CSE1007
Java Programming
Lab Assignment-4

Name: Yogeswari Sahu

Registration Number: 18BCE0928

Slot: L29+L30

Faculty: KUMAR P.J

Question :

Write a program(s) in Java that illustrates the following concepts. The concepts can be applied to any practical scenario such as student administration system or software which maintains various information about students and provides various services to the users. The other scenarios may include retail business management software, banking systems, railway reservation system, online shopping applications, environmental monitoring system etc.

- File Handling
- Collections
  - List
  - Map
  - Set

Solution:

PROGRAM 1:

In a situation where a patient needs plasma for his COVID-19 treatment of Blood group "A+ve" but the blood of this blood group is not available So we need to search from the users of who hasn't donated blood in the last 6 months of the given blood group. Write a program to solve the issue by Defining a class 'Donor' to store the basic details like: Name, age, Address,

Contact number, blood group, date of last donation.Write these objects to a file. Read these objects from the file and display only those donors' details whose blood group are 'At've' and had not donated for the recent six months.

Concepts used:

- File Handling
- Collections
  - List

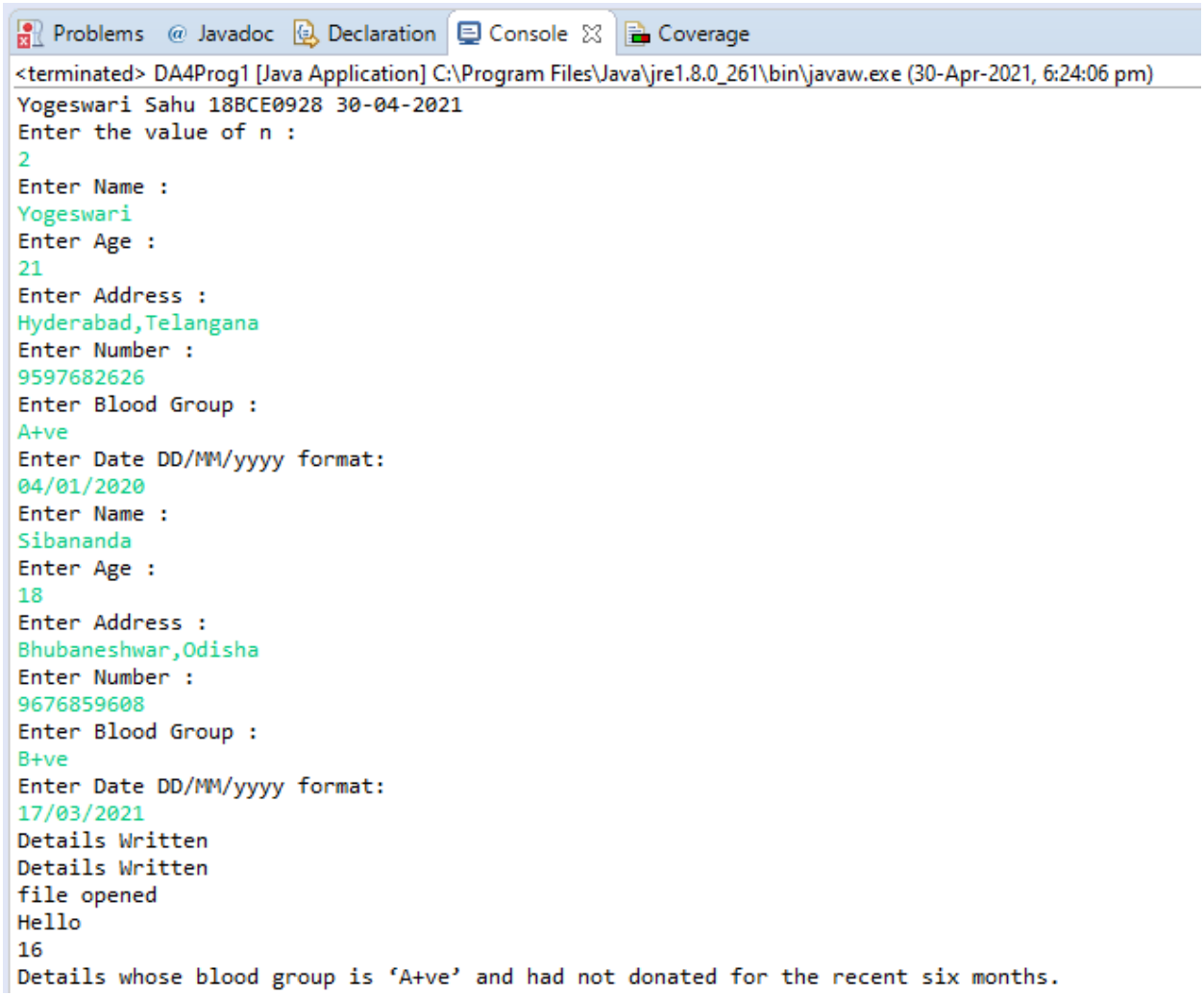
Code:

```
package JavaDA4;

import java.io.*;
import java.text.*;
import java.util.*;
//Defining a class Donor with data types
class Donor{
	public Donor(String name, String age, String address, String number,
String bg, String date) {
		super();
		this.name = name;
		this.age = age;
		this.address = address;
		this.number = number;
		this.bg = bg;
		this.date = date;
	}
	public String name;
	public String age;
	public String address;
	public String number;
	public String bg;
	public String date;
}

public class DA4Prog1 {
	public static void main(String[] args) throws IOException, ParseException
{
		System.out.println("Yogeswari Sahu 18BCE0928 30-04-2021");
		//Taking Input
		Scanner sc = new Scanner(System.in);
		System.out.println("Enter the value of n : ");
		int n = sc.nextInt();
		//Initializing an ArrayList
```

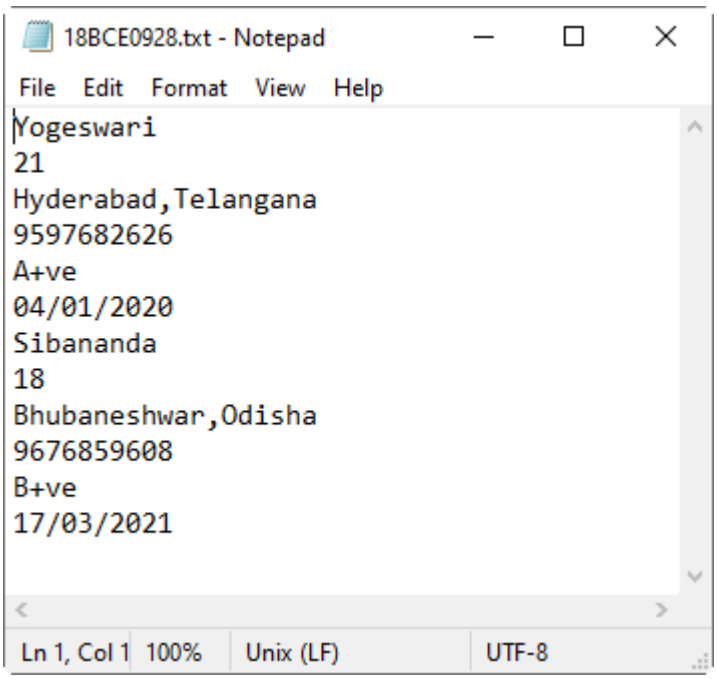
```
            ArrayList<Donor> arr=new ArrayList<Donor>();

            String filename = "18BCE0928.txt";
            String a ;

            for (int i = 0; i < n; i++) {
                  System.out.println("Enter Name : ");
                  String name = sc.next();
                  System.out.println("Enter Age : ");
                  String age = sc.next();
                  System.out.println("Enter Address : ");
                  String address = sc.next();
                  System.out.println("Enter Number : ");
                  String number = sc.next();
                  System.out.println("Enter Blood Group : ");
                  String bg = sc.next();
                  System.out.println("Enter Date DD/MM/yyyy format: ");
                  String date = sc.next();
                  arr.add(new Donor(name,age,address,number,bg,date)) ;
            }
            BufferedWriter bf = new BufferedWriter(new
FileWriter("18BCE0928.txt"));
            for (int i = 0; i < n; i++) {
                  bf.write(arr.get(i).name + '\n');
                  bf.write(arr.get(i).age + '\n');
                  bf.write(arr.get(i).address + '\n');
                  bf.write(arr.get(i).number + '\n');
                  bf.write(arr.get(i).bg + '\n');
                  bf.write(arr.get(i).date + '\n');
                  System.out.println("Details Written");
            }
            bf.close();

            BufferedReader br = new BufferedReader(new FileReader(filename));
            System.out.println("file opened");
            try {
                  while ((a = br.readLine()) != null) {
                        String[] data = new String[6];
                        for (int i = 0; i < 6; i++) {
                              data[i] = a;
                              a = br.readLine();
                        }
                        String dat = data[5];
                        Date d = new SimpleDateFormat("dd/MM/yyyy").parse(dat);
                        Date now = new Date();
                        long difference = (now.getTime() - d.getTime()) /
86400000;
                        long month = difference / 30;
                        System.out.println(month);
                        if (data[4].equals("A+ve") && Math.abs(month) >= 6) {
```

```
                              System.out.println("Details whose blood group is
'A+ve' and had not donated for the recent six months. ");
                              System.out.println("Name:" + data[0]);
                              System.out.println("Age:" + data[1]);
                              System.out.println("Address:" + data[2]);
                              System.out.println("Number:" + data[3]);
                              System.out.println("Blood Group:" + data[4]);
                              System.out.println("Date:" + data[5]);
                        }
                  }
                  br.close();
            } catch (FileNotFoundException e)
            {
                  System.err.println("Oops! Please check for the presence of
file in the path specified.");
            }
      }
}
```

Output:



```
Details whose blood group is 'A+ve' and had not donated for the recent six months.
Name:Yogeswari
Age:21
Address:Hyderabad,Telangana
Number:9597682626
Blood Group:A+ve
Date:04/01/2020
```



PROGRAM 2:

Assuming an organization is maintaining objects of the salary of their employees, Write a program to demonstrate the object serialization and deserialization when one object holds information about the salary the employees of an organization.

Concepts used:

- File Handling(Serialization and Deserialization)

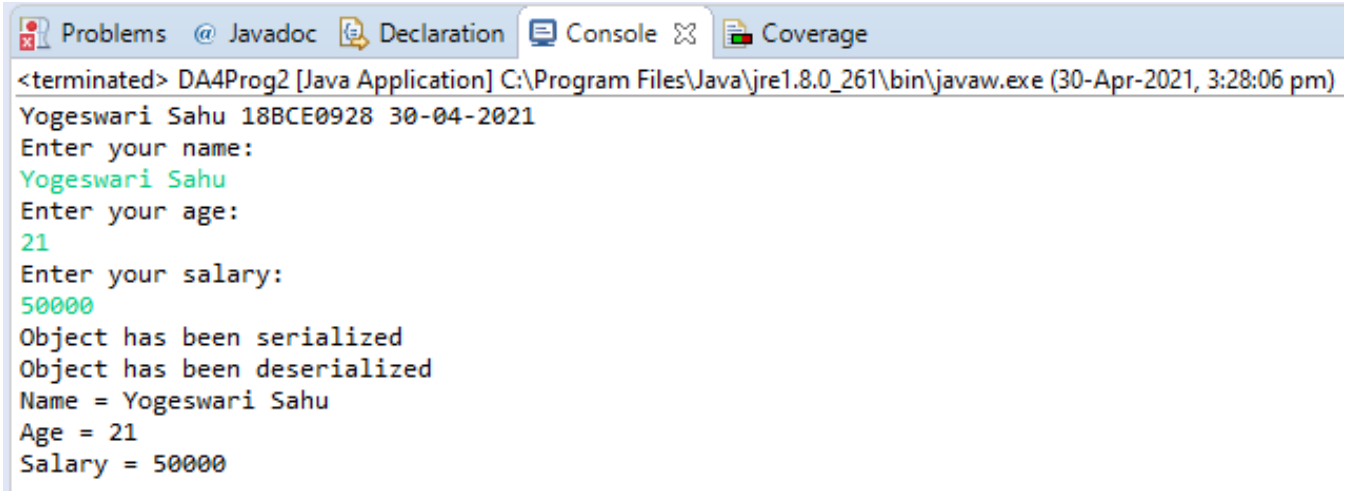
Code:

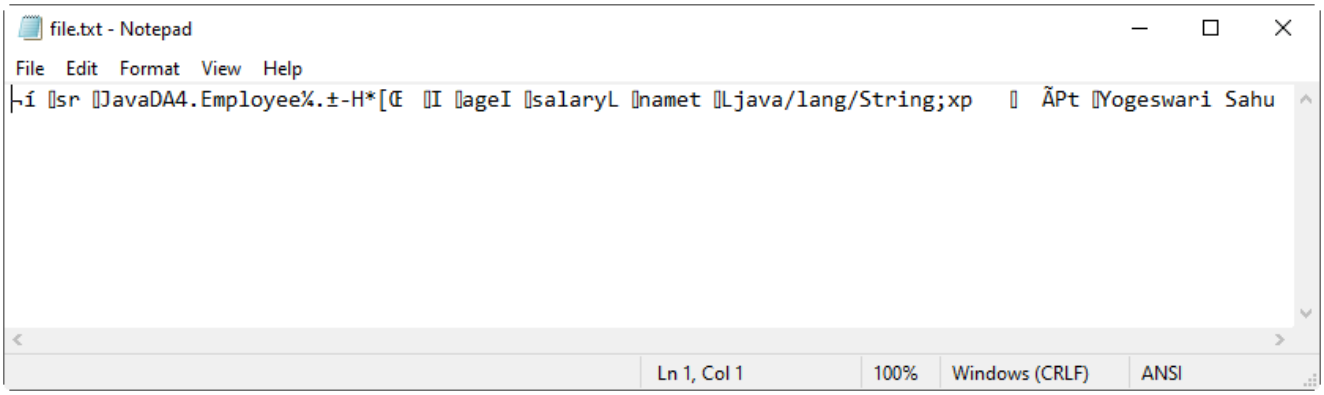
```
package JavaDA4;

import java.io.*;
import java.util.*;
```

```
class Employee implements Serializable
{
	String name;
	int age;
	int salary;
	// Default constructor
	public Employee(String name, int age, int salary)
	{
		this.name = name;
		this.age = age;
		this.salary = salary;
	}
}

public class DA4Prog2
{
	public static void main(String[] args)
	{
		System.out.println("Yogeswari Sahu 18BCE0928 30-04-2021");
		//Taking Input
		Scanner sc=new Scanner(System.in);
		System.out.println("Enter your name:");
		String name =sc.nextLine();
		System.out.println("Enter your age:");
		int age=sc.nextInt();
		System.out.println("Enter your salary:");
		int salary=sc.nextInt();
		Employee object = new Employee(name, age, salary);
		String filename = "file.txt";
		// Serialization
		try
		{
			FileOutputStream file = new FileOutputStream(filename);
			ObjectOutputStream out = new ObjectOutputStream(file);
			out.writeObject(object);
			out.close();
			file.close();
			System.out.println("Object has been serialized");
		}
		catch(IOException ex)
		{
			System.out.println("IOException is caught");
		}
		Employee object1 = null;
		// Deserialization
		try
		{
			FileInputStream file = new FileInputStream(filename);
			ObjectInputStream in = new ObjectInputStream(file);
```

```
                object1 = (Employee)in.readObject();
                in.close();
                file.close();
                System.out.println("Object has been deserialized ");
                System.out.println("Name = " + object1.name);
                System.out.println("Age = " + object1.age);
                System.out.println("Salary = " + object1.salary);
            }
            catch(IOException ex)
            {
                System.out.println("IOException is caught");
            }
            catch(ClassNotFoundException ex)
            {
                System.out.println("ClassNotFoundException is caught");
            }
        }
}
```

Output:





PROGRAM 3:

In a university, let's take the scenario of course allocation of an university. Make appropriate data structures like Hash Maps, Lists ,Sets of Collection Framework for storing the data of Subjects registered by different candidates and Employee ids of the faculties taking up the subjects. Make a system for performing the following actions: registering subjects for a student , deleting the subjects registered by the student , mapping the faculties of any student via their employee IDS.

Concepts used:

- Collections
  - List
  - Map
  - Set

Code:

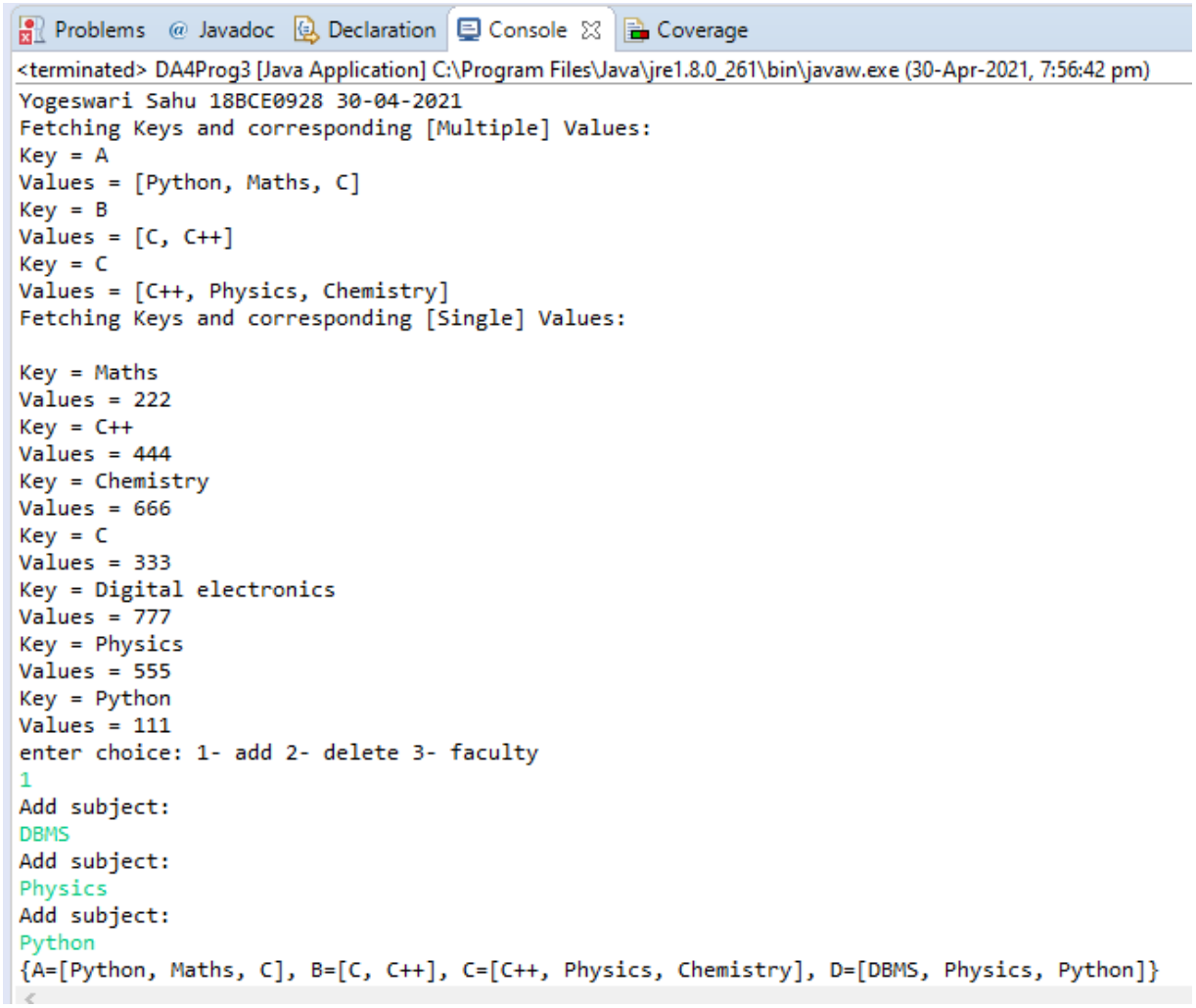
```
package JavaDA4;

import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;
import java.util.Scanner;
import java.util.Set;

public class DA4Prog3 {
	//Initializing Hashmaps
	static Map<String, List<String>> map1 = new HashMap<String,
List<String>>();
	static Map<String, Integer> map2 = new HashMap<String, Integer>();

	//Defining add function
	public void add_in_map1(String key, List<String> values)
	{
		map1.put(key, values);
		System.out.println(map1);
	}

	//Defining delete function
```

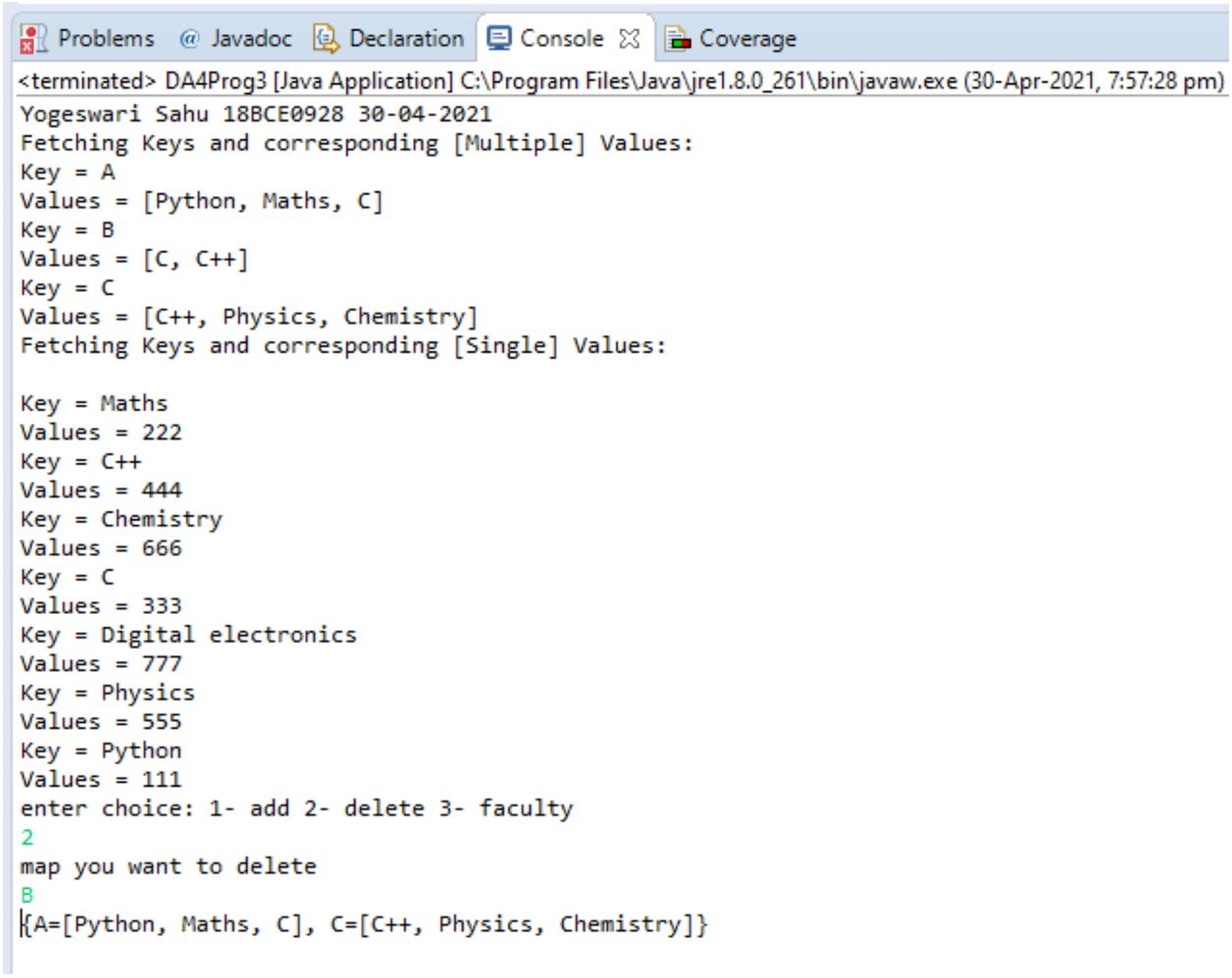
```
public void delete_in_map1(String key) {
	map1.remove(key);
	System.out.println(map1);
}

//Defining findFaculties function
public void findFaculties(String key) {
	if (map1.containsKey(key)) {
		List<Integer> faculties = new ArrayList<Integer>();
		List<String> list = map1.get(key);
		Set<String> keys = map2.keySet();
		for (String l : list) {
			for (String k : keys) {
				if (l.equals(k)) {
					faculties.add(map2.get(k));
				}
			}
		}
		System.out.println(faculties);
	} else {
		System.out.println("No such student");
	}
}

public static void main(String[] args) {
	System.out.println("Yogeswari Sahu 18BCE0928 30-04-2021");
	//Initializing lists with values
	List<String> valSetOne = new ArrayList<String>();
	valSetOne.add("Python");
	valSetOne.add("Maths");
	valSetOne.add("C");
	List<String> valSetTwo = new ArrayList<String>();
	valSetTwo.add("C");
	valSetTwo.add("C++");
	List<String> valSetThree = new ArrayList<String>();
	valSetThree.add("C++");
	valSetThree.add("Physics");
	valSetThree.add("Chemistry");

	//Initializing maps with values
	map1.put("A", valSetOne);
	map1.put("B", valSetTwo);
	map1.put("C", valSetThree);
	map2.put("Python", 111);
	map2.put("Maths", 222);
	map2.put("C", 333);
	map2.put("C++", 444);
	map2.put("Physics", 555);
	map2.put("Chemistry", 666);
```

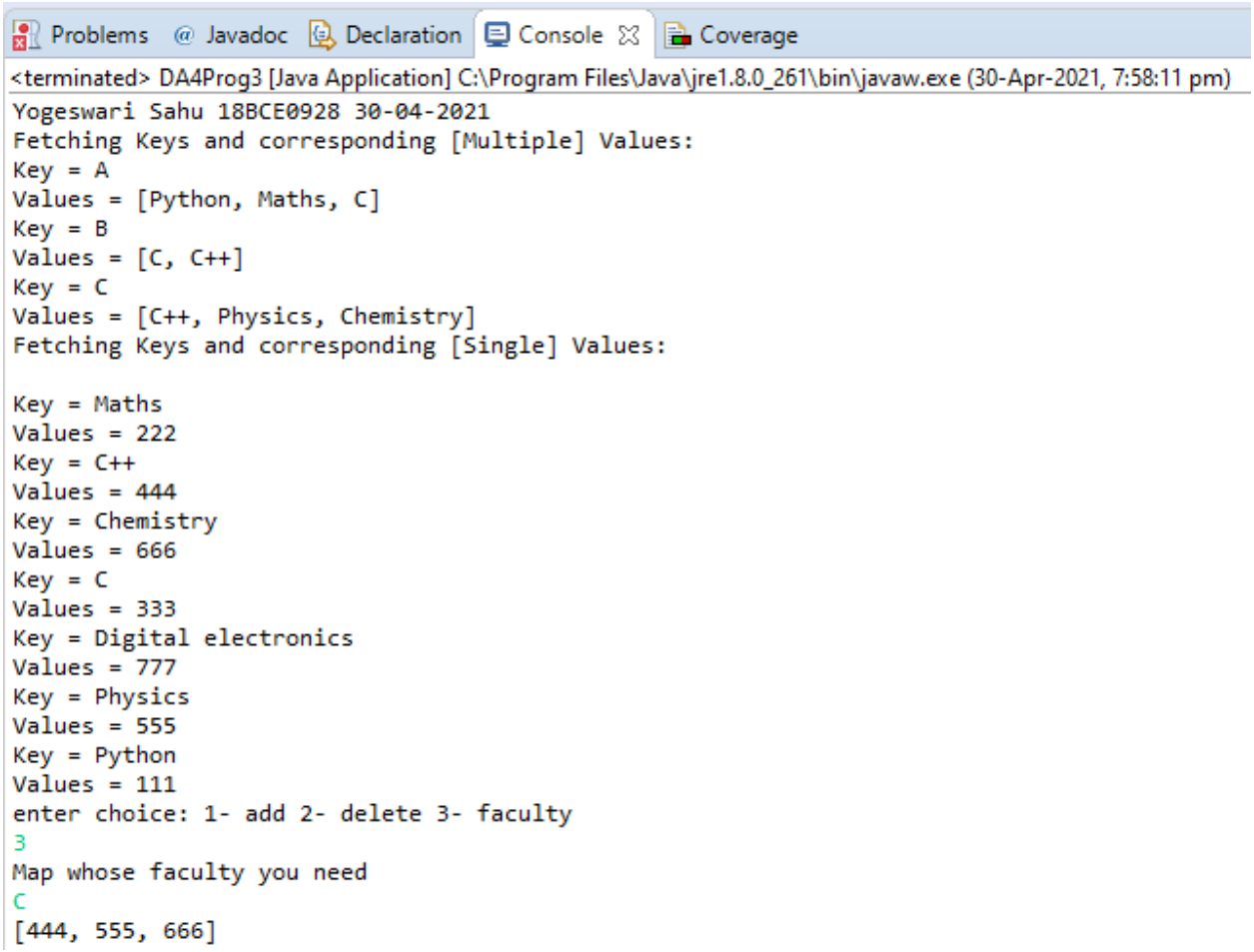
```
            map2.put("Digital electronics", 777);

            //Iterating over maps and displaying the corresponding values
            System.out.println("Fetching Keys and corresponding [Multiple]
Values:");
            for (Map.Entry<String, List<String>> entry1 : map1.entrySet()) {
                  String key1 = entry1.getKey();
                  List<String> values1 = entry1.getValue();
                  System.out.println("Key = " + key1);
                  System.out.println("Values = " + values1);
            }
            System.out.println("Fetching Keys and corresponding [Single]
Values: \n");
            for (Map.Entry<String, Integer> entry2 : map2.entrySet()) {
                  String key2 = entry2.getKey();
                  Integer values2 = entry2.getValue();
                  System.out.println("Key = " + key2);
                  System.out.println("Values = " + values2);
            }

            System.out.println("enter choice: 1- add 2- delete 3- faculty
");

            Scanner sc = new Scanner(System.in);
            int choice = sc.nextInt();
//    Case driven program using Switch based on user's choice
            switch (choice) {

                  case 1:
                        DA4Prog3 c1 = new DA4Prog3();
                        List<String> val = new ArrayList<String>();
                        System.out.println("Add subject:");
                        Scanner sc2 = new Scanner(System.in);
                        String k1 = sc2.next();
                        val.add(k1);
                        System.out.println("Add subject:");
                        String k2 = sc2.next();
                        val.add(k2);
                        System.out.println("Add subject:");
                        String k3 = sc2.next();
                        val.add(k3);
                        c1.add_in_map1("D", val);
                        break;

                  case 2:
                        DA4Prog3 c2 = new DA4Prog3();
                        System.out.println("map you want to delete");
                        Scanner sc1 = new Scanner(System.in);
                        String k = sc1.next();
```

```
                        c2.delete_in_map1(k);
                        break;

                case 3:
                        DA4Prog3 c3 = new DA4Prog3();
                        System.out.println("Map whose faculty you need");
                        Scanner sc3 = new Scanner(System.in);
                        String k5 = sc3.next();
                        c3.findFaculties(k5);
                        break;

                default:
                        System.out.println("error");

            }
        }
}
```

Output:




Problems
Javadoc
Declaration
Console
Coverage
<terminated> DA4Prog3 [Java Application] C:\Program Files\Java\jre1.8.0_261\bin\javaw.exe (30-Apr-2021, 7:57:28 pm)
Yogeswari Sahu 18BCE0928 30-04-2021
Fetching Keys and corresponding [Multiple] Values:
Key = A
Values = [Python, Maths, C]
Key = B
Values = [C, C++]
Key = C
Values = [C++, Physics, Chemistry]
Fetching Keys and corresponding [Single] Values:
Key = Maths
Values = 222
Key = C++
Values = 444
Key = Chemistry
Values = 666
Key = C
Values = 333
Key = Digital electronics
Values = 777
Key = Physics
Values = 555
Key = Python
Values = 111
enter choice: 1- add 2- delete 3- faculty
2
map you want to delete
B
{A=[Python, Maths, C], C=[C++, Physics, Chemistry]}


Problems
Javadoc
Declaration
Console
Coverage
<terminated> DA4Prog3 [Java Application] C:\Program Files\Java\jre1.8.0_261\bin\javaw.exe (30-Apr-2021, 7:58:11 pm)
Yogeswari Sahu 18BCE0928 30-04-2021
Fetching Keys and corresponding [Multiple] Values:
Key = A
Values = [Python, Maths, C]
Key = B
Values = [C, C++]
Key = C
Values = [C++, Physics, Chemistry]
Fetching Keys and corresponding [Single] Values:
Key = Maths
Values = 222
Key = C++
Values = 444
Key = Chemistry
Values = 666
Key = C
Values = 333
Key = Digital electronics
Values = 777
Key = Physics
Values = 555
Key = Python
Values = 111
enter choice: 1- add 2- delete 3- faculty
3
Map whose faculty you need
C
[444, 555, 666]