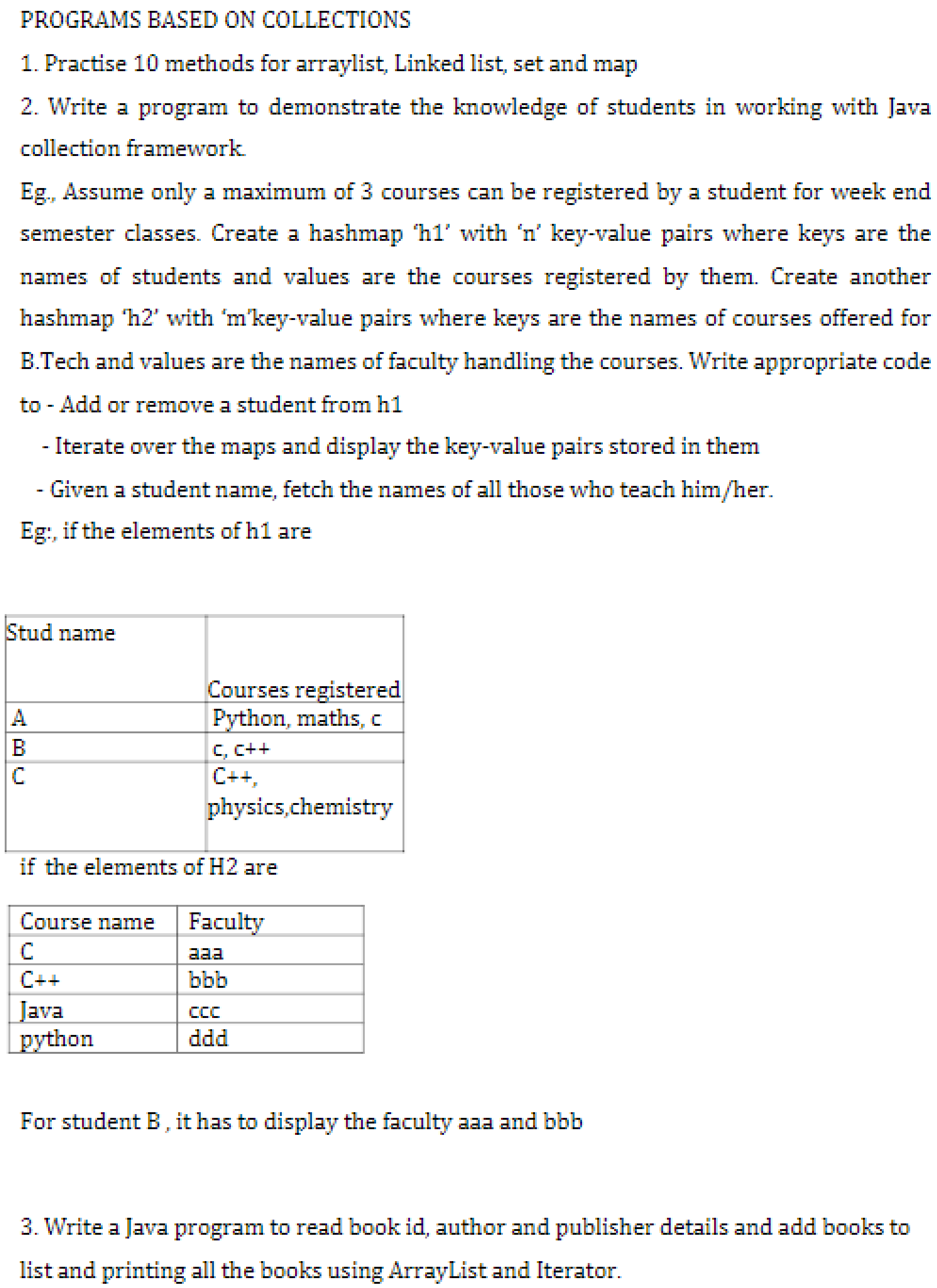
DIGITAL ASSIGNMENT 6

CSE1007 / JAVA LAB

ANISH SHRESTHA 11/24/21 20BCE2893

DIGITAL ASSIGNMENT -6

On Collection:



PROGRAMS BASED ON COLLECTIONS

1. Practise 10 methods for arraylist, Linked list, set and map

2. Write a program to demonstrate the knowledge of students in working with Java collection framework.

Eg., 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.

Eg:, if the elements of h1 are

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

if the elements of H2 are

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

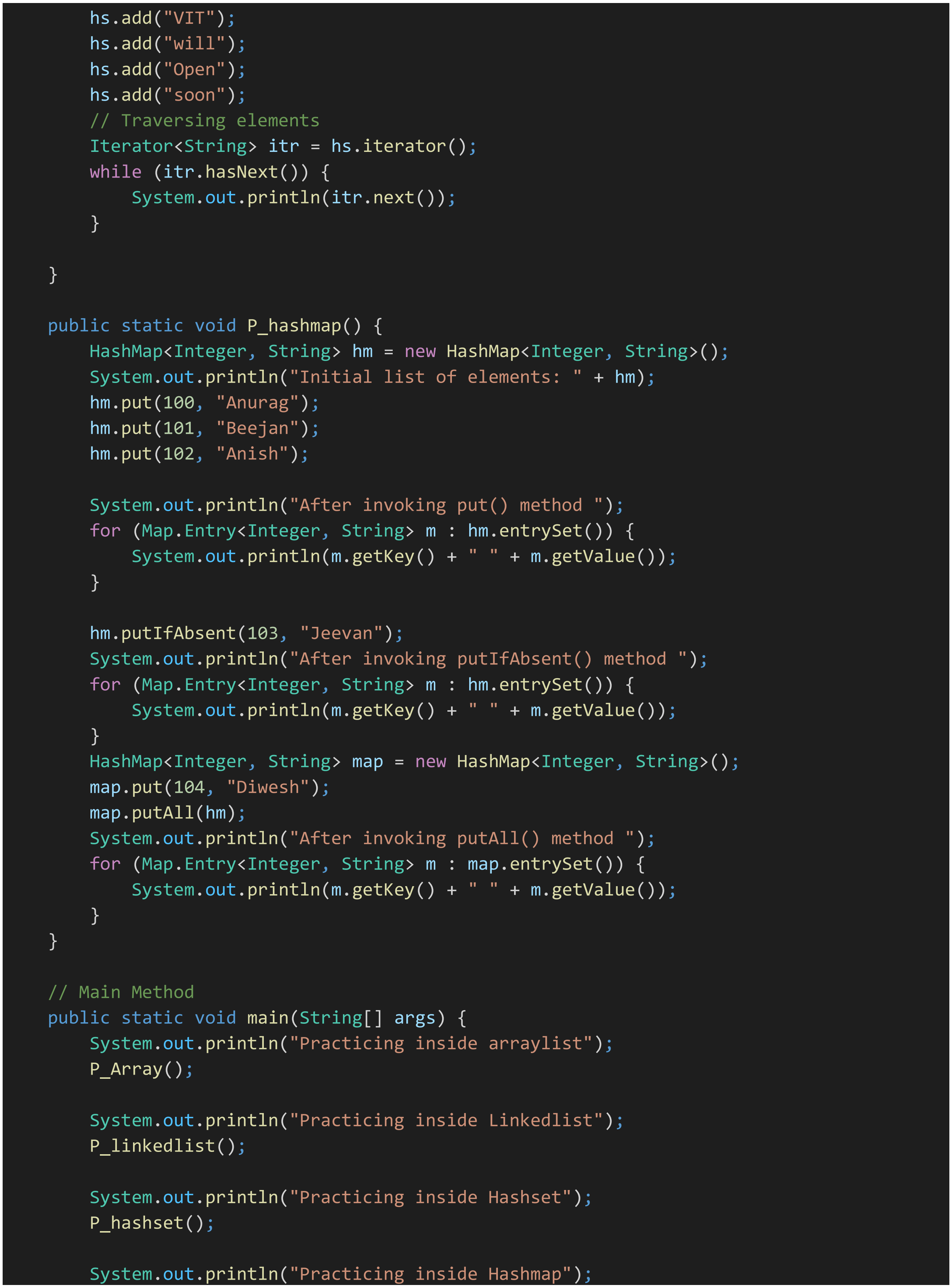
For student B , it has to display the faculty aaa and bbb

3. Write a Java program to read book id, author and publisher details and add books to list and printing all the books using ArrayList and Iterator.

1.

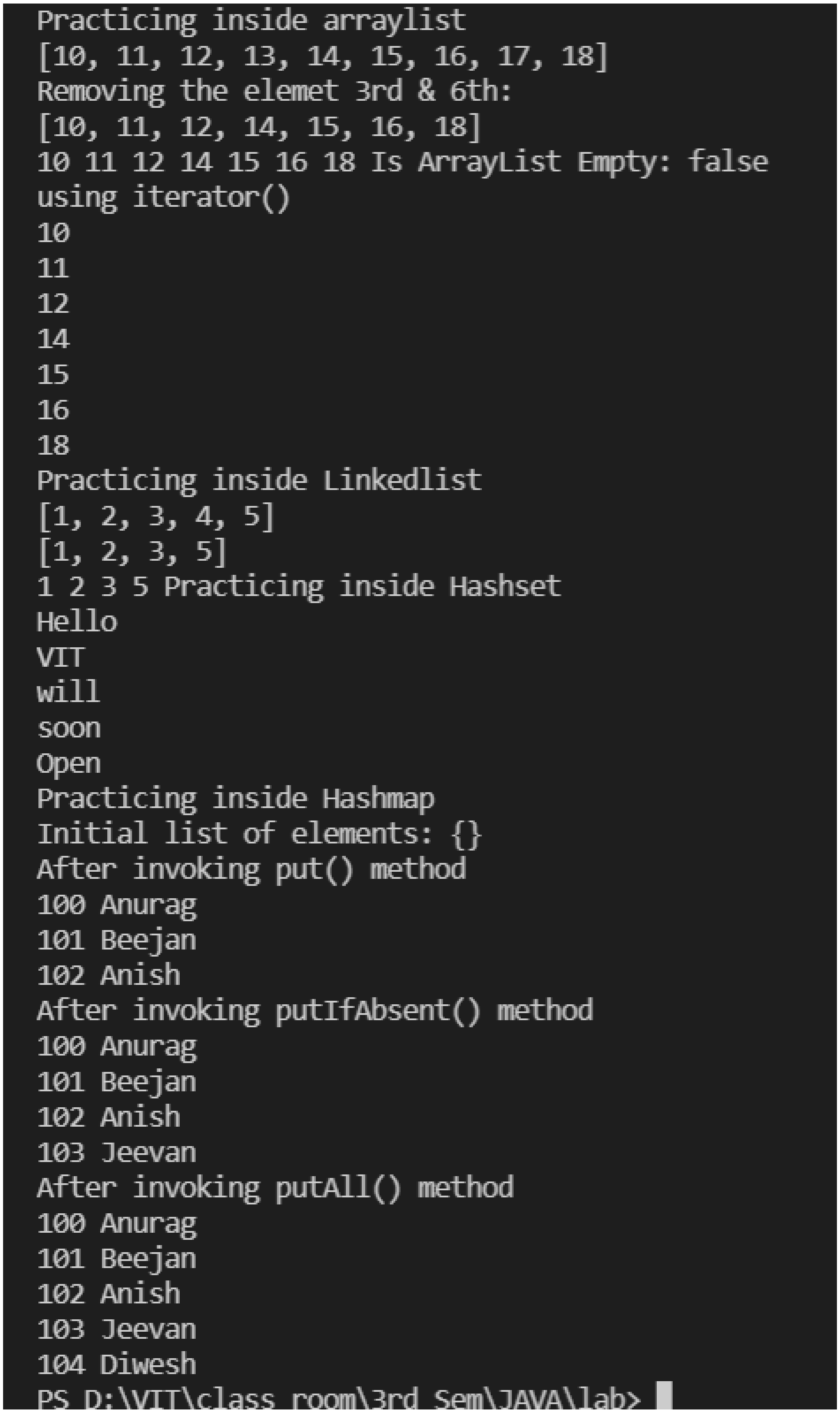
Code:

```
import java.util.*;

public class MethPrac {

    public static void P_Array() {

        ArrayList<Integer> al = new ArrayList<Integer>();
        for (int i = 10; i <= 18; i++)
            al.add(i);
        System.out.println(al);
        System.out.println("Removing the elemet 3rd & 6th:");
        al.remove(3);
        al.remove(6);
        System.out.println(al);

        // Printing elements one by one
        for (int i = 0; i < al.size(); i++)
            System.out.print(al.get(i) + " ");

        System.out.println("Is ArrayList Empty: " + al.isEmpty());
        System.out.println("using iterator()");

        Iterator<Integer> itr = al.iterator();// getting the Iterator
        while (itr.hasNext()) {// check if iterator has the elements
            System.out.println(itr.next());// printing the element and move to next
        }

    }

    public static void P_linkedlist() {
        LinkedList<Integer> ll = new LinkedList<Integer>();

        for (int i = 1; i <= 5; i++)
            ll.add(i);
        System.out.println(ll);

        ll.remove(3);
        System.out.println(ll);
        for (int i = 0; i < ll.size(); i++)
            System.out.print(ll.get(i) + " ");
    }

    public static void P_hashset() {
        HashSet<String> hs = new HashSet<String>();
        hs.add("Hello ");
```



```
        hs.add("VIT");
        hs.add("will");
        hs.add("Open");
        hs.add("soon");
        // Traversing elements
        Iterator<String> itr = hs.iterator();
        while (itr.hasNext()) {
            System.out.println(itr.next());
        }

    }

    public static void P_hashmap() {
        HashMap<Integer, String> hm = new HashMap<Integer, String>();
        System.out.println("Initial list of elements: " + hm);
        hm.put(100, "Anurag");
        hm.put(101, "Beejan");
        hm.put(102, "Anish");

        System.out.println("After invoking put() method ");
        for (Map.Entry<Integer, String> m : hm.entrySet()) {
            System.out.println(m.getKey() + " " + m.getValue());
        }

        hm.putIfAbsent(103, "Jeevan");
        System.out.println("After invoking putIfAbsent() method ");
        for (Map.Entry<Integer, String> m : hm.entrySet()) {
            System.out.println(m.getKey() + " " + m.getValue());
        }
        HashMap<Integer, String> map = new HashMap<Integer, String>();
        map.put(104, "Diwesh");
        map.putAll(hm);
        System.out.println("After invoking putAll() method ");
        for (Map.Entry<Integer, String> m : map.entrySet()) {
            System.out.println(m.getKey() + " " + m.getValue());
        }
    }

    // Main Method
    public static void main(String[] args) {
        System.out.println("Practicing inside arraylist");
        P_Array();

        System.out.println("Practicing inside Linkedlist");
        P_linkedlist();

        System.out.println("Practicing inside Hashset");
        P_hashset();

        System.out.println("Practicing inside Hashmap");
        P_hashmap();

    }
}
```

Output:

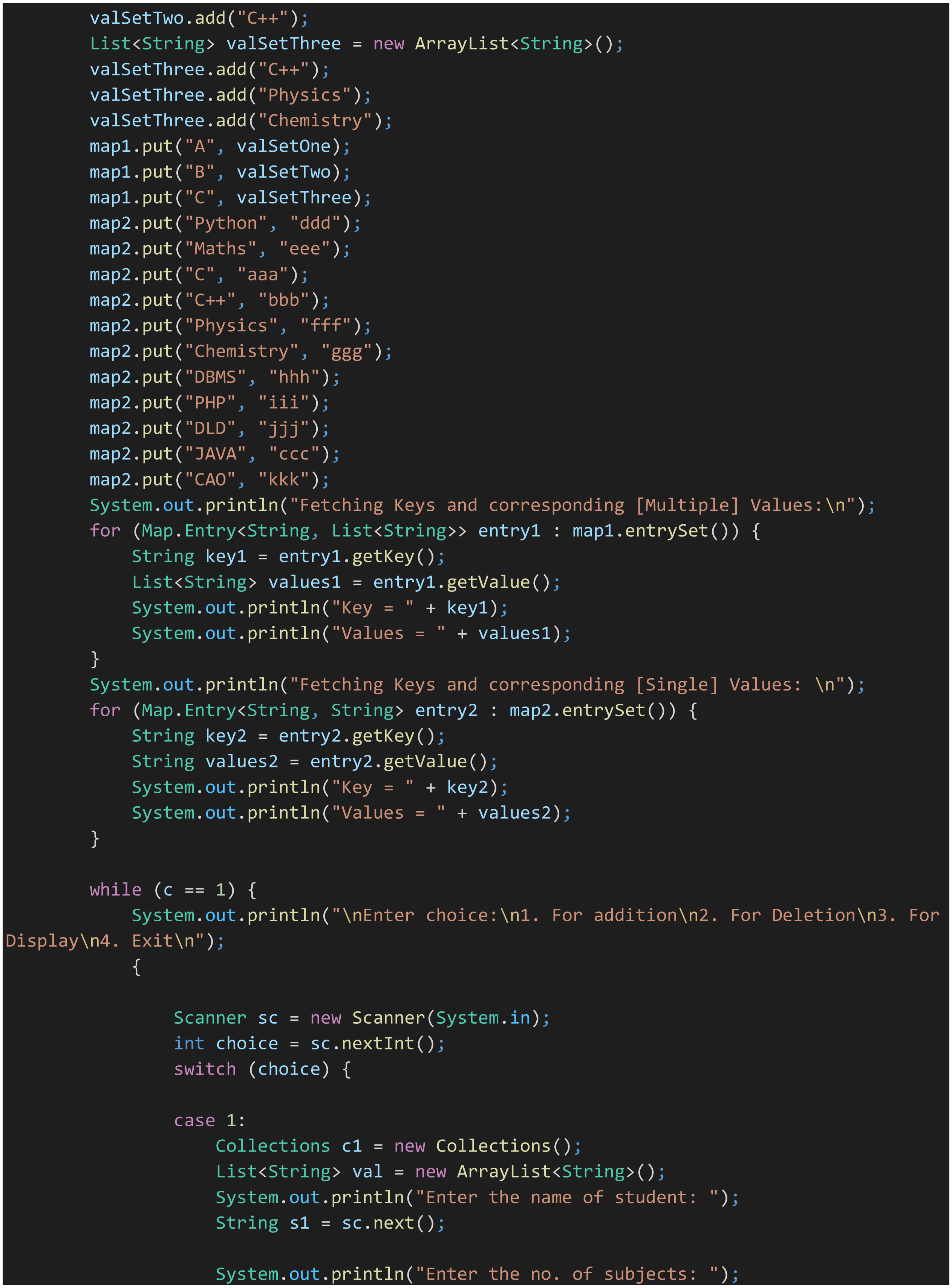


```
Practicing inside arraylist
[10, 11, 12, 13, 14, 15, 16, 17, 18]
Removing the elemet 3rd & 6th:
[10, 11, 12, 14, 15, 16, 18]
10 11 12 14 15 16 18 Is ArrayList Empty: false
using iterator()
10
11
12
14
15
16
18
Practicing inside Linkedlist
[1, 2, 3, 4, 5]
[1, 2, 3, 5]
1 2 3 5 Practicing inside Hashset
Hello
VIT
will
soon
Open
Practicing inside Hashmap
Initial list of elements: {}
After invoking put() method
100 Anurag
101 Beejan
102 Anish
After invoking putIfAbsent() method
100 Anurag
101 Beejan
102 Anish
103 Jeevan
After invoking putAll() method
100 Anurag
101 Beejan
102 Anish
103 Jeevan
104 Diwesh
PS D:\VIT\class room\3rd Sem\JAVA\lab>
```

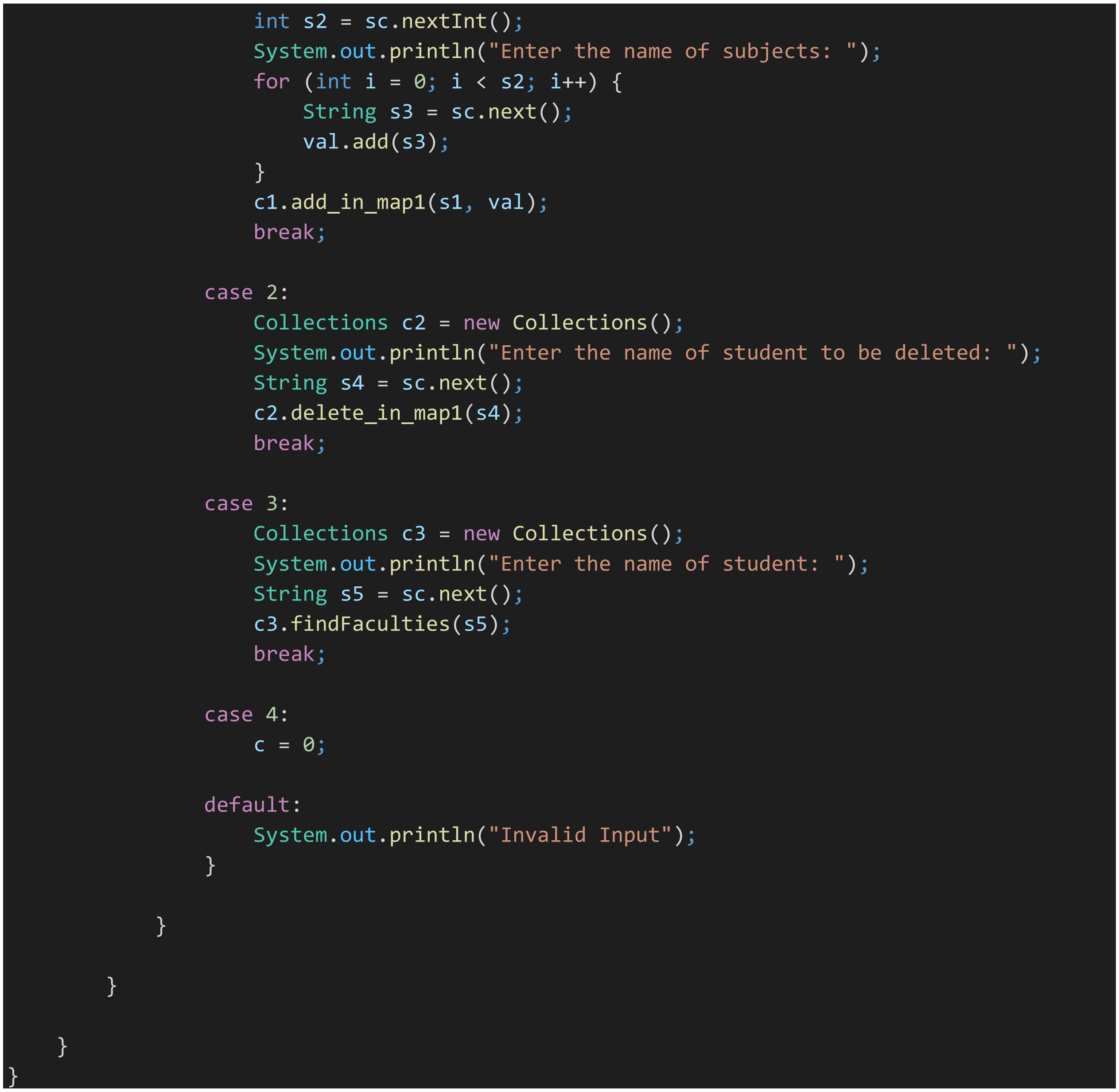
2.

Code:

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

class Collections {
    static Map<String, List<String>> map1 = new HashMap<String, List<String>>();
    static Map<String, String> map2 = new HashMap<String, String>();

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

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

    public void findFaculties(String key) {
        if (map1.containsKey(key)) {
            List<String> faculties = new ArrayList<String>();
            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) {
        int c = 1;
        List<String> valSetOne = new ArrayList<String>();
        valSetOne.add("Python");
        valSetOne.add("Maths");
        valSetOne.add("C");
        List<String> valSetTwo = new ArrayList<String>();
        valSetTwo.add("C");
```

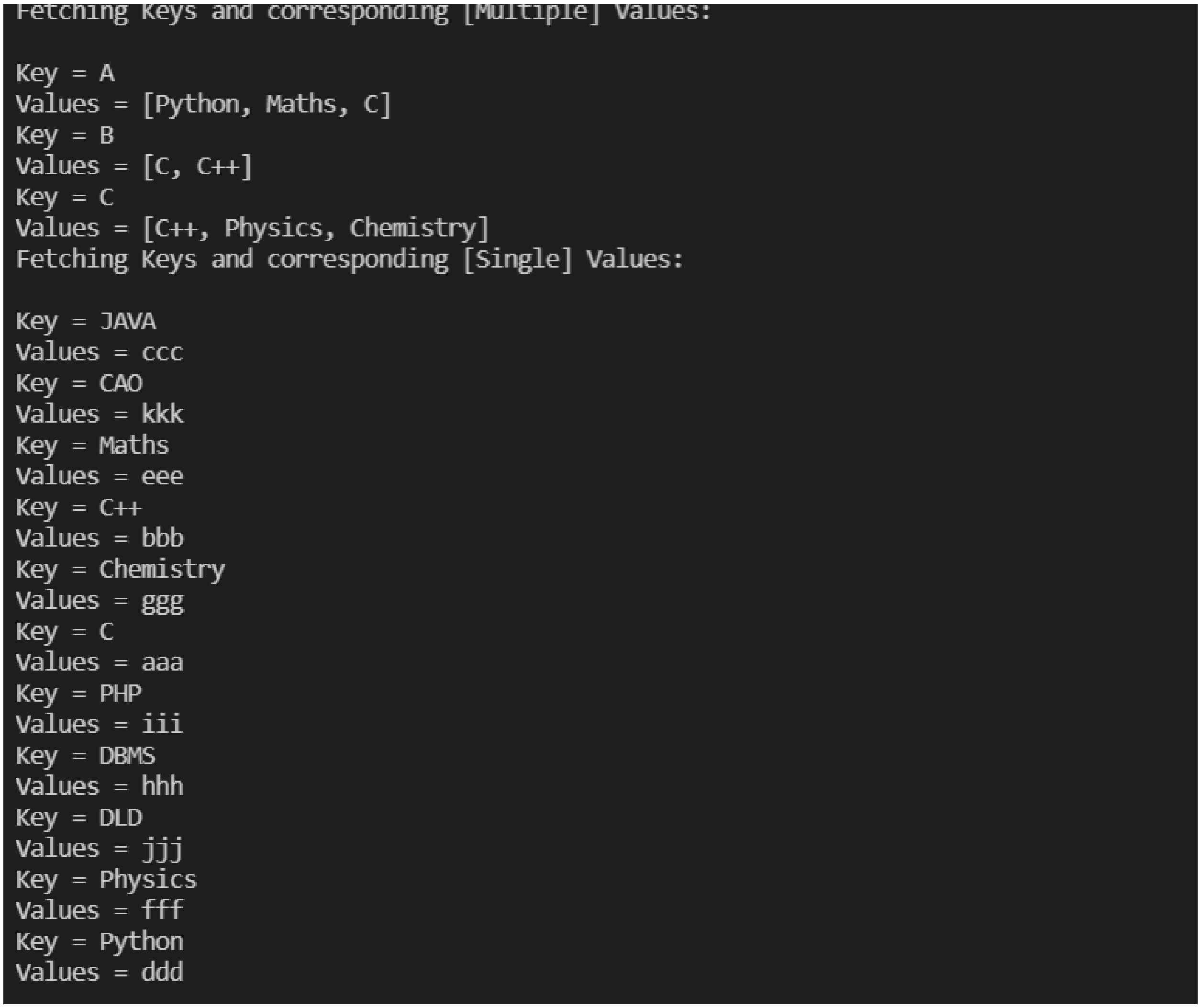


```
        valSetTwo.add("C++");
        List<String> valSetThree = new ArrayList<String>();
        valSetThree.add("C++");
        valSetThree.add("Physics");
        valSetThree.add("Chemistry");
        map1.put("A", valSetOne);
        map1.put("B", valSetTwo);
        map1.put("C", valSetThree);
        map2.put("Python", "ddd");
        map2.put("Maths", "eee");
        map2.put("C", "aaa");
        map2.put("C++", "bbb");
        map2.put("Physics", "fff");
        map2.put("Chemistry", "ggg");
        map2.put("DBMS", "hhh");
        map2.put("PHP", "iii");
        map2.put("DLD", "jjj");
        map2.put("JAVA", "ccc");
        map2.put("CAO", "kkk");
        System.out.println("Fetching Keys and corresponding [Multiple] Values:\n");
        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, String> entry2 : map2.entrySet()) {
            String key2 = entry2.getKey();
            String values2 = entry2.getValue();
            System.out.println("Key = " + key2);
            System.out.println("Values = " + values2);
        }

        while (c == 1) {
            System.out.println("\nEnter choice:\n1. For addition\n2. For Deletion\n3. For
Display\n4. Exit\n");
            {

                Scanner sc = new Scanner(System.in);
                int choice = sc.nextInt();
                switch (choice) {

                case 1:
                    Collections c1 = new Collections();
                    List<String> val = new ArrayList<String>();
                    System.out.println("Enter the name of student: ");
                    String s1 = sc.next();

                    System.out.println("Enter the no. of subjects: ");
```



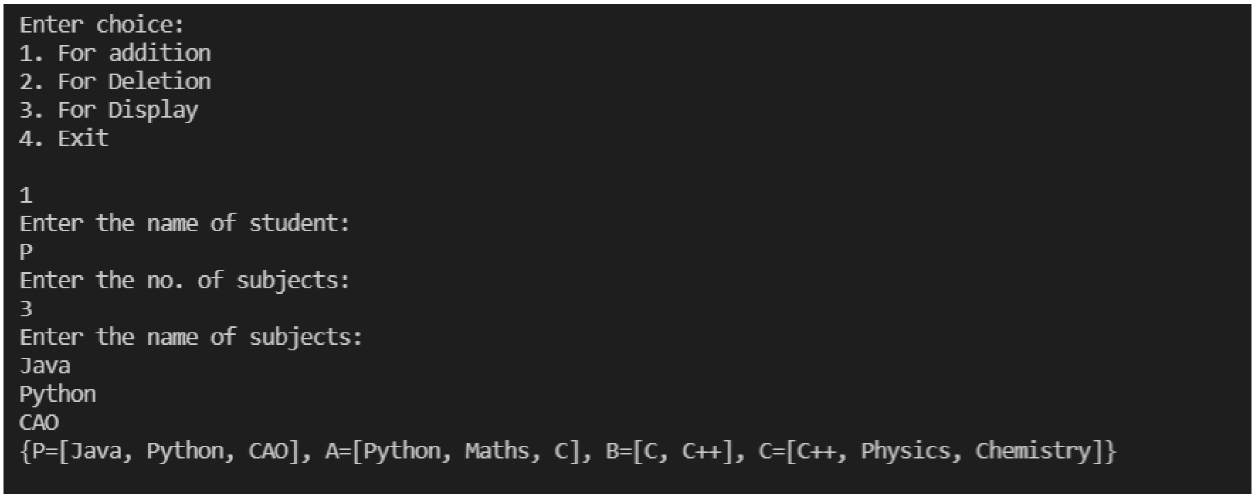
```
                    int s2 = sc.nextInt();
                    System.out.println("Enter the name of subjects: ");
                    for (int i = 0; i < s2; i++) {
                        String s3 = sc.next();
                        val.add(s3);
                    }
                    c1.add_in_map1(s1, val);
                    break;

                case 2:
                    Collections c2 = new Collections();
                    System.out.println("Enter the name of student to be deleted: ");
                    String s4 = sc.next();
                    c2.delete_in_map1(s4);
                    break;

                case 3:
                    Collections c3 = new Collections();
                    System.out.println("Enter the name of student: ");
                    String s5 = sc.next();
                    c3.findFaculties(s5);
                    break;

                case 4:
                    c = 0;

                default:
                    System.out.println("Invalid Input");
                }

            }

        }

    }
}
```

Output:



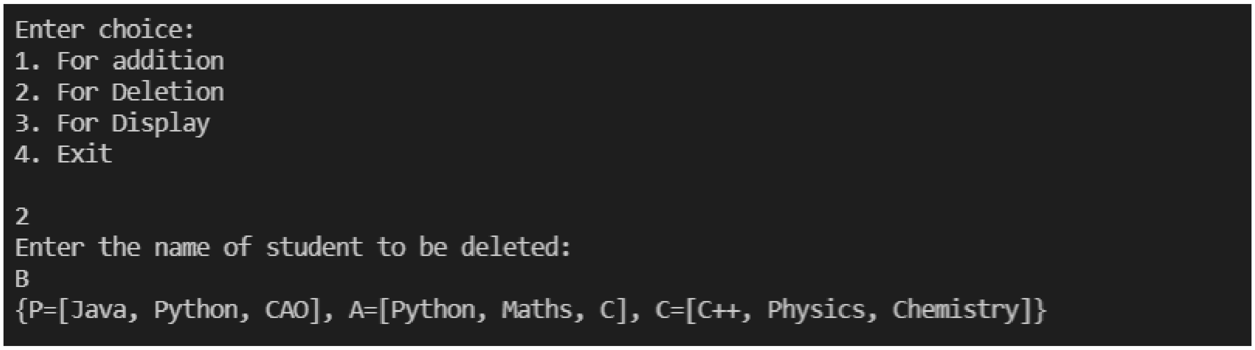
```
Fetching Keys and corresponding [Multiple] Values:

Key = A
Values = [Python, Maths, C]
Key = B
Values = [C, C++]
Key = C
Values = [C++, Physics, Chemistry]
Fetching Keys and corresponding [Single] Values:

Key = JAVA
Values = ccc
Key = CAO
Values = kkk
Key = Maths
Values = eee
Key = C++
Values = bbb
Key = Chemistry
Values = ggg
Key = C
Values = aaa
Key = PHP
Values = iii
Key = DBMS
Values = hhh
Key = DLD
Values = jjj
Key = Physics
Values = fff
Key = Python
Values = ddd
```

Adding :



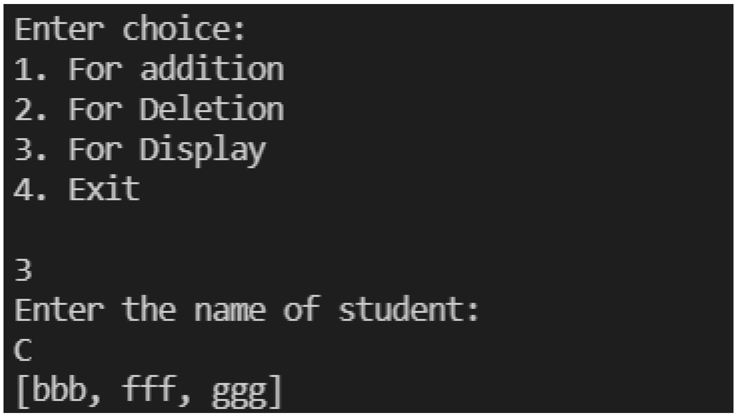
```
Enter choice:
1. For addition
2. For Deletion
3. For Display
4. Exit

1
Enter the name of student:
P
Enter the no. of subjects:
3
Enter the name of subjects:
Java
Python
CAO
{P=[Java, Python, CAO], A=[Python, Maths, C], B=[C, C++], C=[C++, Physics, Chemistry]}
```

Deleting:



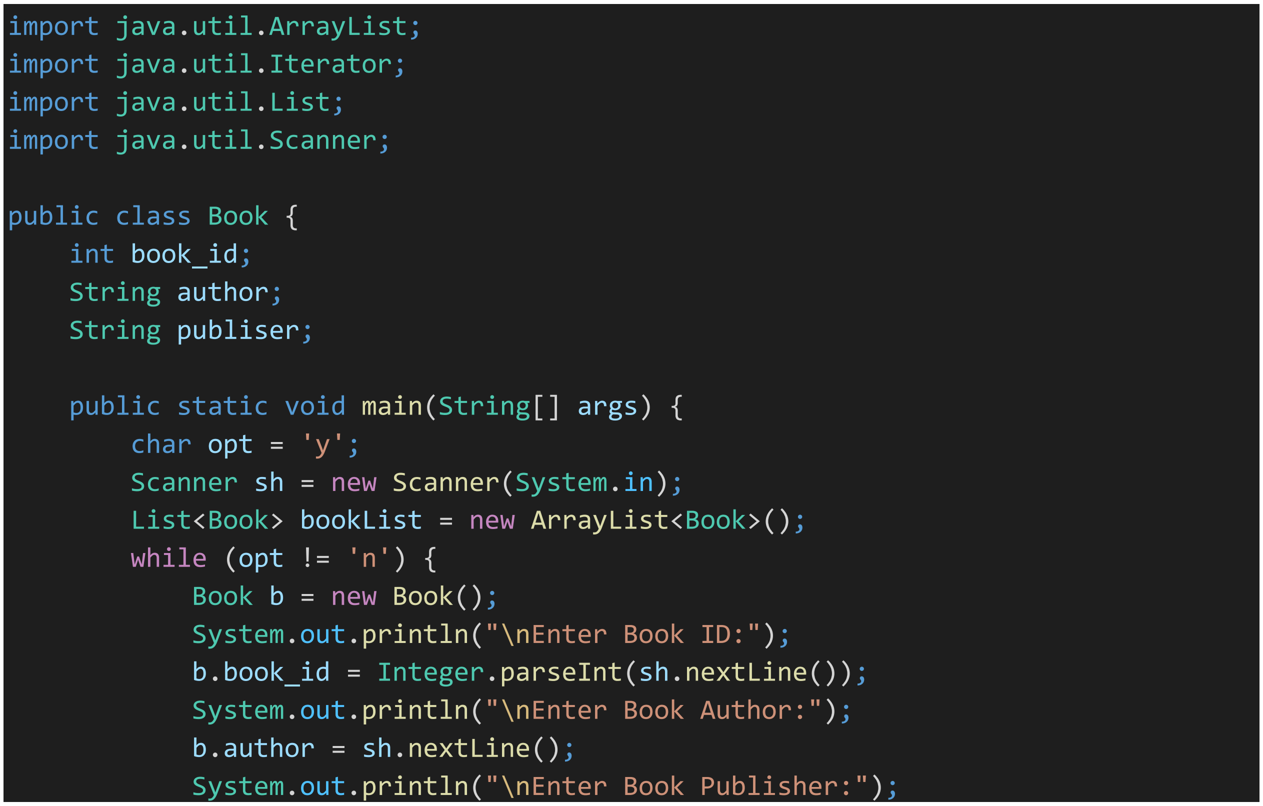
```
Enter choice:
1. For addition
2. For Deletion
3. For Display
4. Exit

2
Enter the name of student to be deleted:
B
{P=[Java, Python, CAO], A=[Python, Maths, C], C=[C++, Physics, Chemistry]}
```

Fetching:

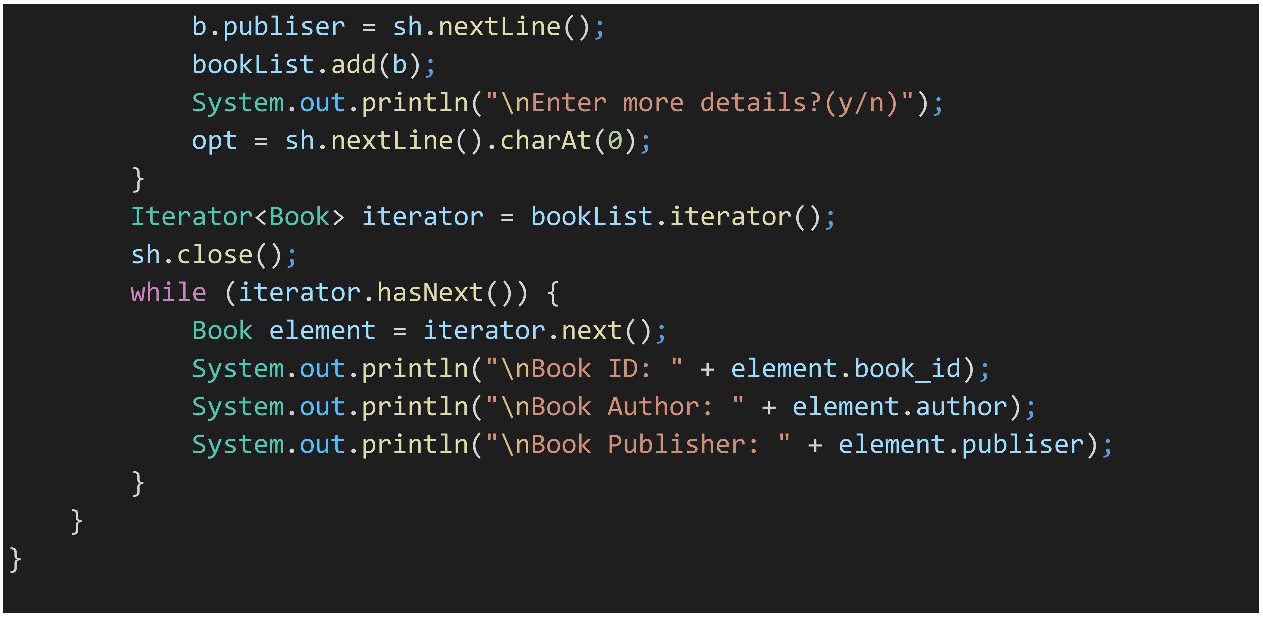


```
Enter choice:
1. For addition
2. For Deletion
3. For Display
4. Exit

3
Enter the name of student:
C
[bbb, fff, ggg]
```

3.

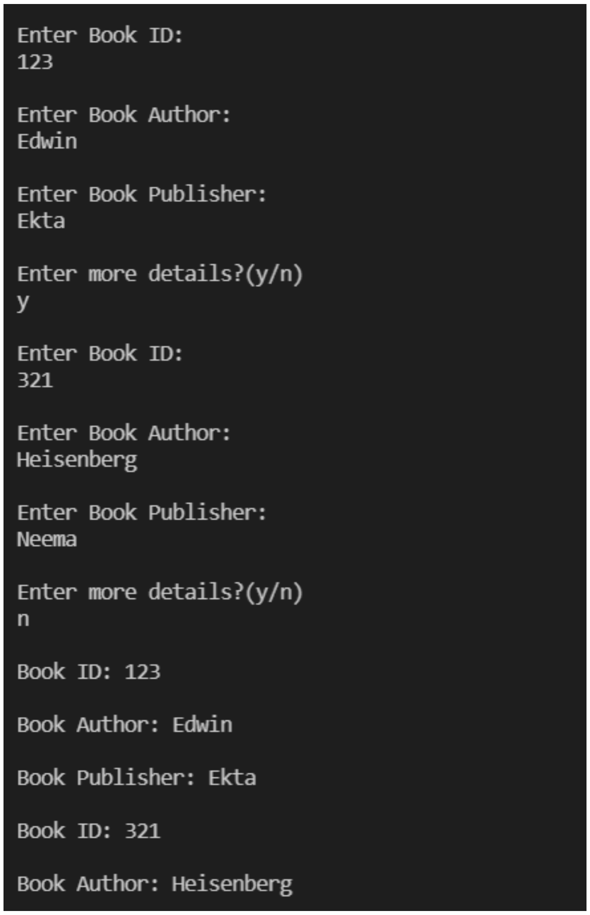


```java
import java.util.ArrayList;
import java.util.Iterator;
import java.util.List;
import java.util.Scanner;

public class Book {
    int book_id;
    String author;
    String publiser;

    public static void main(String[] args) {
        char opt = 'y';
        Scanner sh = new Scanner(System.in);
        List<Book> bookList = new ArrayList<Book>();
        while (opt != 'n') {
            Book b = new Book();
            System.out.println("\nEnter Book ID:");
            b.book_id = Integer.parseInt(sh.nextLine());
            System.out.println("\nEnter Book Author:");
            b.author = sh.nextLine();
            System.out.println("\nEnter Book Publisher:");
```



```
            b.publiser = sh.nextLine();
            bookList.add(b);
            System.out.println("\nEnter more details?(y/n)");
            opt = sh.nextLine().charAt(0);
        }
        Iterator<Book> iterator = bookList.iterator();
        sh.close();
        while (iterator.hasNext()) {
            Book element = iterator.next();
            System.out.println("\nBook ID: " + element.book_id);
            System.out.println("\nBook Author: " + element.author);
            System.out.println("\nBook Publisher: " + element.publiser);
        }
    }
}
```

Output:

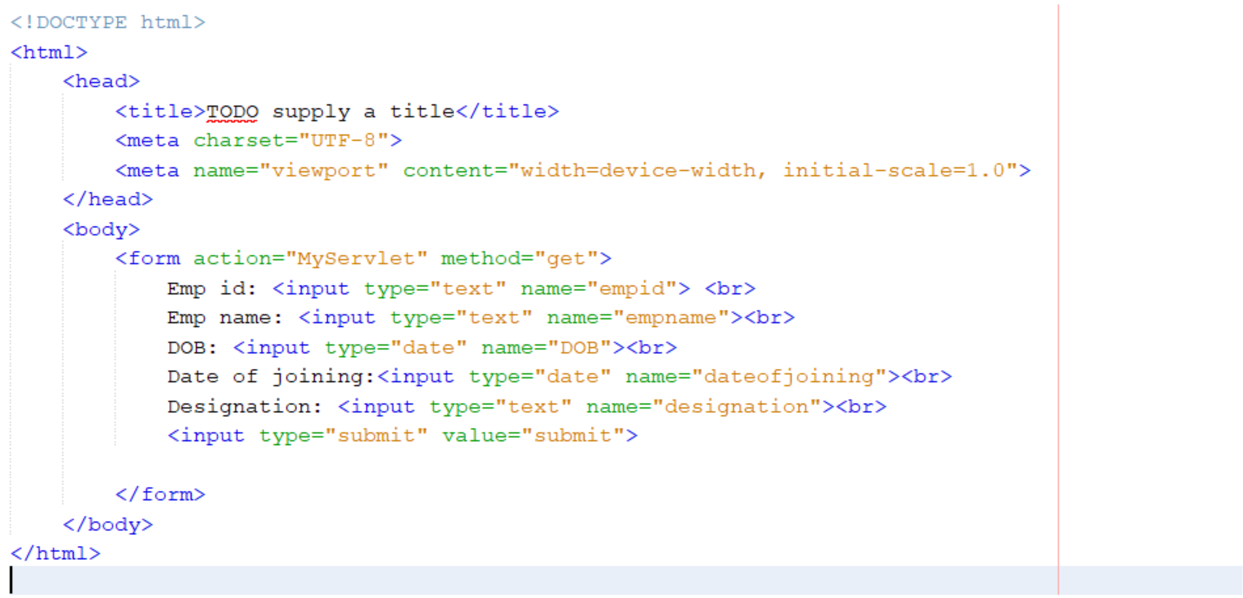


```
Enter Book ID:
123

Enter Book Author:
Edwin

Enter Book Publisher:
Ekta

Enter more details?(y/n)
y

Enter Book ID:
321

Enter Book Author:
Heisenberg

Enter Book Publisher:
Neema

Enter more details?(y/n)
n

Book ID: 123

Book Author: Edwin

Book Publisher: Ekta

Book ID: 321

Book Author: Heisenberg
```

Make a simple FORM using SERVLET :

Code:

Index.html:



```
<!DOCTYPE html>
<html>
    <head>
        <title>TODO supply a title</title>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
    </head>
    <body>
        <form action="MyServlet" method="get">
            Emp id: <input type="text" name="empid"> <br>
            Emp name: <input type="text" name="empname"><br>
            DOB: <input type="date" name="DOB"><br>
            Date of joining:<input type="date" name="dateofjoining"><br>
            Designation: <input type="text" name="designation"><br>
            <input type="submit" value="submit">

        </form>
    </body>
</html>
```

Servlet.java:

import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

/**

 *

 * @author ALFA SHERIA

 */

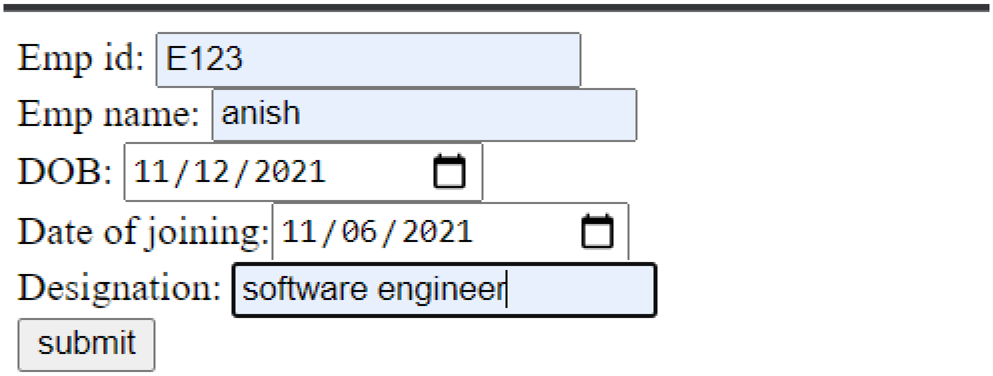
```
public class MyServlet extends HttpServlet {
  @Override
   protected void doGet(HttpServletRequest request, HttpServletResponse response)
          throws ServletException, IOException {
      response.setContentType("text/html");
      PrintWriter out = response.getWriter();


      String empid =request.getParameter("empid");
      String empname =request.getParameter("empname");
      String DOB =request.getParameter("DOB");
      String date_of_joining =request.getParameter("dateofjoining");
      String designation =request.getParameter("designation");


      out.print("Employee details:" + empid + empname + DOB + date_of_joining + designation);
   }
   }
```

Servlet mapping:

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app version="3.1" xmlns="http://xmlns.jcp.org/xml/ns/javaee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org/xml/ns/javaee/web-app_3_1.xsd">
   <servlet>
      <servlet-name>MyServlet</servlet-name>
      <servlet-class>MyServlet</servlet-class>
   </servlet>
   <servlet-mapping>
      <servlet-name>MyServlet</servlet-name>
```

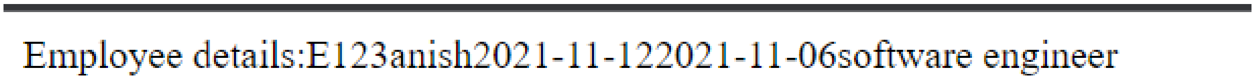
```
    <url-pattern>/MyServlet</url-pattern>
  </servlet-mapping>
  <session-config>
    <session-timeout>
      30
    </session-timeout>
  </session-config>
</web-app>
```

Output:



Emp id: E123
Emp name: anish
DOB: 11/12/2021
Date of joining: 11/06/2021
Designation: software engineer

After submission:



Employee details:E123anish2021-11-122021-11-06software engineer

Apache tomcat used for server.