 2);
        Thread t2 = new Thread(writeFile2);

        ReadThread readFile1 = new ReadThread(
            "C:/Users/Anirudh/OneDrive/Desktop/java_codes/activity9/activity9q3file1.txt", 1);
        Thread t3 = new Thread(readFile1);
        ReadThread readFile2 = new ReadThread(
            "C:/Users/Anirudh/OneDrive/Desktop/java_codes/activity9/activity9q3file2.txt", 2);
        Thread t4 = new Thread(readFile2);

        t1.start();
        t2.start();
        t1.join();
        t2.join();
        t3.start();
        t4.start();
        t3.join();
        t4.join();
    }
}
```

ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC

OUTPUT:

Writing 1 to 100 in file1:

```
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes> c:; cd 'c:\Users\Anirudh\OneDrive\Desktop\java_codes'; & 'C:\Liberica-JDK-17\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Anirudh\AppData\Roaming\Code\User\workspaceStorage\30032ff6b908f564bf959529c93335e9\redhat.java\jdt_ws\java_codes_72fe121c\bin' 'activity9.activity9q3'
ANIRUDH VADERA (20BCE2940)
Waiting 1 sec after writing.....prime : 2
Waiting 1 sec after writing.....prime : 3
Waiting 1 sec after writing.....prime : 5
Waiting 1 sec after writing.....prime : 7
Waiting 1 sec after writing.....prime : 11
Waiting 1 sec after writing.....prime : 13
Waiting 1 sec after writing.....prime : 17
Waiting 1 sec after writing.....prime : 19
Waiting 1 sec after writing.....prime : 23
Waiting 1 sec after writing.....prime : 29
Waiting 1 sec after writing.....prime : 31
Waiting 1 sec after writing.....prime : 37
Waiting 1 sec after writing.....prime : 41
Waiting 1 sec after writing.....prime : 43
Waiting 1 sec after writing.....prime : 47
Waiting 1 sec after writing.....prime : 53
Waiting 1 sec after writing.....prime : 59
Waiting 1 sec after writing.....prime : 61
Waiting 1 sec after writing.....prime : 67
Waiting 1 sec after writing.....prime : 71
Waiting 1 sec after writing.....prime : 73
Waiting 1 sec after writing.....prime : 79
Waiting 1 sec after writing.....prime : 83
Waiting 1 sec after writing.....prime : 89
Waiting 1 sec after writing.....prime : 97
Successfully written prime numbers from 1 to 100 in file1...
```

Writing 101 to 201 in file1:

```
Waiting 1 sec after writing.....prime : 101
Waiting 1 sec after writing.....prime : 103
Waiting 1 sec after writing.....prime : 107
Waiting 1 sec after writing.....prime : 109
Waiting 1 sec after writing.....prime : 113
Waiting 1 sec after writing.....prime : 127
Waiting 1 sec after writing.....prime : 131
Waiting 1 sec after writing.....prime : 137
Waiting 1 sec after writing.....prime : 139
Waiting 1 sec after writing.....prime : 149
Waiting 1 sec after writing.....prime : 151
Waiting 1 sec after writing.....prime : 157
Waiting 1 sec after writing.....prime : 163
Waiting 1 sec after writing.....prime : 167
Waiting 1 sec after writing.....prime : 173
Waiting 1 sec after writing.....prime : 179
Waiting 1 sec after writing.....prime : 181
Waiting 1 sec after writing.....prime : 191
Waiting 1 sec after writing.....prime : 193
Waiting 1 sec after writing.....prime : 197
Waiting 1 sec after writing.....prime : 199
Successfully written prime numbers from 101 to 201 in file2...
```

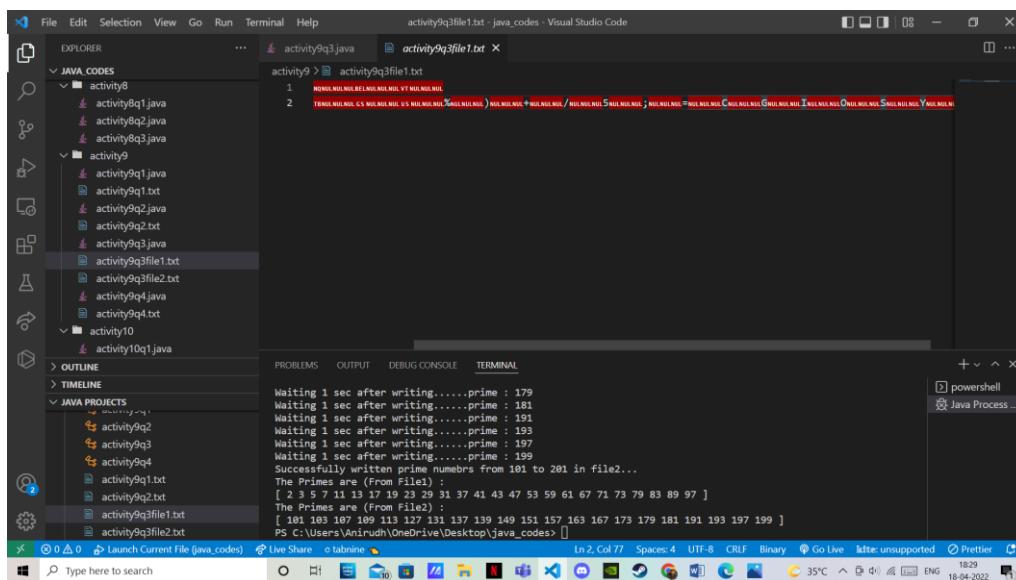
ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC**

Reading from file:

```
Successfully written prime numbers from 101 to 201 in file2...
The Primes are (From File1) :
[ 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97 ]
The Primes are (From File2) :
[ 101 103 107 109 113 127 131 137 139 149 151 157 163 167 173 179 181 191 193 197 199 ]
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes>
```

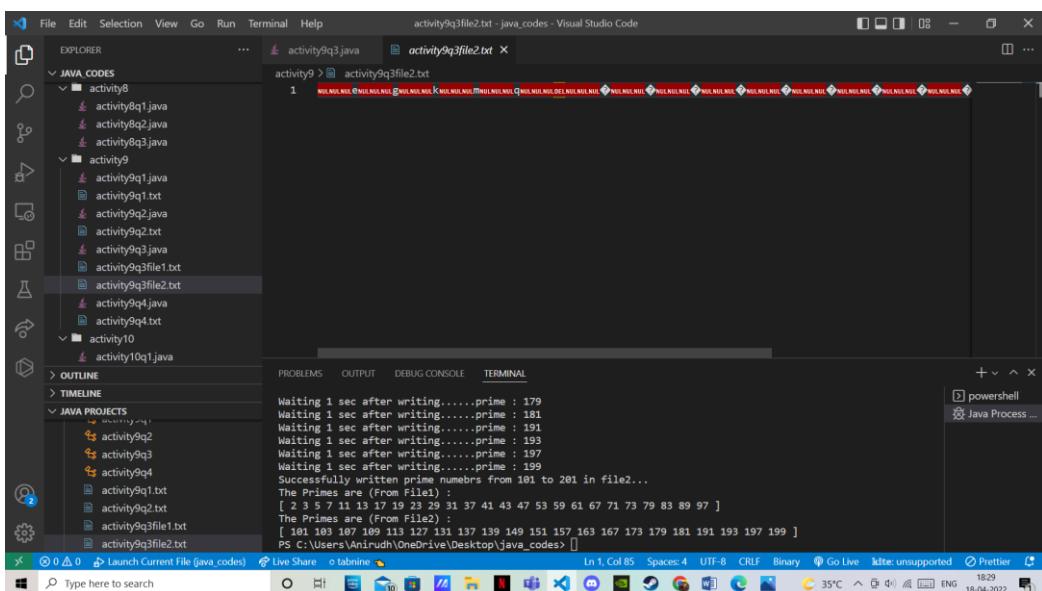
File Contents:

File 1:



```
activity9 > activity9q3file1.txt
1
2
Waiting 1 sec after writing.....prime : 179
Waiting 1 sec after writing.....prime : 181
Waiting 1 sec after writing.....prime : 191
Waiting 1 sec after writing.....prime : 193
Waiting 1 sec after writing.....prime : 197
Waiting 1 sec after writing.....prime : 199
Successfully written prime numbers from 101 to 201 in file2...
The Primes are (From File1) :
[ 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97 ]
The Primes are (From File2) :
[ 101 103 107 109 113 127 131 137 139 149 151 157 163 167 173 179 181 191 193 197 199 ]
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes>
```

File 2:



```
activity9 > activity9q3file2.txt
1
Waiting 1 sec after writing.....prime : 179
Waiting 1 sec after writing.....prime : 181
Waiting 1 sec after writing.....prime : 191
Waiting 1 sec after writing.....prime : 193
Waiting 1 sec after writing.....prime : 197
Waiting 1 sec after writing.....prime : 199
Successfully written prime numbers from 101 to 201 in file2...
The Primes are (From File1) :
[ 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97 ]
The Primes are (From File2) :
[ 101 103 107 109 113 127 131 137 139 149 151 157 163 167 173 179 181 191 193 197 199 ]
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes>
```

ANIRUDH VADERA

(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER THREADING,FILES, COLLECTION, JAVAFX AND JDBC

QUESTION 4:

2. Create threads that will perform the following:

Thread1- Reads all stop words ("a","and","the" are considered as stop words here) from the file file1.txt and displays number of times each stop word occurs in the file.

Thread2- Reads all words other than stop words from the file2.txt and displays number of times the word that start with Character H is part of the file.

Tread3- displays Thanks for using our software to the user after the file read operations in Thread1 and Thread2 is complete.

Use Scanner object to perform file reads and set the Priority for Thread1 as 3, Thread 2 as 2 and Thread3 as 1 in your code.

CODE:

```
package activity9;

import java.io.File;
import java.io.FileNotFoundException;
import java.util.Scanner;

class readFile implements Runnable {

    int type;
    String filePath;

    readFile(int type, String filePath) {
        this.type = type;
        this.filePath = filePath;
    }

    public void run() {
        synchronized (readFile.class) {
            File file = new File(filePath);
            Scanner in = null;
            try {
                in = new Scanner(file);
                in.useDelimiter(" ");
                if (type == 1) {
                    String[] stop = { "a", "and", "the" };
                    int[] times = new int[3];
                    while (in.hasNext()) {
                        String word = in.next();
                        if (word.equals(stop[0])) {
                            times[0] = times[0] + 1;
                        }
                    }
                }
            } catch (FileNotFoundException e) {
                e.printStackTrace();
            }
        }
    }
}
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
        }
        if (word.equals(stop[1])) {
            times[1] = times[1] + 1;
        }
        if (word.equals(stop[2])) {
            times[2] = times[2] + 1;
        }
    }

System.out.println("Displaying the Number of Stop Words :");
for (int i = 0; i < stop.length; i++) {
    System.out.println("Stop Word : " + stop[i] + " ::" +
Number of times appeared : " + times[i]);
}

} else {
    String[] stop = { "a", "and", "the" };
    int times = 0;
    while (in.hasNext()) {
        String word = in.next();
        if (word.equals(stop[0]) || word.equals(stop[1]) ||
word.equals(stop[2])) {
            } else {
                if (word.startsWith("H")) {
                    times++;
                }
            }
        }
    System.out.println("Number of times Non Stop words
starting with 'H' Appeared : " + times);
}
} catch (FileNotFoundException e) {
    e.printStackTrace();
} finally {
    in.close();
}
}

}

public class activity9q4 {
public static void main(String[] args) throws InterruptedException {
    System.out.println("ANIRUDH VADERA (20BCE2940)");
    readFile readThread1 = new readFile(1,
        "C:/Users/Anirudh/OneDrive/Desktop/java_codes/activity9/activi
ty9q4file1.txt");
}
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
Thread t1 = new Thread(readThread1);
readFile readThread2 = new readFile(2,
        "C:/Users/Anirudh/OneDrive/Desktop/java_codes/activity9/activity9q4file2.txt");
Thread t2 = new Thread(readThread2);
Thread t3 = new Thread(() -> {
    System.out.println("Waiting for Thread 1 and 2 to complete ....");
    try {
        Thread.sleep(2000);
    } catch (InterruptedException e) {
        System.out.println(e);
    }
    System.out.println("Thank you for using our Software ...");
});

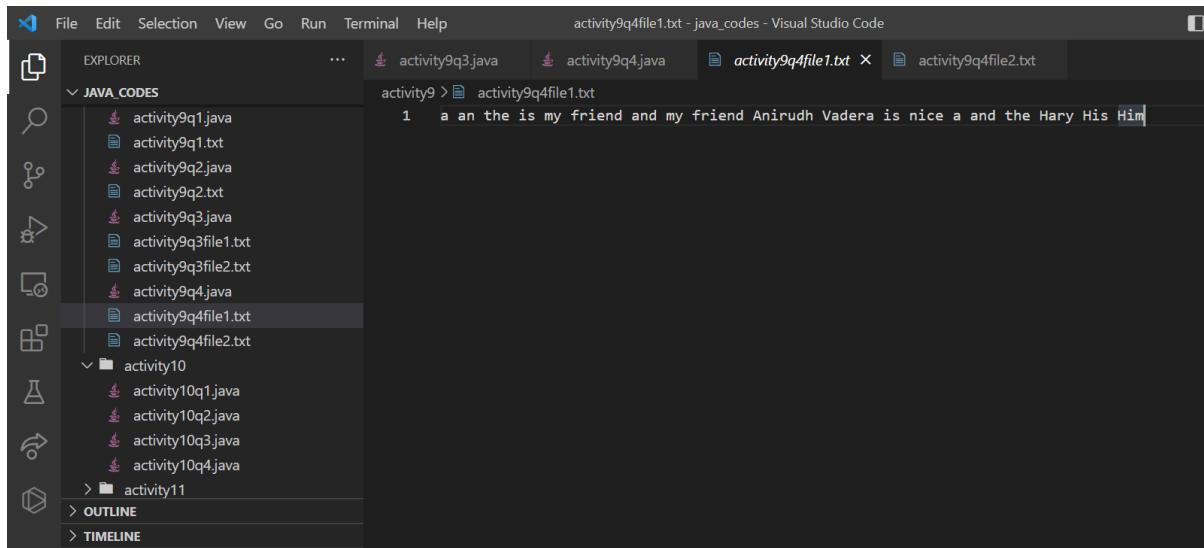
t1.setPriority(3);
t2.setPriority(2);
t3.setPriority(1);

t1.start();
t2.start();
t3.start();
t1.join();
t2.join();
t3.join();
}
}
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC**

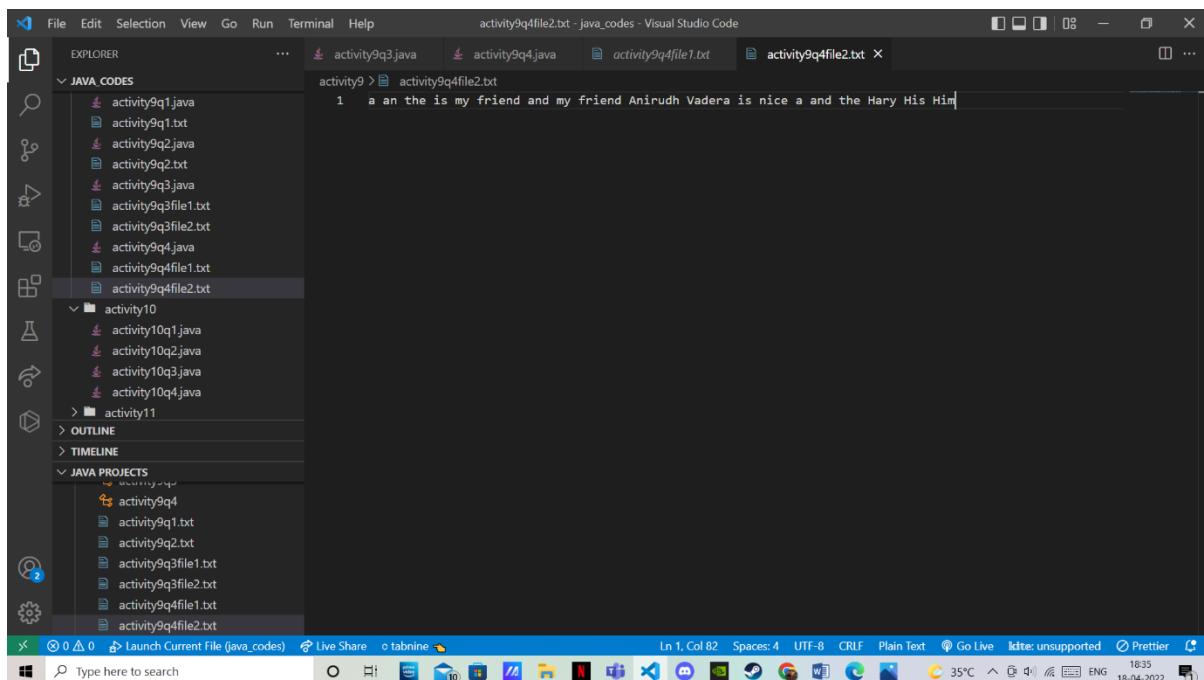
OUTPUT:

File 1 content:



The screenshot shows the Visual Studio Code interface with the title bar "activity9q4file1.txt - java_codes - Visual Studio Code". The Explorer sidebar on the left shows a "JAVA CODES" folder containing files like activity9q1.java, activity9q1.txt, etc. The main editor area displays the content of activity9q4file1.txt, which contains the text: "1 a an the is my friend and my friend Anirudh Vadera is nice a and the Harry His Him".

File 2 content:



The screenshot shows the Visual Studio Code interface with the title bar "activity9q4file2.txt - java_codes - Visual Studio Code". The Explorer sidebar on the left shows a "JAVA CODES" folder and a "JAVA PROJECTS" section for a project named "activity9q4". The main editor area displays the content of activity9q4file2.txt, which contains the same text as File 1: "1 a an the is my friend and my friend Anirudh Vadera is nice a and the Harry His Him". The status bar at the bottom shows various toolbars and system information.

ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC

Output:

```
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes> c:; cd 'c:\Users\Anirudh\OneDrive\Desktop\java_codes'; & 'C:\Liberica-JDK-17\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Anirudh\AppData\Roaming\Code\User\workspaceStorage\30032ff6b908f564bf959529c93335e9\redhat.java\jdt_ws\java_codes_72fe121c\bin' 'activity9.activity9q4'
ANIRUDH VADERA (20BCE2940)
Waiting for Thread 1 and 2 to complete ....
Displaying the Number of Stop Words :
Stop Word : a :: Number of times appeared : 2
Stop Word : and :: Number of times appeared : 2
Stop Word : the :: Number of times appeared : 2
Number of times Non Stop words starting with 'H' Appeared : 3
Thank you for using our Software ...
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes>
```

ACTIVITY – 10:(Collection)

QUESTION 1:

1. Write a java program for the following:-

Make an arraylist containing a few Strings. Sort it by

- Length (i.e., shortest to longest)
- Reverse length (i.e., longest to shortest)
- Alphabetically by the first character only
- Strings that contain "e" first, everything else second. For now, put the code directly in the Lambda

CODE:

```
package activity10;

import java.util.ArrayList;
import java.util.Comparator;

public class activity10q1 {
    public static void main(String[] args) {
        ArrayList<String> list = new ArrayList<String>();
        System.out.println("ANIRUDH VADERA (20BCE2940)");
        list.add("Volkswagen");
        list.add("Toyota");
        list.add("Porsche");
        list.add("Ferrari");
        list.add("Mercedes-Benz");
        list.add("Audi");
        list.add("Rolls-Royce");
        list.add("BMW");
        list.add("E_Coe");
        System.out.println("Before Sorting: " + list);
        list.sort((p1, p2) -> p1.length() - p2.length());
        System.out.println("After Sorting(Length): " + list);
        list.sort((p1, p2) -> p2.length() - p1.length());
        System.out.println("After Sorting(Reverse Length): " + list);
        list.sort((str1, str2) -> str1.charAt(0) - str2.charAt(0));
        System.out.println("After Sorting(First Character): " + list);
        list.sort(Comparator.comparingInt(a -> (a.startsWith("E") ||
a.startsWith("E") ? 0 : 1)));
        System.out.println("After Sorting(First e): " + list);
    }
}
```

ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC

OUTPUT:

```
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes> c:; cd 'c:\Users\Anirudh\OneDrive\Desktop\java_codes'; & 'C:\Liberica-JDK-17\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Anirudh\AppData\Roaming\activity10.activity10q1'
ANIRUDH VADERA (20BCE2940)
Before Sorting: [Volkswagen, Toyota, Porsche, Ferrari, Mercedes-Benz, Audi, Rolls-Royce, BMW, E_Coe]
After Sorting(Length): [BMW, Audi, E_Coe, Toyota, Porsche, Ferrari, Volkswagen, Rolls-Royce, Mercedes-Benz]
After Sorting(Reverse Length): [Mercedes-Benz, Rolls-Royce, Volkswagen, Porsche, Ferrari, Toyota, E_Coe, Audi, BMW]
After Sorting(First Character): [Audi, BMW, E_Coe, Ferrari, Mercedes-Benz, Porsche, Rolls-Royce, Toyota, Volkswagen]
After Sorting(First e): [E_Coe, Audi, BMW, Ferrari, Mercedes-Benz, Porsche, Rolls-Royce, Toyota, Volkswagen]
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes>
```

QUESTION 2:

2. Write a java program using Lambda to "capitalize" a string. Change the first letter of each word in the string to upper case (if it is not already upper case). For example, a capitalized version of "vellore institute of technology" is "Vellore Institute Of Technology ". Write a method named printCapitalized that will print a capitalized version of a string to standard output. The string to be printed should be a parameter to the method. Test your method with a main() routine that gets a line of input from the user and applies the method to it.

CODE:

```
package activity10;

import java.util.Scanner;

interface Line {
    public void printCapitalized(String s);
}

class Worker implements Line {
    public void printCapitalized(String s) {
        StringBuffer str = new StringBuffer(s);

        if (str.charAt(0) >= 97 && str.charAt(0) <= 122)
            str.setCharAt(0, (char) ((int) str.charAt(0) - 32));

        for (int i = 0; i < str.length(); i++) {
            if (str.charAt(i) == ' ') {
                if (str.charAt(i + 1) >= 97 && str.charAt(i + 1) <= 122)
                    str.setCharAt(i + 1, (char) ((int) str.charAt(i + 1) -
32));
            }
        }
        System.out.println(str.toString());
    }
}

public class activity10q2 {
    static void doWork(Line w) {
        Scanner sc = new Scanner(System.in);
        w.printCapitalized(sc.nextLine());
    }

    public static void main(String[] args) {
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
doWork(  
    (String s) -> {  
        StringBuffer str = new StringBuffer(s);  
  
        if (str.charAt(0) >= 97 && str.charAt(0) <= 122)  
            str.setCharAt(0, (char) ((int) str.charAt(0) - 32));  
  
        for (int i = 0; i < str.length(); i++) {  
            if (str.charAt(i) == ' ') {  
                if (str.charAt(i + 1) >= 97 && str.charAt(i + 1)  
                    <= 122)  
                    str.setCharAt(i + 1, (char) ((int)  
str.charAt(i + 1) - 32));  
            }  
        }  
        System.out.println(str.toString());  
    });  
    System.out.println("ANIRUDH VADERA (20BCE2940)");  
}  
  
}
```

OUTPUT:

```
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes> cd 'c:\Users\Anirudh\OneDrive\Desktop\java_codes'; & 'C:\Liberica-JDK-17\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Anirudh\AppData\Roaming\Code\User\workspaceStorage\b30032ff6b908f564bf959529c93335e9\redhat.java\jdt_ws\java_codes_72fe121c\bin' 'activity10.activity10q2'  
vellore institute of technology  
Vellore Institute Of Technology  
ANIRUDH VADERA (20BCE2940)  
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes>
```

QUESTION 3:

3. Write a java program to create class **Books** with the data members **book name, author, price, type(fiction, comic, cooking)**. Use input methods to get the input values. Create three array list (fiction, comic and cooking).
- a. Depending upon the type of the book, insert the book object into the respective list.
 - b. Display the list of books in each type.
 - c. Sort the list of books in each list with respect to their book name.
 - d. Display the min and max priced books of each list.

CODE:

```
package activity10;

import java.util.List;
import java.util.ArrayList;
import java.util.Scanner;

class Book {
    String bookName;
    String author;
    int price;
    String type;

    Book() {
        bookName = "";
        author = "";
        price = 0;
        type = "";
    }

    Book(Book b) {
        this.bookName = b.bookName;
        this.type = b.type;
        this.price = b.price;
        this.author = b.author;
    }

    void input(String bookName, String author, int price, String type) {
        this.author = author;
        this.bookName = bookName;
        this.price = price;
        this.type = type;
    }
}
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
}

public class activity10q3 {
    static List<Book> fiction = new ArrayList<>();
    static List<Book> comic = new ArrayList<>();
    static List<Book> cooking = new ArrayList<>();
    static int book_count;

    static void sort_by_Bookname(List<Book> list) {
        for (int i = 0; i < list.size() - 1; i++) {
            int min = i;
            for (int j = i + 1; j < list.size(); j++) {
                if ((list.get(j).bookName).compareTo((list.get(min).bookName)) < 0)
                    min = j;
            }
            if (min != i) {
                Book temp = list.get(i);
                list.set(i, list.get(min));
                list.set(min, temp);
            }
        }
    }

    static Book maxPriced(List<Book> list, boolean flag) {
        Book max = list.get(0);
        for (Book i : list) {
            if (flag == true && i.price > max.price)
                max = i;
            else if (flag == false && i.price < max.price)
                max = i;
        }
        return max;
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the number of books : ");
        book_count = sc.nextInt();
        sc.nextLine();

        Book temp = new Book();
        for (int i = 0; i < book_count; i++) {

            System.out.println("Enter book name : ");
            temp.bookName = sc.nextLine();
            System.out.println("Enter author name : ");
        }
    }
}
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
temp.author = sc.nextLine();
System.out.println("Enter price : ");
temp.price = sc.nextInt();
sc.nextLine();
System.out.println("Enter book type (fiction, comic, cooking) :");
");
temp.type = sc.nextLine();

Book insert = new Book(temp);

if (temp.type.equals("comic"))
    comic.add(insert);
else if (temp.type.equals("fiction"))
    fiction.add(insert);
else if (temp.type.equals("cooking"))
    cooking.add(insert);
else {
    System.out.println("Wrong type entered!");
    System.exit(0);
}
}

System.out.println("Comic books : ");
for (Book i : comic)
    System.out.print(i.bookName + "\t");
System.out.println();

System.out.println("Fiction books : ");
for (Book i : fiction)
    System.out.print(i.bookName + "\t");
System.out.println();

System.out.println("Cooking books : ");
for (Book i : cooking)
    System.out.print(i.bookName + "\t");
System.out.println();

sort_by_Bookname(comic);
System.out.println("(After Sorting) Comic books : ");
for (Book i : comic)
    System.out.print(i.bookName + "\t");
System.out.println();

sort_by_Bookname(fiction);
System.out.println("(After Sorting) Fiction books : ");
for (Book i : fiction)
    System.out.print(i.bookName + "\t");
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
System.out.println();

sort_by_Bookname(cooking);
System.out.println("(After Sorting)Cooking books : ");
for (Book i : cooking)
    System.out.print(i.bookName + "\t");
System.out.println();

System.out.println("Max priced comic book : " + maxPriced(comic,
true).bookName + "Price : "
+ maxPriced(comic, true).price);
System.out.println("Min priced comic book : " + maxPriced(comic,
false).bookName + "Price : "
+ maxPriced(comic, false).price);

System.out.println("Max priced fiction book : " + maxPriced(fiction,
true).bookName + "Price : "
+ maxPriced(fiction, true).price);
System.out.println("Min priced fiction book : " + maxPriced(fiction,
false).bookName + "Price : "
+ maxPriced(fiction, false).price);

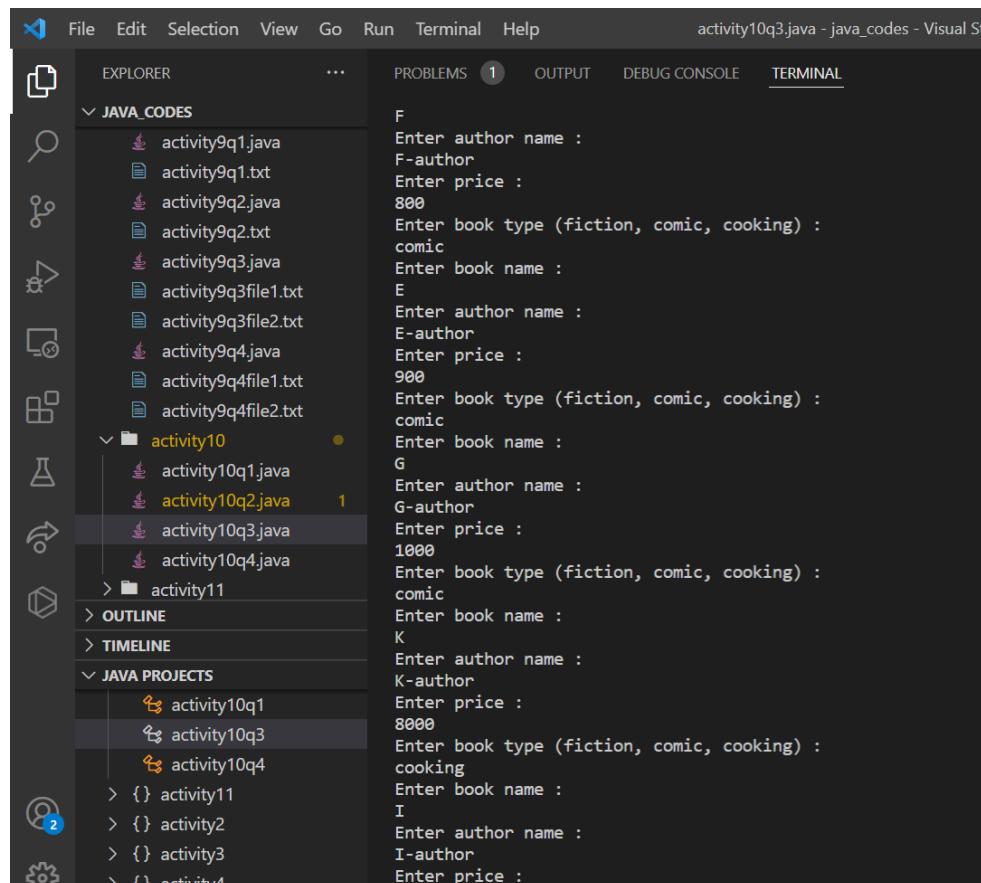
System.out.println("Max priced cooking book : " + maxPriced(cooking,
true).bookName + "Price : "
+ maxPriced(cooking, true).price);
System.out.println("Min priced cooking book : " + maxPriced(cooking,
false).bookName + "Price : "
+ maxPriced(cooking, false).price);
System.out.println("ANIRUDH VADERA (20BCE2940)");
sc.close();
}
}
```

ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC

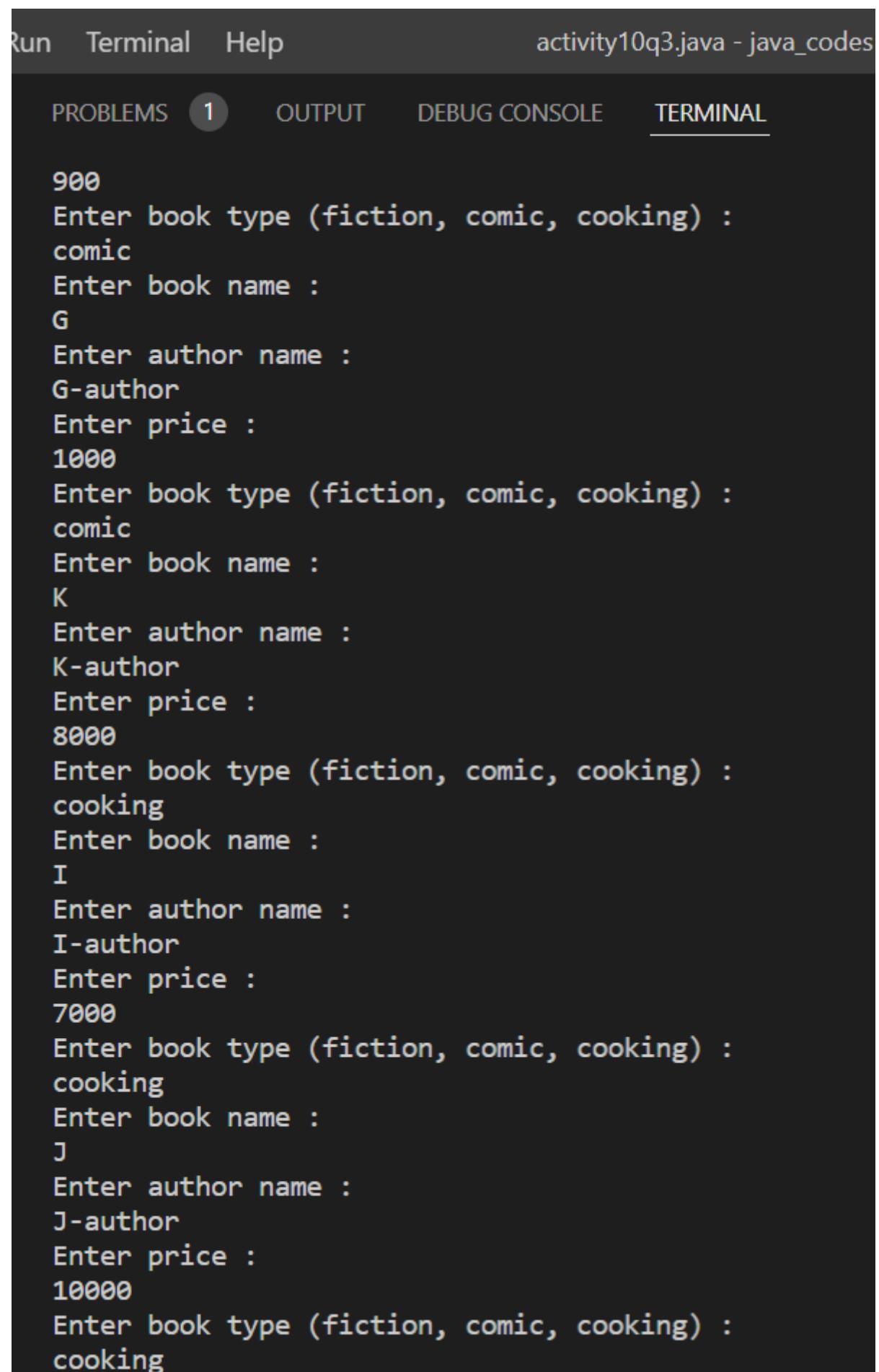
OUTPUT:

Entering the details:

```
ANIRUDH VADERA (20BCE2940)
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes> cd 'c:\Users\Anirudh\OneDrive\Desktop\java_codes'; & 'C:\Libe
rice-JDK-17\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Anirudh\AppData\Roaming\Code\Us
er\workspaceStorage\30032ff6b908f564bf959529c93335e9\redhat.java\jdt_ws\java_codes_72fe121c\bin' 'activity10.acti
vity10q3'
Enter the number of books :
10
Enter book name :
D
Enter author name :
D-author
Enter price :
2000
Enter book type (fiction, comic, cooking) :
fiction
Enter book name :
B
Enter author name :
B-author
Enter price :
100
Enter book type (fiction, comic, cooking) :
fiction
Enter book name :
A
Enter author name :
A-author
Enter price :
400
Enter book type (fiction, comic, cooking) :
fiction
Enter book name :
C
Enter author name :
C-author
Enter price :
```



ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC**



The screenshot shows a terminal window with a dark background and light-colored text. At the top, there are tabs for "Run", "Terminal", and "Help", followed by the path "activity10q3.java - java_codes". Below the tabs, there are four buttons: "PROBLEMS" (with a value of 1), "OUTPUT", "DEBUG CONSOLE", and "TERMINAL" (which is underlined). The main area of the terminal displays the following Java code execution:

```
900
Enter book type (fiction, comic, cooking) :
comic
Enter book name :
G
Enter author name :
G-author
Enter price :
1000
Enter book type (fiction, comic, cooking) :
comic
Enter book name :
K
Enter author name :
K-author
Enter price :
8000
Enter book type (fiction, comic, cooking) :
cooking
Enter book name :
I
Enter author name :
I-author
Enter price :
7000
Enter book type (fiction, comic, cooking) :
cooking
Enter book name :
J
Enter author name :
J-author
Enter price :
10000
Enter book type (fiction, comic, cooking) :
cooking
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC**

List of Book in Each type:

```
Comic books :  
F      E      G  
Fiction books :  
D      B      A      C  
Cooking books :  
K      I      J
```

After sorting List of Book in each type:

```
(After Sorting)Comic books :  
E      F      G  
(After Sorting)Fiction books :  
A      B      C      D  
(After Sorting)Cooking books :  
I      J      K
```

Min and Max priced book of each type:

```
Max priced comic book : GPrice : 1000  
Min priced comic book : FPrice : 800  
Max priced fiction book : DPrice : 2000  
Min priced fiction book : BPrice : 100  
Max priced cooking book : JPrice : 10000  
Min priced cooking book : IPrice : 7000  
ANIRUDH VADERA (20BCE2940)  
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes>
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC**

QUESTION 4:

4. Assume only a maximum of 3 courses can be registered by a student for week end semester classes. Create a hashmap 'h1' with 'n' key-value pairs where keys are the names of students and values are the courses registered by them. Create another hashmap 'h2' with 'm'key-value pairs where keys are the names of courses offered for B.Tech and values are the names of faculty handling the courses. Write appropriate code to

- Add or remove a student from h1
- Iterate over the maps and display the key-value pairs stored in them
- Given a student name, fetch the names of all those who teach him/her.

Eq:
if the elements of h1 are

Stud name	Courses registered
A	Python, maths, c
B	c, c++
C	C C++, physics,chemistry

If the elements of H2 are

Course name	Faculty
C	aaa
C++	bbb
Java	ccc
Python	ddd

For student B, it has to display the faculty aaa and bbb

CODE:

```
package activity10;  
  
import java.util.*;  
  
public class activity10q4 {  
    Map<String, List<String>> h1 = new HashMap<>();  
    Map<String, String> h2 = new HashMap<>();
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
int n, m;

void insert_to_studentList(String student, String course) {
    if (h1.containsKey(student)) {
        List<String> courses = new ArrayList<>(h1.get(student));
        courses.add(course);
        h1.put(student, courses);
    } else {
        List<String> courseTemp = new ArrayList<>();
        courseTemp.add(course);
        h1.put(student, courseTemp);
    }
}

void removeStudent(String student) {
    if (h1.containsKey(student))
        h1.remove(student);
}

void showMaps() {
    System.out.println("Student Name\tCourses");
    Set<Map.Entry<String, List<String>>> keySet = h1.entrySet();
    for (Map.Entry<String, List<String>> set : keySet)
        System.out.println(set.getKey() + "\t" + set.getValue());

    System.out.println("Courses\tTeachers");
    Set<Map.Entry<String, String>> newSet = h2.entrySet();
    for (Map.Entry<String, String> set : newSet)
        System.out.println(set.getKey() + "\t" + set.getValue());
}

void showFaculty(String student) {
    System.out.println("Teachers for " + student);
    if (h1.containsKey(student))
        for (String course : h1.get(student))
            System.out.print(h2.get(course) + "\t");
}

public static void main(String[] args) {
    activity10q4 q = new activity10q4();
    Scanner sc = new Scanner(System.in);

    System.out.println("Enter n : ");
    q.n = sc.nextInt();

    System.out.println("Enter m : ");
}
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC**

```
q.m = sc.nextInt();

sc.nextLine();

System.out.println("Enter student details");
for (int i = 0; i < q.n; i++) {
    System.out.println("Enter student name : ");
    String name = sc.nextLine();
    System.out.println("Enter " + name + "'s courses : (Press e to
stop)");
    while (true) {
        String course = sc.nextLine();
        if (course.length() == 1 && course.charAt(0) == 'e')
            break;
        q.insert_to_studentList(name, course);
    }
}

System.out.println("Enter faculty details");
for (int i = 0; i < q.m; i++) {
    System.out.println("Enter course name : ");
    String name = sc.nextLine();
    System.out.println("Enter " + name + " faculty name : ");
    q.h2.put(name, sc.nextLine());
}

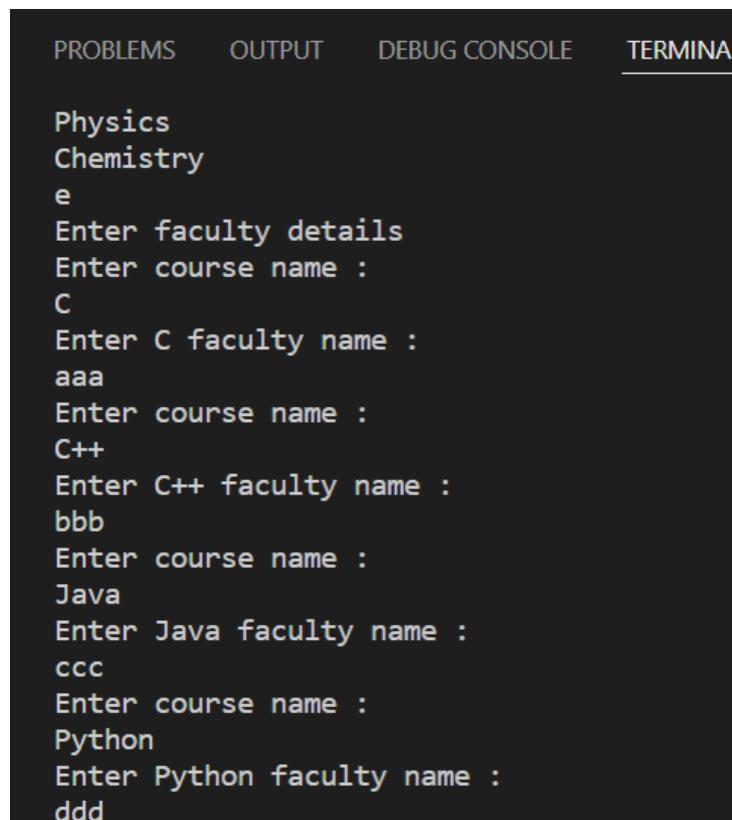
q.showMaps();
System.out.println("Enter the student name whose faculty names you
want to view : ");
String name = sc.nextLine();
q.showFaculty(name);
System.out.println("ANIRUDH VADERA (20BCE2940)");
sc.close();
}
```

ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC

OUTPUT:

Entering the details of the student:

```
ANIRUDH VADERA (20BCE2940)
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes> cd 'c:\Users\Anirudh\OneDrive\Desktop\java_codes'; & 'C:\Libe
rica-JDK-17\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Anirudh\AppData\Roaming\Code\Us
er\workspaceStorage\30032ff6b908f564bf959529c93335e9\redhat.java\jdt_ws\java_codes_72fe121c\bin' 'activity10.activi
ty10q4'
Enter n :
3
Enter m :
4
Enter student details
Enter student name :
A
Enter A's courses : (Press e to stop)
Python
Maths
C
e
Enter student name :
B
Enter B's courses : (Press e to stop)
C
C++
e
Enter student name :
C
Enter C's courses : (Press e to stop)
C++
Physics
Chemistry
e
Enter faculty details
Enter course name :
C
Enter C faculty name :
aaa
Enter course name :
```



The screenshot shows a terminal window with four tabs at the top: PROBLEMS, OUTPUT, DEBUG CONSOLE, and TERMINAL. The TERMINAL tab is active, displaying the following Java code execution:

```
PROBLEMS      OUTPUT      DEBUG CONSOLE      TERMINAL

Physics
Chemistry
e
Enter faculty details
Enter course name :
C
Enter C faculty name :
aaa
Enter course name :
C++
Enter C++ faculty name :
bbb
Enter course name :
Java
Enter Java faculty name :
ccc
Enter course name :
Python
Enter Python faculty name :
ddd
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC**

Iterating through maps and showing the stored details:

```
Student Name    Courses
A      [Python, Maths, C]
B      [C, C++]
C      [C++, Physics, Chemistry]
Courses Teachers
Java    ccc
C++    bbb
C      aaa
Python  ddd
```

Adding a student:

```
Enter the student name whom you want to add :
D
Enter D's courses : (Press e to stop)
Python
C
e
Student Name    Courses
A      [Python, Maths, C]
B      [C, C++]
C      [C++, Physics, Chemistry]
D      [Python, C]
Courses Teachers
Java    ccc
C++    bbb
C      aaa
Python  ddd
```

Removing a student:

```
Enter the student name whom you want to remove :
D
Student Name    Courses
A      [Python, Maths, C]
B      [C, C++]
C      [C++, Physics, Chemistry]
Courses Teachers
Java    ccc
C++    bbb
C      aaa
Python  ddd
```

ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC

Given a student name fetching details of the teacher:

```
Enter the student name whose faculty names you want to view :  
B  
Teachers for B  
aaa      bbb  
ANIRUDH VADERA (20BCE2940)  
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes> █
```

ACTIVITY – 11:(JAVAFX)

QUESTION 1:

1. Write a JavaFX program to create a page with 5 buttons named "red", "green", "blue", "pink", "black", on clicking the button, print the color name in the console. In the same program, add necessary code to change the color of button based on the name of the button.

CODE:

```
package activity11;

import javafx.application.Application;
import javafx.event.ActionEvent;
import javafx.event.EventHandler;
import javafx.geometry.HPos;
import javafx.geometry.Pos;
import javafx.geometry.VPos;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.scene.layout.GridPane;
import javafx.stage.Stage;

public class activity11q1 extends Application {

    public void start(Stage primaryStage) {
        int[] flag = { 0, 0, 0, 0, 0 };
        Button btn1 = new Button();
        btn1.setText("Red");
        Button btn2 = new Button();
        btn2.setText("Green");
        Button btn3 = new Button();
        btn3.setText("Blue");
        Button btn4 = new Button();
        btn4.setText("Pink");
        Button btn5 = new Button();
        btn5.setText("Black");

        btn1.setOnAction(new EventHandler<ActionEvent>() {

            @Override
            public void handle(ActionEvent event) {
                if (flag[0] == 0) {
                    flag[0] = 1;
                }
                else if (flag[1] == 0) {
                    flag[1] = 1;
                }
                else if (flag[2] == 0) {
                    flag[2] = 1;
                }
                else if (flag[3] == 0) {
                    flag[3] = 1;
                }
                else if (flag[4] == 0) {
                    flag[4] = 1;
                }
            }
        });
    }
}
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
        btn2.setStyle("");
        btn3.setStyle("");
        btn4.setStyle("");
        btn5.setStyle("");
        flag[1] = 0;
        flag[2] = 0;
        flag[3] = 0;
        flag[4] = 0;

    } else {
        flag[0] = 0;
    }
System.out.println("Red");
if (flag[0] == 1) {
    btn1.setStyle("-fx-background-color: red;-fx-text-fill:
white; ");
} else {
    btn1.setStyle("");
}
}

});

btn2.setOnAction(new EventHandler<ActionEvent>() {

    public void handle(ActionEvent event) {
        if (flag[1] == 0) {
            flag[1] = 1;
            btn1.setStyle("");
            btn3.setStyle("");
            btn4.setStyle("");
            btn5.setStyle("");
            flag[0] = 0;
            flag[2] = 0;
            flag[3] = 0;
            flag[4] = 0;
        } else {
            flag[1] = 0;
        }
System.out.println("Green");
if (flag[1] == 1) {
    btn2.setStyle("-fx-background-color: green;-fx-text-fill:
white; ");
} else {
    btn2.setStyle("");
}
    }
});
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
btn3.setOnAction(new EventHandler<ActionEvent>() {

    public void handle(ActionEvent event) {
        if (flag[2] == 0) {
            flag[2] = 1;
            btn2.setStyle("");
            btn1.setStyle("");
            btn4.setStyle("");
            btn5.setStyle("");
            flag[1] = 0;
            flag[0] = 0;
            flag[3] = 0;
            flag[4] = 0;
        } else {
            flag[2] = 0;
        }
        System.out.println("Blue");
        if (flag[2] == 1) {
            btn3.setStyle("-fx-background-color: blue;-fx-text-fill:
white; ");
        } else {
            btn3.setStyle("");
        }
    }
});

btn4.setOnAction(new EventHandler<ActionEvent>() {

    public void handle(ActionEvent event) {
        if (flag[3] == 0) {
            flag[3] = 1;
            btn2.setStyle("");
            btn3.setStyle("");
            btn1.setStyle("");
            btn5.setStyle("");
            flag[1] = 0;
            flag[2] = 0;
            flag[0] = 0;
            flag[4] = 0;
        } else {
            flag[3] = 0;
        }
        System.out.println("Pink");
        if (flag[3] == 1) {
            btn4.setStyle("-fx-background-color: pink;-fx-text-fill:
white; ");
        }
    }
});
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
        } else {
            btn4.setStyle("");
        }
    });
}

btn5.setOnAction(new EventHandler<ActionEvent>() {
    public void handle(ActionEvent event) {
        if (flag[4] == 0) {
            flag[4] = 1;
            btn2.setStyle("");
            btn3.setStyle("");
            btn4.setStyle("");
            btn1.setStyle("");
            flag[1] = 0;
            flag[2] = 0;
            flag[3] = 0;
            flag[0] = 0;
        } else {
            flag[4] = 0;
        }
        System.out.println("Black");
        if (flag[4] == 1) {
            btn5.setStyle("-fx-background-color: black;-fx-text-fill:
white; ");
        } else {
            btn5.setStyle("");
        }
    }
});
btn1.setMaxSize(100, 200);
btn2.setMaxSize(100, 200);
btn3.setMaxSize(100, 200);
btn4.setMaxSize(100, 200);
btn5.setMaxSize(100, 200);
GridPane root = new GridPane();
root.add(btn1, 0, 0);
root.add(btn2, 0, 1);
root.add(btn3, 1, 0);
root.add(btn4, 1, 1);
root.add(btn5, 0, 2);

Scene scene = new Scene(root, 300, 250);
root.setVgap(10);
root.setHgap(10);
root.setAlignment(Pos.CENTER);
GridPane.setAlignment(btn1, HPos.CENTER);
```

ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC

```
        GridPane.setAlignment(btn1, VPos.CENTER);
        GridPane.setAlignment(btn2, HPos.CENTER);
        GridPane.setAlignment(btn2, VPos.CENTER);
        GridPane.setAlignment(btn3, HPos.CENTER);
        GridPane.setAlignment(btn3, VPos.CENTER);
        GridPane.setAlignment(btn4, HPos.CENTER);
        GridPane.setAlignment(btn4, VPos.CENTER);
        GridPane.setAlignment(btn5, HPos.CENTER);
        GridPane.setAlignment(btn5, VPos.CENTER);

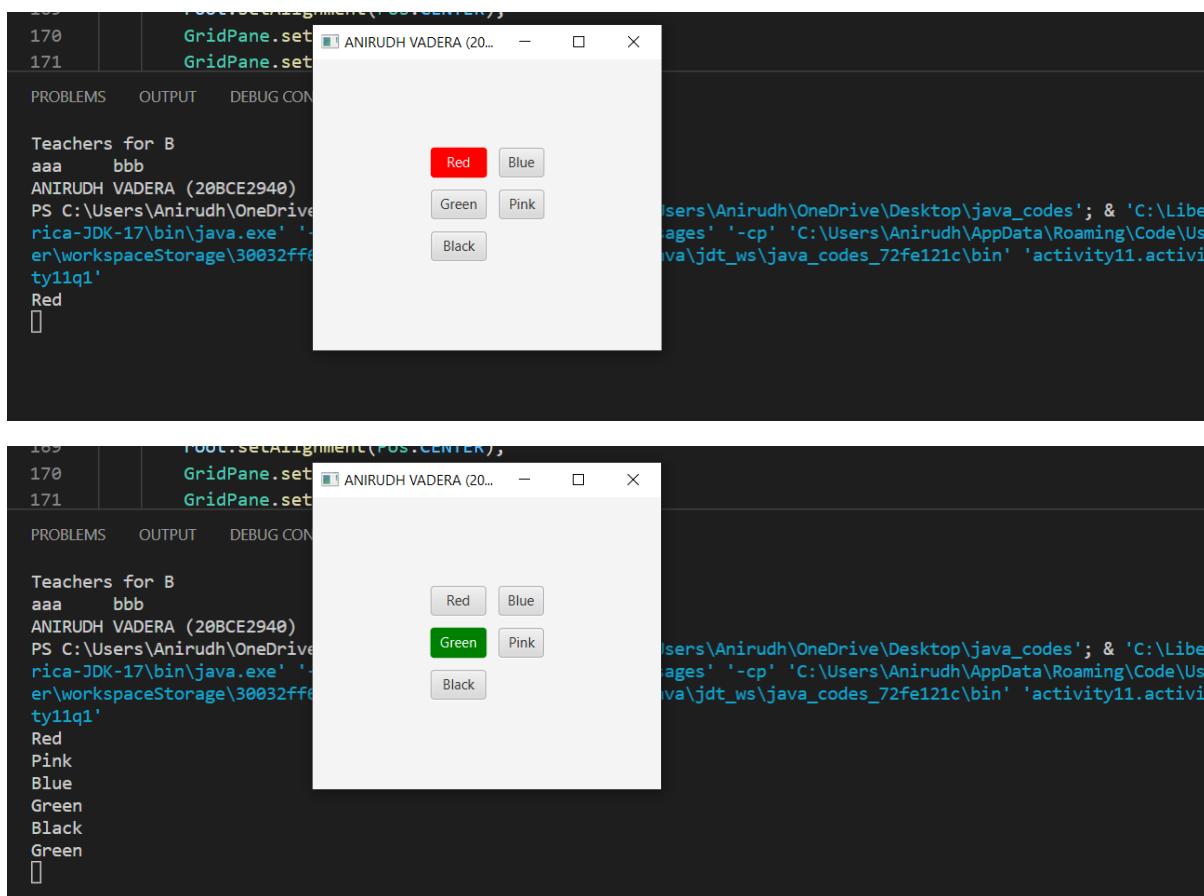
    primaryStage.setTitle("ANIRUDH VADERA (20BCE2940)");
    primaryStage.setScene(scene);
    primaryStage.show();
}

public static void main(String[] args) {
    launch(args);
}

}
```

ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC

OUTPUT:



QUESTION 2:

2. Develop an Air ticket booking Home page by using Text and Button objects (Source, Destination, Depature Date, No.of passengers, Class(Business, First, Economy)with appropriate Layout to describe about the booking. On clicking the button “Book”, display the collected information on the next page.

CODE:

```
package activity11;

import javafx.application.Application;
import javafx.event.ActionEvent;
import javafx.event.EventHandler;
import javafx.geometry.HPos;
import javafx.geometry.Pos;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.scene.control.ComboBox;
import javafx.scene.control.Label;
import javafx.scene.control.TextField;
import javafx.scene.layout.GridPane;
import javafx.stage.Stage;
import javafx.scene.text.Text;
```

```
public class activity11q2 extends Application {
    public void start(Stage primaryStage) throws Exception {
        GridPane rootNode = new GridPane();
        rootNode.setHgap(5);
        rootNode.setVgap(5);
        rootNode.setAlignment(Pos.CENTER);

        Scene myScene = new Scene(rootNode, 700, 500);

        rootNode.add(new Label("Enter Source:"), 0, 0);
        TextField source = new TextField();
        rootNode.add(source, 1, 0);
        rootNode.add(new Label("Enter Destination:"), 0, 1);
        TextField dest = new TextField();
        rootNode.add(dest, 1, 1);
        rootNode.add(new Label("Enter Date of departure:"), 0, 2);
        TextField dod = new TextField();
        rootNode.add(dod, 1, 2);
        rootNode.add(new Label("Enter Number of passengers:"), 0, 3);
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
TextField nop = new TextField();
rootNode.add(nop, 1, 3);
rootNode.add(new Label("Enter class: "), 0, 4);
ComboBox comboBox = new ComboBox();
comboBox.getItems().add("Business");
comboBox.getItems().add("First");
comboBox.getItems().add("Economy");
rootNode.add(comboBox, 1, 4);
Button btn = new Button("Book");
rootNode.add(btn, 1, 5);
GridPane.setAlignment(btn, HPos.LEFT);

GridPane rootNode2 = new GridPane();
Scene myScene2 = new Scene(rootNode2, 700, 500);
rootNode2.setHgap(5);
rootNode2.setVgap(5);
rootNode2.setAlignment(Pos.CENTER);
Button btn2 = new Button("Edit details");
int flag = 0;
btn.setOnAction(new EventHandler<ActionEvent>() {
    public void handle(ActionEvent event) {
        String s = "";
        if (flag == 0) {
            try {
                String src = source.getText();
                String dest1 = dest.getText();
                String dod1 = dod.getText();
                String nop1 = nop.getText();
                String cls1 = (String) comboBox.getValue();

                if (src.equals("") || dest1.equals("") ||
dod1.equals("") || nop1.equals("") ||
                    || cls1.equals("")){
                    s = "Please fill all the fields";
                } else {
                    s = "Source: " + src + "\nDestination: " + dest1 +
"\nDate of departure: " +
                    dod1
                    + "\nNumber of passengers: " + nop1 +
"\nClass: " + cls1;
                }
            Text t = new Text(s);

            rootNode2.add(t, 1, 1);
            rootNode2.add(btn2, 1, 2);

        } catch (Exception e) {
```

ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC

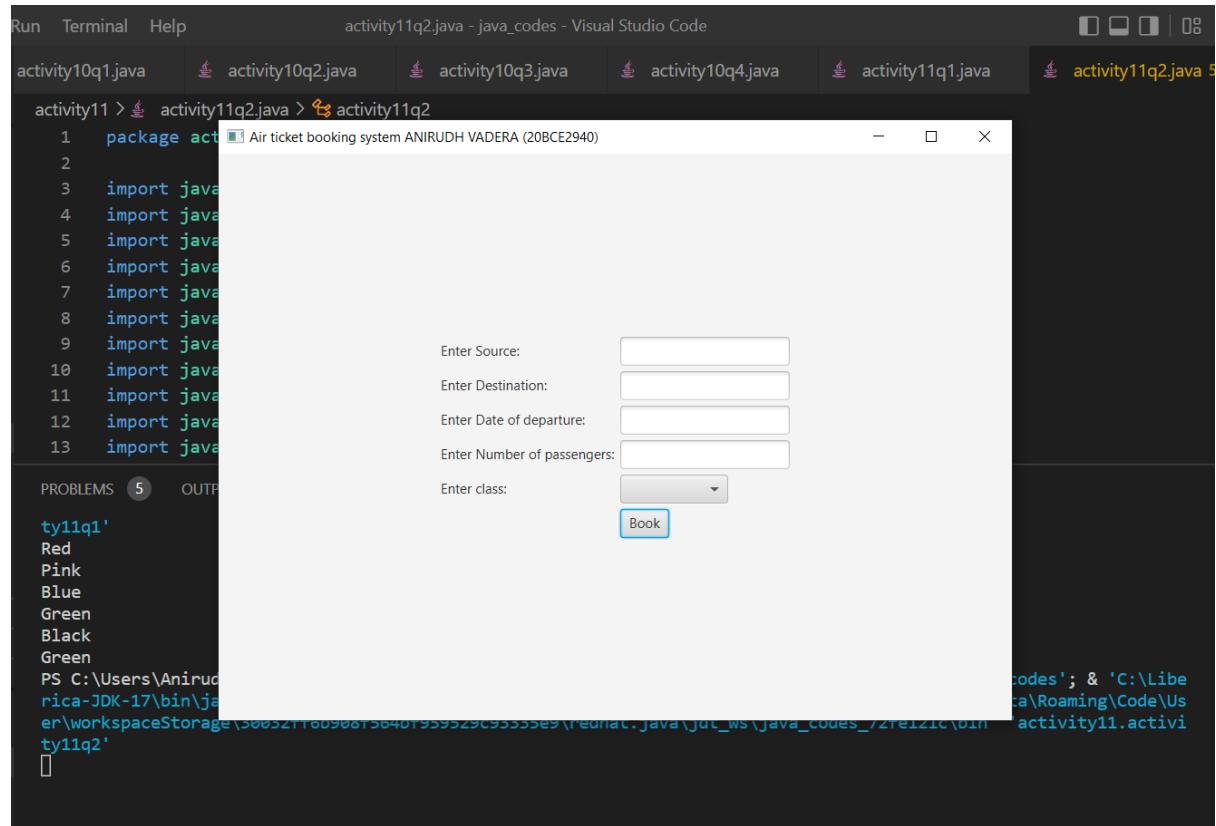
```
        s = "";
    }
    rootNode2.add(new Label("Details of Booking:"), 1, 0);
    primaryStage.setScene(myScene2);
    s = "";
}
}
});
btn2.setOnAction(new EventHandler<ActionEvent>() {
    public void handle(ActionEvent event) {
        primaryStage.setScene(myScene);
        rootNode2.getChildren().clear();
    }
});
primaryStage.setScene(myScene);
primaryStage.setTitle("Air ticket booking system ANIRUDH VADERA
(20BCE2940)");
primaryStage.setScene(myScene);
primaryStage.show();

}

public static void main(String[] args) {
    launch(args);
}
}
```

ANIRUDH VADERA
(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC

OUTPUT:



The screenshot shows a Visual Studio Code interface with multiple tabs open. The active tab is "activity11q2.java". The code in the editor is as follows:

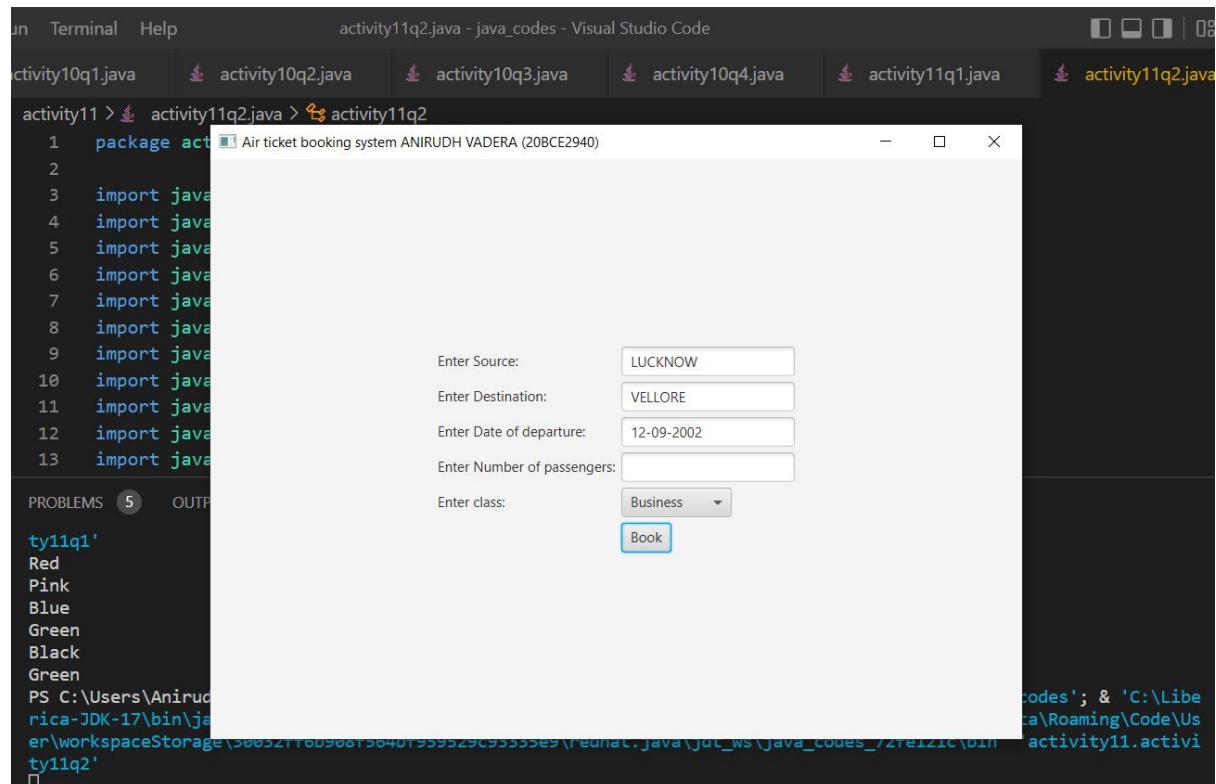
```
activity11 > activity11q2.java > activity11q2
1 package act
2
3 import java
4 import java
5 import java
6 import java
7 import java
8 import java
9 import java
10 import java
11 import java
12 import java
13 import java
```

Below the code, there are several error messages in red:

```
ty11q1'
Red
Pink
Blue
Green
Black
Green
PS C:\Users\Anirudh Vadera> cd ..\src\main\java\com\javatutorial\airticketbooking
activity11q2'
```

A modal window titled "Air ticket booking system ANIRUDH VADERA (20BCE2940)" is displayed over the code editor. It contains four input fields labeled "Enter Source:", "Enter Destination:", "Enter Date of departure:", and "Enter Number of passengers:". Below these fields is a dropdown menu labeled "Enter class:" with options "Economy", "Business", and "Premium". A blue "Book" button is located at the bottom right of the modal.

IF SOME DETAILS ARE MISSING:



The screenshot shows a Visual Studio Code interface with multiple tabs open. The active tab is "activity11q2.java". The code in the editor is identical to the previous screenshot:

```
activity11 > activity11q2.java > activity11q2
1 package act
2
3 import java
4 import java
5 import java
6 import java
7 import java
8 import java
9 import java
10 import java
11 import java
12 import java
13 import java
```

Below the code, there are several error messages in red:

```
ty11q1'
Red
Pink
Blue
Green
Black
Green
PS C:\Users\Anirudh Vadera> cd ..\src\main\java\com\javatutorial\airticketbooking
activity11q2'
```

A modal window titled "Air ticket booking system ANIRUDH VADERA (20BCE2940)" is displayed over the code editor. The "Enter Source:" field contains "LUCKNOW", the "Enter Destination:" field contains "VELLORE", the "Enter Date of departure:" field contains "12-09-2002", and the "Enter Number of passengers:" field is empty. The "Enter class:" dropdown menu is set to "Business". The "Book" button is blue.

ANIRUDH VADERA

(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER THREADING,FILES, COLLECTION, JAVAFX AND JDBC

```
activity11 > activity11q2.java > activity11q2 > main(String[])
85
86
87     });
88     btn
89
90
91
92
93     );
94     pri
95     pri
96     pri
97     pri
98     .... }
99
100

PROBLEMS 5 OUTPUT

Pink
Blue
Green
Black
Green
PS C:\Users\Anirudh\OneDrive\Desktop\java_codes> cd C:\Users\Anirudh\OneDrive\Desktop\java_codes; & 'C:\Lib
rica-JDK-17\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Anirudh\AppData\Roaming\Code\Us
er\workspaceStorage\30032ff6b908f564bf959529c93335e9\redhat.java\jdt_ws\java_codes_72fe121c\bin' 'activity11.acti
ty11q2'
```

ALL DETAILS SUCCESSFULLY FILLED:

The screenshot shows a Java application running in a terminal window within Visual Studio Code. The application is titled "Air ticket booking system ANIRUDH VADERA (20BCE2940)". The code includes several print statements displaying booking details:

```
85
86
87     });
88     btr
89
90
91
92     });
93     pri
94     pri
95     pri
96     pri
97     pri
98
99     ....}
100
```

Details of Booking:
Source: LUCKNOW
Destination: VELLORE
Date of departure: 12-09-2002
Number of passengers: 10
Class: Business

Output window content:

```
Pink
Blue
Green
Black
Green
PS C:\Users\Anirudh\OneDrive\Desktop\Java_codes> C:\ca c:\Users\Anirudh\OneDrive\Desktop\Java_codes'; & 'C:\Librica-JDK-17\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\Anirudh\AppData\Roaming\Code\Usr\workspaceStorage\300832ff6b908f564bf959529c93335e9\redhat.java\jdt_ws\java_codes_72fe121c\bin' 'activity11.activity11q2'
```

ACTIVITY 12:(JDBC)

QUESTION 1:

Develop an Air ticket booking Home page by using Text and Button objects (Source, Destination, Departure Date, No. of passengers, Class(Business, First, Economy)with appropriate Layout to describe about the booking.

For the above application use JDBC connection and store the data entered using JavaFX into the Database. Upon clicking the button “Book”, display as “Data submitted successfully”|

CODE:

```
package javaapplication1;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;
import java.sql.Statement;

import javafx.application.Application;
import javafx.event.ActionEvent;
import javafx.event.EventHandler;
import javafx.geometry.HPos;
import javafx.geometry.Pos;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.scene.control.ComboBox;
import javafx.scene.control.Label;
import javafx.scene.control.TextField;
import javafx.scene.layout.GridPane;
import javafx.stage.Stage;
import javafx.scene.text.Text;
import javafx.scene.control.DatePicker;

public class activity11q3 extends Application {

    public void start(Stage primaryStage)
        throws Exception, ClassNotFoundException, InstantiationException,
IllegalAccessException {
        GridPane rootNode = new GridPane();
        rootNode.setHgap(5);
        rootNode.setVgap(5);
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
rootNode.setAlignment(Pos.CENTER);

Scene myScene = new Scene(rootNode, 700, 500);
rootNode.add(new Label("Enter Name:"), 0, 0);
TextField name = new TextField();
rootNode.add(name, 1, 0);
rootNode.add(new Label("Enter phone number:"), 0, 1);
TextField phno = new TextField();
rootNode.add(phno, 1, 1);
rootNode.add(new Label("Enter Source:"), 0, 2);
TextField source = new TextField();
rootNode.add(source, 1, 2);
rootNode.add(new Label("Enter Destination:"), 0, 3);
TextField dest = new TextField();
rootNode.add(dest, 1, 3);
rootNode.add(new Label("Enter Date of departure:"), 0, 4);
DatePicker dod = new DatePicker();
rootNode.add(dod, 1, 4);
rootNode.add(new Label("Enter Number of passengers:"), 0, 5);
TextField nop = new TextField();
rootNode.add(nop, 1, 5);
rootNode.add(new Label("Enter class: "), 0, 6);
ComboBox comboBox = new ComboBox();
comboBox.getItems().add("Business");
comboBox.getItems().add("First");
comboBox.getItems().add("Economy");
rootNode.add(comboBox, 1, 6);
Button btn = new Button("Book");
rootNode.add(btn, 1, 7);
GridPane.setAlignment(btn, HPos.LEFT);

GridPane rootNode2 = new GridPane();
Scene myScene2 = new Scene(rootNode2, 700, 500);
rootNode2.setHgap(5);
rootNode2.setVgap(5);
rootNode2.setAlignment(Pos.CENTER);
Button btn2 = new Button("Book more tickets");

btn.setOnAction(new EventHandler<ActionEvent>() {
    public void handle(ActionEvent event) {
        String s = "";
        int flag = 2;
        try {
            String names = name.getText();
            String phone = phno.getText();
            String src = source.getText();
            String dest1 = dest.getText();
            String date = dod.getValue().toString();
            String classs = comboBox.getValue().toString();
            String no = nop.getText();
            if (names.equals("") || phone.equals("") || src.equals("") || dest1.equals("") || date.equals("") || classs.equals("") || no.equals("")) {
                flag = 1;
            }
            if (flag == 2) {
                Alert alert = new Alert(AlertType.INFORMATION);
                alert.setTitle("Information Message");
                alert.setHeaderText(null);
                alert.setContentText("Your ticket has been booked successfully!");
                alert.showAndWait();
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
});
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
String dod1 = dod.getValue().toString();
String nop1 = nop.getText();
String cls1 = (String) comboBox.getValue();

if (names.equals("") || phone.equals("") || src.equals(""))
|| dest1.equals("") || dod1.equals(""))
|| nop1.equals("")
|| cls1.equals("")){
    s = "Please fill all the fields";
    s += "\nName: " + names + "\nPhone number: " + phone +
"\nSource: " + src + "\nDestination: "
    + dest1 + "\nDate of departure: " + dod1
    + "\nNumber of passengers: " + nop1 +
"\nClass: " + cls1;
} else {
    s = "Name: " + names + "\nPhone number: " + phone +
"\nSource: " + src + "\nDestination: "
    + dest1 + "\nDate of departure: " + dod1
    + "\nNumber of passengers: " + nop1 +
"\nClass: " + cls1;
}

try {

    Connection con = null;
    Class.forName("com.mysql.cj.jdbc.Driver").newInstance();
    con =
    DriverManager.getConnection("jdbc:mysql://localhost:3307/javacodes", "root",
                                "");
    System.out.println("Connection Successful");
    Statement sqlStatements = con.createStatement();
    String querryString = "INSERT INTO
airticketbooking VALUES('"
    + src + "','" + dest1 + "','" + dod1 +
"','" + nop1 + "','" + cls1 + "','" + names
    + "','" + phone + "');";
    System.out.println(querryString);
    sqlStatements.executeUpdate(querryString);
    System.out.println("Record inserted");
    sqlStatements.close();
    con.close();
    flag = 0;

} catch (SQLException e) {
    System.out.println(e.getMessage());
    flag = 1;
}
}
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

```
Text t = new Text(s);

rootNode2.add(t, 1, 1);
rootNode2.add(btn2, 1, 2);

} catch (Exception e) {
    s = "";
}
Label heading;
if (flag == 0) {
    heading = new Label("Booking Successful");
} else {
    heading = new Label("Booking Unsuccessful");
}

heading.setStyle("-fx-font-size: 20px;");
rootNode2.add(heading, 1, 0);
primaryStage.setScene(myScene2);
s = "";

})
});
btn2.setOnAction(new EventHandler<ActionEvent>() {
@Override
public void handle(ActionEvent event) {
    primaryStage.setScene(myScene);
    rootNode2.getChildren().clear();
}
});
primaryStage.setScene(myScene);
primaryStage.setTitle("Air ticket booking system ANIRUDH VADERA
(20BCE2940)");
primaryStage.setScene(myScene);
primaryStage.show();

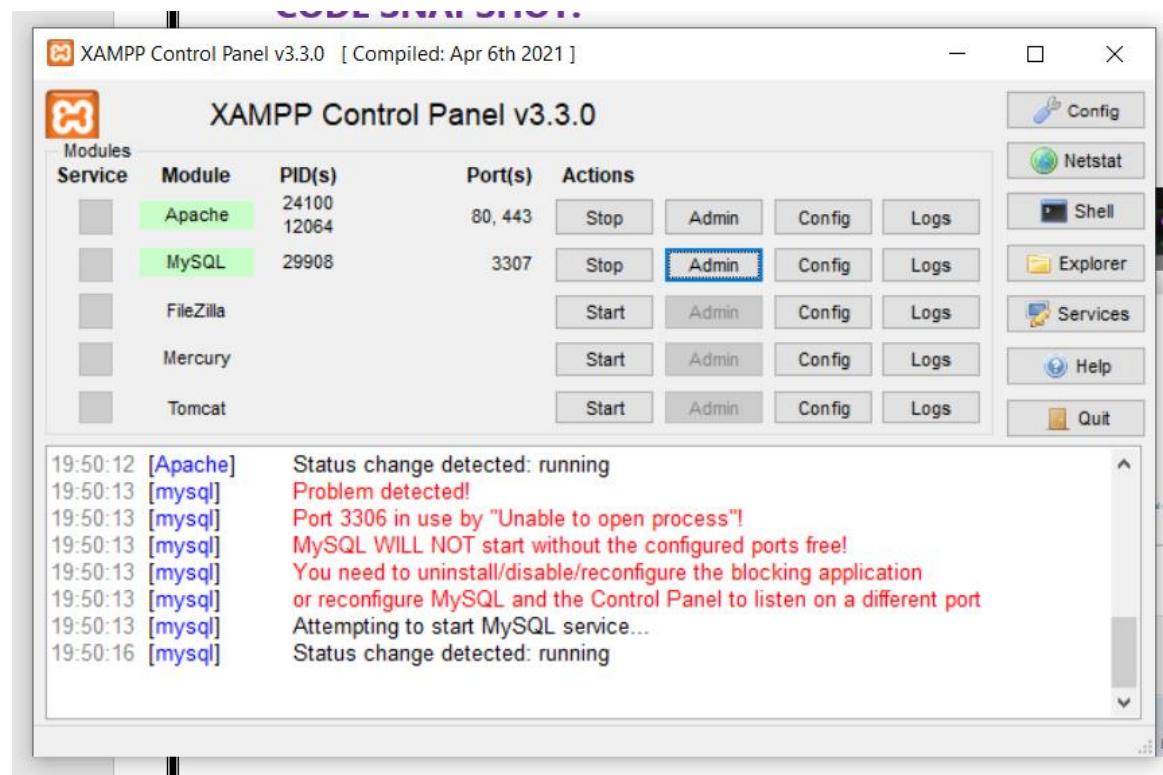
}

public static void main(String[] args) {
    launch(args);
}
}
```

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES, COLLECTION, JAVAFX AND JDBC**

OUTPUT:

Php my admin table Structure:



The screenshot shows the phpMyAdmin interface for the 'javacodes' database. On the left, the database structure is shown with tables like 'information_schema', 'javacodes', 'mysql', 'performance_schema', 'phpmyadmin', and 'test'. The 'airticketbooking' table is selected. The 'Structure' tab is active, displaying the table's columns:

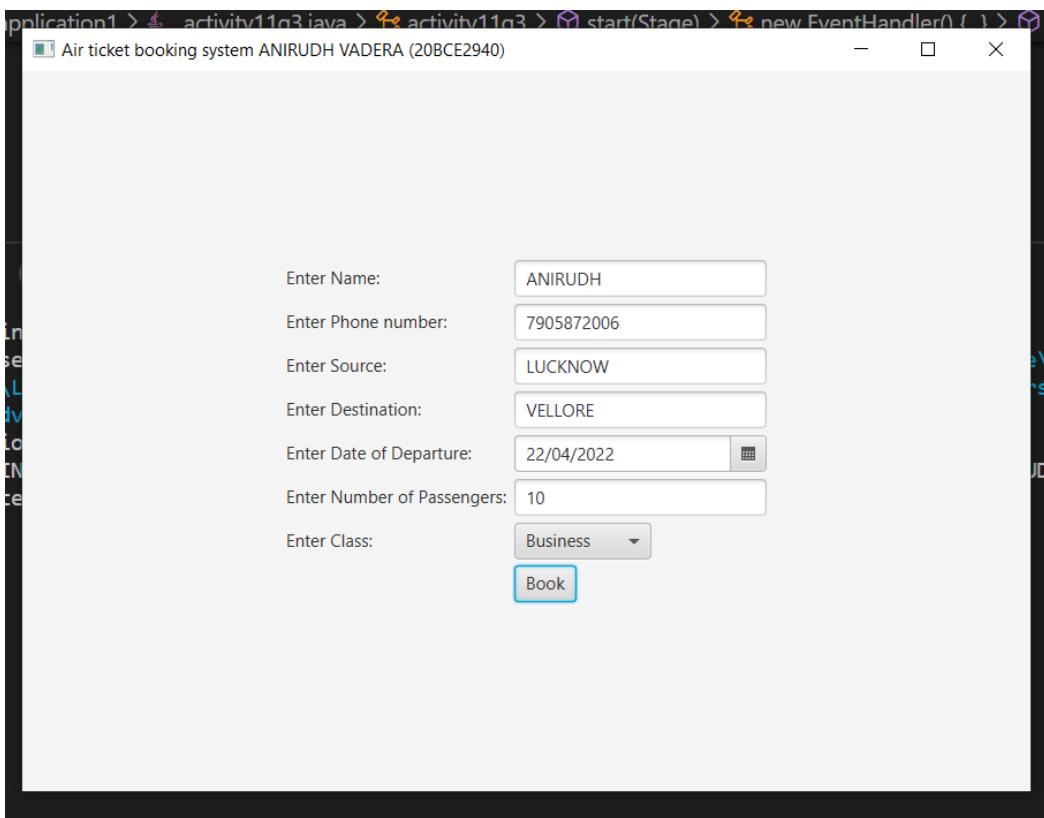
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	Source	varchar(255)	utf8mb4_general_ci		No	None			More
2	Destination	varchar(255)	utf8mb4_general_ci		No	None			More
3	DepartureDate	date			No	None			More
4	NPassengers	int(20)			No	None			More
5	Class	varchar(255)	utf8mb4_general_ci		No	None			More
6	Name	varchar(255)	utf8mb4_general_ci		No	None			More
7	PhoneNo	bigint(20)			No	None			More

Below the table structure, the 'Indexes' section shows one index:

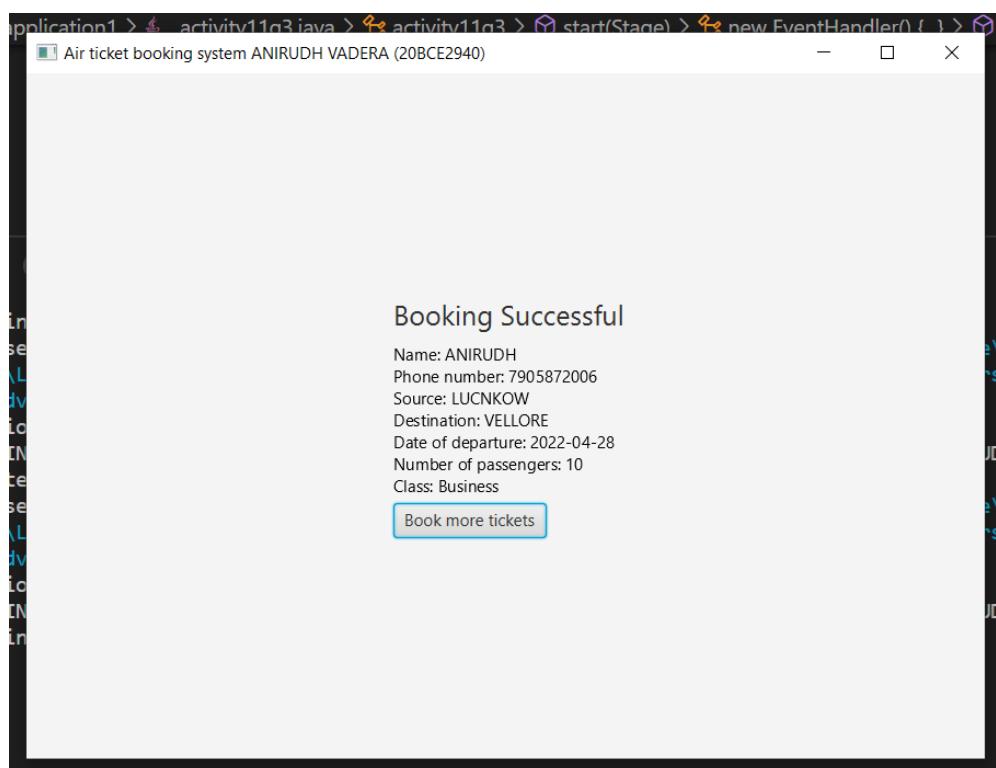
Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
	PRIMARY	BTREE	Yes	No	Name	0	A	No	
					PhoneNo	0	A	No	

ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC**

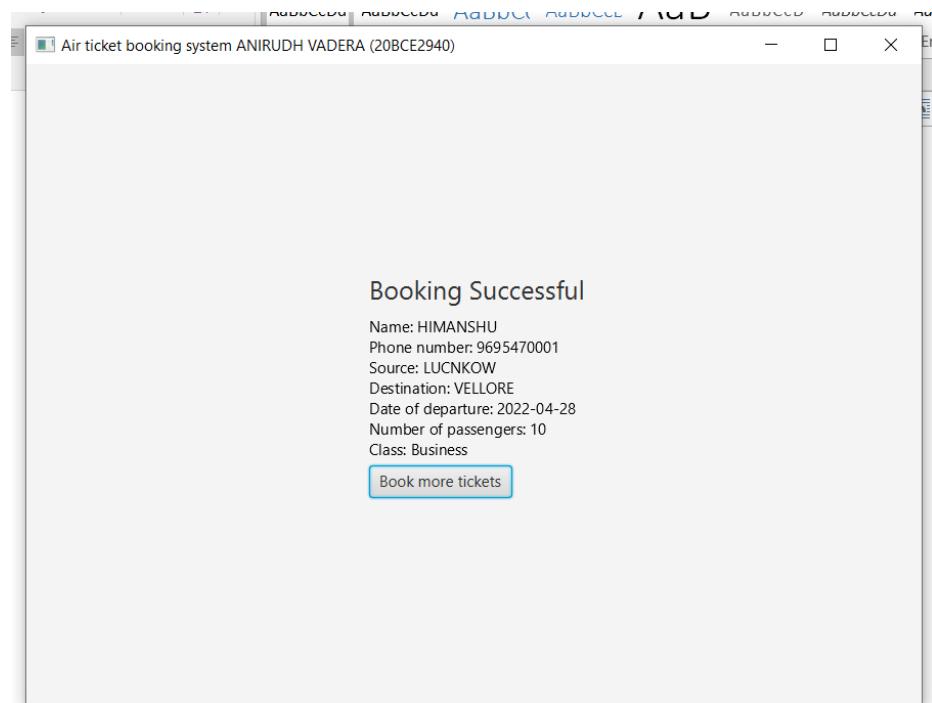
Booking the tickets:



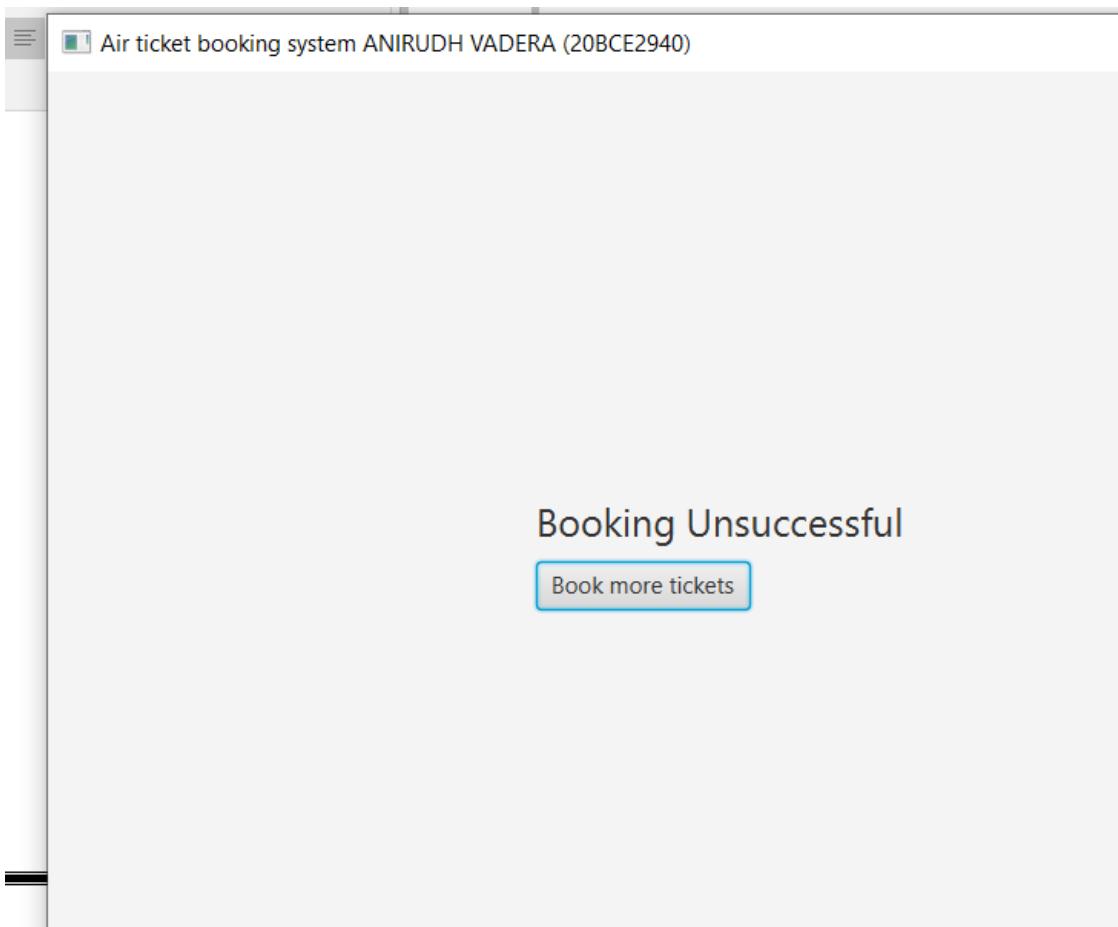
SUCCESSFUL:



ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC**



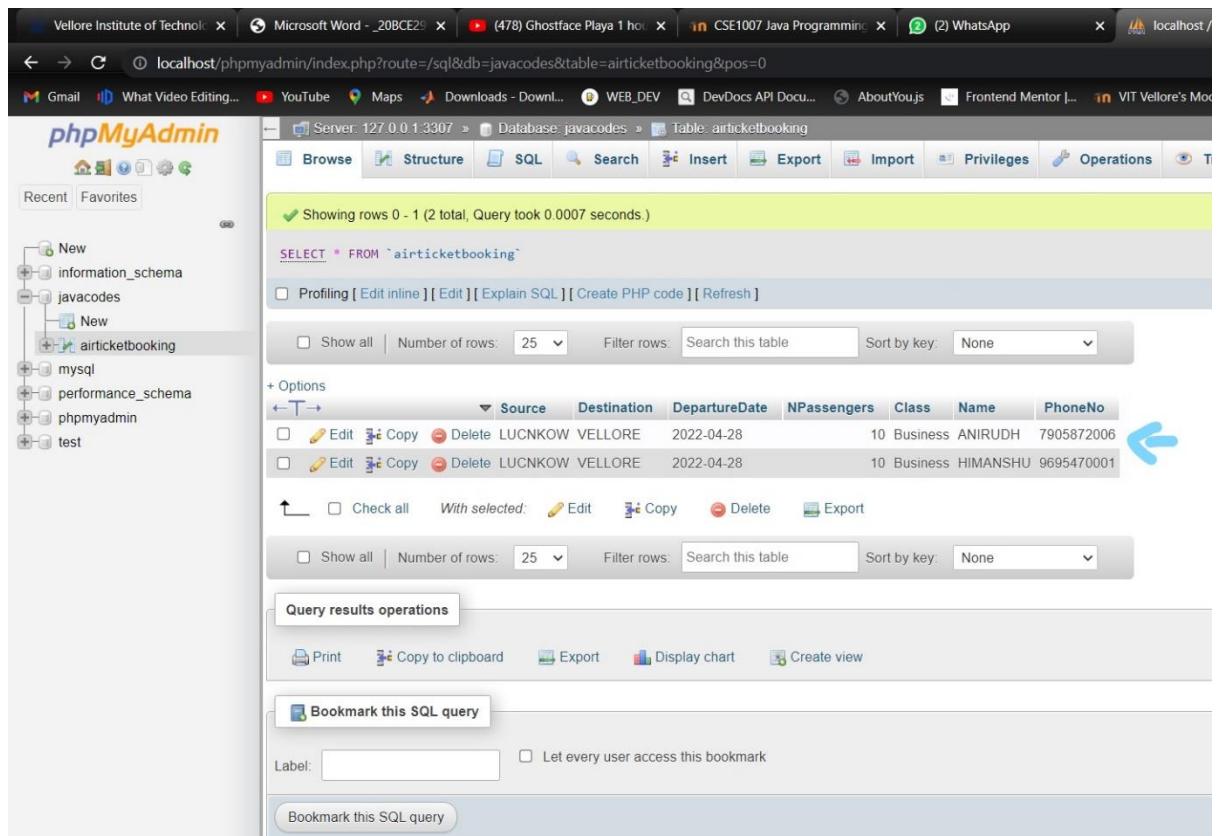
UNSUCCESFULL:



ANIRUDH VADERA
**(DIGITAL ASSIGNMENT - 5) SYNCHRONIZATION,INTER
THREADING,FILES,COLLECTION, JAVAFX AND JDBC**

After adding the tickets: the table is as follows:

```
PS C:\Users\Anirudh\OneDrive\Desktop\JavaApplication1> c:; cd 'c:\Users\Anirudh\OneDrive\Desktop\JavaApplication1'; & 'C:\Liberica-JDK-17\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '@C:\Users\Anirudh\AppData\Local\Temp\cp_9dvguywwvf80ip4dnil173wl.argfile' 'javaapplication1.activity11q3'
Connection Successful
INSERT INTO airticketbooking VALUES('LUCNKOW','VELLORE','2022-04-28',10,'Business','ANIRUDH',7905872006);
Record inserted
Connection Successful
INSERT INTO airticketbooking VALUES('LUCNKOW','VELLORE','2022-04-28',10,'Business','HIMANSHU',9695470001);
Record inserted
```



The screenshot shows the phpMyAdmin interface connected to a MySQL database named 'javacodes'. The 'airticketbooking' table is selected. The table structure is as follows:

Source	Destination	DepartureDate	NPassengers	Class	Name	PhoneNo
LUCNKOW	VELLORE	2022-04-28	10	Business	ANIRUDH	7905872006
LUCNKOW	VELLORE	2022-04-28	10	Business	HIMANSHU	9695470001

Below the table, there are buttons for 'Edit', 'Copy', 'Delete', 'Export', and 'Display chart'. There is also a 'Bookmark this SQL query' section with a 'Label' input field and a 'Let every user access this bookmark' checkbox.