

**CHANDIGARH UNIVERSITY  
UNIVERSITY INSTITUTE OF NGINEERING  
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**



<b>Submitted By:</b> Vivek Kumar(21BCS8129)		<b>Submitted To:</b> Neeru Sharma(E12950)	
<b>Subject Name</b>	Project Based Learning in Java Lab		
<b>Subject Code</b>	20CSP-321		
<b>Branch</b>	Computer Science and Engineering		
<b>Semester</b>	5 <sup>th</sup>		

## Experiment - 1

**Student Name: Vivek Kumar**

**Branch: BE-CSE(LEET)**

**Semester: 5<sup>th</sup>**

**Subject Name: Project Based Learning in Java Lab**

**UID: 21BCS8129**

**Section/Group: WM-20BCS-616/A**

**Date of Performance: 09/08/2022**

**Subject Code: 20CSP-321**

### 1. Aim/Overview of the practical:

Create an application to save the employee information using arrays.

### 2. Task to be done/ Which logistics used:

Write the program to create an application to save the employee information using arrays.

### 3. Software Requirements (For programming-based labs):

- JDK-8 or any
- Eclipse-IDE for Java

### 4. Steps for experiment/practical/Code:

```
import java.io.BufferedReader;
```

```
import java.io.IOException;
```

```
import java.io.InputStreamReader;
```

```
public class Employee {
```

```
    String empId;
```

```
    String depName;
```

```
    String empDesignation;
```

```
    String empName;
```

```
    String dateJoin;
```

```
    int basic;
```

```
    int hra;
```

```
    int it;
```

```
    char designationCode;
```

```
    public static int da;
```

```
    public Employee(
```

```
        String empId,
```

```
        String depName,
```

```
        String empDesignation,
```

```
        String empName,
```

```
        String dateJoin,
```

```
        int basic,
```

```
        int hra,
```

```
        int it,
```

```
        char designationCode
```

```
    ) {
```

```
        this.empId = empId;
```

```
        this.depName = depName;
```

```
        this.empDesignation = empDesignation;
```

```
        this.empName = empName;
```

```
this.dateJoin = dateJoin;
this.basic = basic;
this.hra = hra;
this.it = it;
this.designationCode = designationCode;
}

public static int da(char designationCode) {
    switch (designationCode) {
        case 'e':
        {
            da = 20000;
            break;
        }
        case 'c':
        {
            da = 32000;
            break;
        }
        case 'k':
        {
            da = 12000;
            break;
        }
        case 'r':
        {
            da = 15000;
            break;
        }
        case 'm':
        {
            da = 40000;
            break;
        }
        default:
            throw new IllegalStateException("Unexpected value: " + designationCode);
    }
    return da;
}

public static int salary(int basic, int hra, int da, int it) {
    int salary = basic + hra + da - it;
    return salary;
}
```

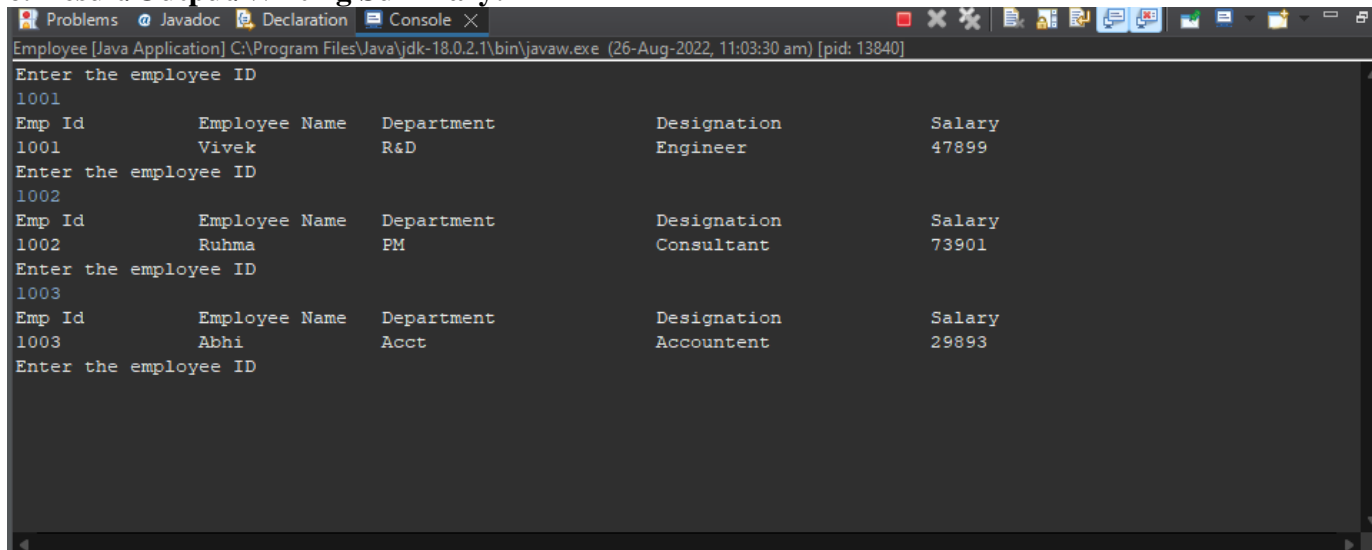
```
public static void details(  
    String empId,  
    String empName,  
    String depName,  
    String empDesignation,  
    int salary  
) {  
    System.out.println(  
        "Emp Id\tEmployee Name\tDepartment\tDesignation\tSalary"  
    );  
    System.out.println(empId + "\t" + empName + "\t" + depName + "\t" + empDesignation + "\t" + salary);  
}  
  
public static void main(String[] args) throws IOException {  
    boolean val = true;  
    BufferedReader bufferedReader = new BufferedReader(  
        new InputStreamReader(System.in)  
    );  
    String empId;  
    int c = 0;  
    Employee[] employees = new Employee[3];  
    employees[0] =  
        new Employee(  
            "1001",  
            "R&D",  
            "Engineer",  
            "Vivek",  
            "1/04/2022",  
            20000,  
            8000,  
            3000,  
            'e'  
        );  
    employees[1] =  
        new Employee(  
            "1002",  
            "PM",  
            "Consultant",  
            "Ruhma",  
            "23/08/2022",  
            30000,  
            12000,  
            9000,  
            'c'  
        );  
};
```

```
employees[2] =
    new Employee(
        "1003",
        "Acct",
        "Accountent",
        "Abhi",
        "12/11/2008",
        10000,
        8000,
        1000,
        'k'
    );
while(val) {
    System.out.println("Enter the employee ID ");
    empId = bufferedReader.readLine();
    for (int i = 0; i < 3; i++) {
        if (employees[i].empId.equals(empId)) {
            c = 1;
            int salary = salary(
                employees[i].basic,
                employees[i].hra,
                da(employees[i].designationCode),
                employees[i].designationCode
            );
            details(
                employees[i].empId,
                employees[i].empName,
                employees[i].depName,
                employees[i].empDesignation,
                salary
            );
            break;
        }
    }
    if (c != 1) System.out.println("Entered employee ID not found");
}
}
```

## 5. Observations/Discussions/ Complexity Analysis:

Here we have created the Array with the size of 3 and Data inserted, calculated the DA and Actual salary. Moreover, I've given the Search method with EmpId.

## 6. Result/Output/Writing Summary:



```

Employee [Java Application] C:\Program Files\Java\jdk-18.0.2.1\bin\javaw.exe (26-Aug-2022, 11:03:30 am) [pid: 13840]
Enter the employee ID
1001
Emp Id      Employee Name  Department  Designation  Salary
1001      Vivek          R&D         Engineer     47899
Enter the employee ID
1002
Emp Id      Employee Name  Department  Designation  Salary
1002      Ruhma          PM          Consultant   73901
Enter the employee ID
1003
Emp Id      Employee Name  Department  Designation  Salary
1003      Abhi           Acct        Accountant   29893
Enter the employee ID
  
```

## Learning outcomes (What I have learnt):

1. Learn How to create the array.
2. Array manipulation in java.

## Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.			
2.			
3.			

## Experiment - 2

**Student Name: Vivek Kumar**

**Branch: BE-CSE(LEET)**

**Semester: 5<sup>th</sup>**

**Subject Name: Project Based Learning in Java Lab**

**UID: 21BCS8129**

**Section/Group: WM-20BCS-616/A**

**Date of Performance: 20/08/2022**

**Subject Code: 20CSP-321**

### 1. Aim/Overview of the practical:

Design and implement a simple inventory control system for a small video rental store.

### 2. Task to be done/ Which logistics used:

Write the program to design and implement a simple inventory control system for a small video rental store.

### 3. Software Requirements (For programming-based labs):

- JDK-8 or any
- Eclipse-IDE for Java

### 4. Steps for experiment/practical/Code:

```
package Unit1;
```

```
import java.util.Scanner;
```

```
class Video {
```

```
    String videoName;
```

```
    boolean checkOut;
```

```
    int rating;
```

```
    String getName()
```

```
    {
```

```
        return videoName;
```

```
    }
```

```
    void doCheckOut()
```

```
    {
```

```
        checkOut = true;
```

```
    }
```

```
    void doReturn()
```

```
    {
```

```
        checkOut = false;
```

```
    }
```

```
    void receiveRating(int rating)
```

```
{
    this.rating = rating;
}

int getRating()
{
    return rating;
}

boolean getCheckOut()
{
    return checkOut;
}

public Video(String videoName)
{
    this.videoName = videoName;
}

}

class VideoStore {
    Video store[]= new Video[20];
    static int a=0;
    void addVideo(String name)
    {
        store[a] = new Video(name);
        store[a].checkOut = false;
        store[a].receiveRating(0);
        System.out.println("video " +name +" added sucessfully");
        a++;
    }

    void doCheckOut(String name)
    {
        for(int i=0; i<a;i++)
        {
            if(store[i].getName().equals(name))
            {
```



```
store[i].doCheckOut();
System.out.println("Video " +name +" removed successfully from "+i +" location");
} else {
    System.out.println("No such video exists at:" +i+" location");
}
}
}
void doReturn(String name)
{
    for(int i= 0; i<a;i++)
    {
        if(store[i].getName().equals(name))
        {
            store[i].doReturn();
            System.out.println("Video returned: " +name +" from location "+i);
        } else{
            System.out.println("No such video exists at locations:" +i);
        }
    }
}
void receiveRating(String name, int rating)
{
    for(int i= 0; i<a;i++)
    {
        if(store[i].getName().equals(name))
        {
            store[i].receiveRating(rating);
        }
    }
    System.out.println("Ratings " +rating +" has been mapped to the video " +name);
}
void listInventory()
{
    for(int i= 0; i<a;i++)
    {
        if(!store[i].getCheckOut())
        {
            System.out.print("Videos (location "+i+ "): "+store[i].videoName+" Ratings (location "+i+ "):
"+store[i].getRating()+"\n");
        }
    }
}
```

```
}  
public void exit() {  
    System.exit(0);  
}  
}  
class VideoStoreLauncher {  
    public static void main(String[] args) {  
  
        VideoStore obj = new VideoStore();  
        int choice;  
        String videoName;  
        int rating;  
        boolean status = true;  
        while(status)  
        {  
            System.out.println("MAIN MENU");  
            System.out.println("*****");  
            System.out.println("1.Add Videos:");  
            System.out.println("2.Check Out Videos:");  
            System.out.println("3.Return Videos:");  
            System.out.println("4.Receive Rating:");  
            System.out.println("5.List Inventory:");  
            System.out.println("6.Exit");  
            System.out.println("Enter your choice:");  
            Scanner sc = new Scanner(System.in);  
            choice = sc.nextInt();  
            switch(choice)  
            {  
            case 1:  
                {  
                    System.out.println("Enter the name of the video you want to add");  
                    videoName = sc.next();  
                    obj.addVideo(videoName);  
                    break;  
                }  
  
            case 2:  
                {
```

```
        System.out.println("Enter the name of video to checkout");
        videoName = sc.next();
        obj.doCheckOut(videoName);
        break;
    }
    case 3:
    {
        System.out.println("Enter the video name to return");
        videoName = sc.next();
        obj.doReturn(videoName);
        break;
    }
    case 4:
    {
        System.out.println("Enter the name of video you want to rate");
        videoName = sc.next();
        System.out.println("Enter the Ratings for this video");
        rating = sc.nextInt();
        obj.receiveRating(videoName, rating);
        break;
    }
    case 5:
    {
        obj.listInventory();
        break;
    }
    case 6:
    {System.out.println("Exiting...!! Thanks for using the application");
        obj.exit();
        break;
    }
    default:
    {
        System.out.println("Wrong input!!");
    }
    }
}
```

## 5. Observations/Discussions/ Complexity Analysis:

Here we have created the VideoStoreLauncher, VideoStore and Video and I have Passed all the Parameters according to the Requirement given in the question.

## 6. Result/Output/Writing Summary:

```
Java - JavaLab/src/Unit1/VideoStoreLauncher.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
<terminated> VideoStoreLauncher (1) [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\java.exe (27-Aug-2022, 7:33:17 pm - 7:40:19 pm) [pid: 11096]
Problems Javadoc Declaration Console X
MAIN MENU
*****
1.Add Videos:
2.Check Out Videos:
3.Return Videos:
4.Receive Rating:
5.List Inventory:
6.Exit
Enter your choice:
1
Enter the name of the video you want to add
DDLJ
video DDLJ added successfully
MAIN MENU
*****
1.Add Videos:
2.Check Out Videos:
3.Return Videos:
4.Receive Rating:
5.List Inventory:
6.Exit
Enter your choice:
1
Enter the name of the video you want to add
Mann
video Mann added successfully
MAIN MENU
*****
1.Add Videos:
2.Check Out Videos:
3.Return Videos:
4.Receive Rating:
5.List Inventory:
6.Exit
Enter your choice:
1
Enter the name of the video you want to add
Harry
video Harry added successfully
MAIN MENU
*****
1.Add Videos:
2.Check Out Videos:
3.Return Videos:
4.Receive Rating:
5.List Inventory:
6.Exit
Enter your choice:
5
Videos (location 0): DDLJ Ratings (location 0): 0
Videos (location 1): Mann Ratings (location 1): 0
Videos (location 2): Harry Ratings (location 2): 0
MAIN MENU
*****
1.Add Videos:
2.Check Out Videos:
3.Return Videos:
4.Receive Rating:
5.List Inventory:
6.Exit
Enter your choice:
4
Enter the name of video you want to rate
DDLJ
Enter the Ratings for this video
5
Ratings 5 has been mapped to the video DDLJ
MAIN MENU
*****
1.Add Videos:
2.Check Out Videos:
3.Return Videos:
4.Receive Rating:
5.List Inventory:
6.Exit
Enter your choice:
4
Enter the name of video you want to rate
Mann
Enter the Ratings for this video
6
Ratings 6 has been mapped to the video Mann
MAIN MENU
*****
1.Add Videos:
2.Check Out Videos:
3.Return Videos:
4.Receive Rating:
5.List Inventory:
6.Exit
Enter your choice:
4
Enter the name of video you want to rate
Harry
Enter the Ratings for this video
4
Ratings 4 has been mapped to the video Harry
MAIN MENU
*****
1.Add Videos:
2.Check Out Videos:
3.Return Videos:
```

```

4.Receive Rating:
5.List Inventory:
6.Exit
Enter your choice:
5
Videos (location 0): DDLJ Ratings (location 0): 5
Videos (location 1): Mann Ratings (location 1): 6
Videos (location 2): Harry Ratings (location 2): 4
MAIN MENU
*****
1.Add Videos:
2.Check Out Videos:
3.Return Videos:
4.Receive Rating:
5.List Inventory:
6.Exit
Enter your choice:
3
Enter the video name to return
DDLJ
Video returned: DDLJ from location 0
No such video exists at locations:1
No such video exists at locations:2
MAIN MENU
*****
1.Add Videos:
2.Check Out Videos:
3.Return Videos:
4.Receive Rating:
5.List Inventory:
6.Exit
Enter your choice:
2
Enter the name of Video to checkout
Mann
No such video exists at:0 location
Video Mann removed successfully from 1 location
No such video exists at:2 location
MAIN MENU
*****
1.Add Videos:
2.Check Out Videos:
3.Return Videos:
4.Receive Rating:
5.List Inventory:
6.Exit
Enter your choice:
5
Videos (location 0): DDLJ Ratings (location 0): 5
Videos (location 2): Harry Ratings (location 2): 4
MAIN MENU
*****
1.Add Videos:
2.Check Out Videos:
3.Return Videos:
4.Receive Rating:
5.List Inventory:
6.Exit
Enter your choice:
6
Exiting.....!! Thanks for using the application
4

```

## Learning outcomes (What I have learnt):

1. Learn How use the inheritance concept.
2. java classes and all the features.

## Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.			
2.			
3.			

## Experiment - 3

**Student Name: Vivek Kumar**

**Branch: BE-CSE(LEET)**

**Semester: 5<sup>th</sup>**

**Subject Name: Project Based Learning in Java Lab**

**UID: 21BCS8129**

**Section/Group: WM-20BCS-616/A**

**Date of Performance: 16/08/2022**

**Subject Code: 20CSP-321**

### 1. Aim/Overview of the practical:

Create an application to calculate interest for FDs, RDs based on certain conditions using inheritance.

### 2. Task to be done/ Which logistics used:

Write a program to create an application to make an Account holders list and calculate interest for FDs, RDs based on certain conditions using inheritance.

### 3. Software Requirements (For programming-based labs):

- JDK-8 or any
- Eclipse-IDE for Java

### 4. Steps for experiment/practical/Code:

```
package Unit1;
```

```
import java.util.Scanner;
```

```
class InvalidAgeException extends Exception{ }
```

```
class InvalidAmountException extends Exception{ }
```

```
class InvalidDaysException extends Exception{ }
```

```
class InvalidMonthsException extends Exception{ }
```

```
abstract class Account {
```

```
    double interestRate;
```

```
    double amount;
```

```
    abstract double calculateInterest(double amount)throws
```

```
InvalidMonthsException,InvalidAgeException,InvalidAmountException ,InvalidDaysException;
```

```
}
```

```
class FDaccount extends Account {
```

```
    double FDinterestRate;
```

```
    double FDAmount;
```

```
    int noOfDays;
```

```
    int ageOfACHolder;
```

```
    double General, SCitizen;
```

```
    Scanner FDSscanner = new Scanner(System.in);
```

```
double calculateInterest(double amount) throws
InvalidAgeException,InvalidAmountException,InvalidDaysException {
    this.FDAmount = amount;

    System.out.println("Enter FD days");
    noOfDays = FDScanner.nextInt();
    System.out.println("Enter FD age holder ");
    ageOfACHolder = FDScanner.nextInt();
    if (amount < 0) {
        throw new InvalidAmountException();
    }
    if(noOfDays<0){
        throw new InvalidDaysException();
    }
    if(ageOfACHolder<0){
        throw new InvalidAgeException();
    }
    if (amount < 10000000) {
        if (noOfDays >= 7 && noOfDays <= 14) {
            General = 0.0450;
            SCitizen = 0.0500; }
        else if (noOfDays >= 15 && noOfDays <= 29) {
            General = 0.0470;
            SCitizen = 0.0525;
        } else if (noOfDays >= 30 && noOfDays <= 45) {
            General = 0.0550;
            SCitizen = 0.0600;
        } else if (noOfDays >= 45 && noOfDays <= 60) {
            General = 0.0700;
            SCitizen = 0.0750;
        } else if (noOfDays >= 61 && noOfDays <= 184) {
            General = 0.0750;
            SCitizen = 0.0800;
        } else if (noOfDays >= 185 && noOfDays <= 365) {
            General = 0.0800;
            SCitizen = 0.0850;
        }
        FDinterestRate = (ageOfACHolder < 50) ? General : SCitizen;
    } else {
        if (noOfDays >= 7 && noOfDays <= 14) {
            interestRate = 0.065;
        } else if (noOfDays >= 15 && noOfDays <= 29) {
            interestRate = 0.0675;
        } else if (noOfDays >= 30 && noOfDays <= 45) {
            interestRate = 0.00675;
        }
    }
}
```

```
} else if (noOfDays >= 45 && noOfDays <= 60) {
    interestRate = 0.080;
} else if (noOfDays >= 61 && noOfDays <= 184) {
    interestRate = 0.0850;
} else if (noOfDays >= 185 && noOfDays <= 365) {
    interestRate = 0.10;
}
}
return FDAmount * FDinterestRate;
}
}
```

```
class RDaccount extends Account {
    double RDInterestRate;
    double RDamount;
    int noOfMonths;
    double monthlyAmount;
    double General, SCitizen;
    Scanner RDScanner = new Scanner(System.in);
    double calculateInterest(double Ramount) throws InvalidMonthsException, InvalidAmountException, InvalidAgeException {
        this.RDamount = Ramount;
        System.out.println("Enter RD months");
        noOfMonths = RDScanner.nextInt();
        System.out.println("Enter RD holder age");
        int age = RDScanner.nextInt();
        if (RDamount < 0) {
            throw new InvalidAmountException();
        }
        if(noOfMonths<0){
            throw new InvalidMonthsException();
        }
        if(age<0){
            throw new InvalidAgeException();
        }
        if (noOfMonths >= 0 && noOfMonths <= 6) {
            General = .0750;
            SCitizen = 0.080;
        } else if (noOfMonths >= 7 && noOfMonths <= 9) {
            General = .0775;
            SCitizen = 0.0825;
        } else if (noOfMonths >= 10 && noOfMonths <= 12) {
            General = .0800;
            SCitizen = 0.0850;
        } else if (noOfMonths >= 13 && noOfMonths <= 15) {
```



```
        General = .0825;
        SCitizen = 0.0875;
    } else if (noOfMonths >= 16 && noOfMonths <= 18) {
        General = .0850;
        SCitizen = 0.0900;
    } else if (noOfMonths >= 22) {
        General = .0875;
        SCitizen = 0.0925;
    }
    RDInterestRate = (age < 50) ? General : SCitizen;
    return RDamount * RDInterestRate;

}

}

class SBaccount extends Account {
    double SBamount , SbInterestRate, interest;
    Scanner SBScanner = new Scanner(System.in);

    double calculateInterest(double amount) throws InvalidAmountException{
        this.SBamount = amount;
        if(SBamount < 0 ){
            throw new InvalidAmountException();
        }
        System.out.println("Select account type \n1. NRI \n2. Normal ");
        int accountChoice = SBScanner.nextInt();
        switch (accountChoice) {
            case 1:
                SbInterestRate = .06;
                break;
            case 2:
                SbInterestRate = .04;
                break;
            default:
                System.out.println("Please choose right account again");
        }
        return amount * SbInterestRate;
    }
}

public class InterestCalculator {

    public static void main(String[] args) {
        boolean val=true;
```

```
Scanner sc = new Scanner(System.in);
while(val) {
    System.out.println("SELECT THE OPTIONS " + "\n1." + " Interest Calculator-SB" + " \n2." + "
Interest Calculator-FD" + "\n3." + " InterestCalculator-RD" + "\n4 " + " Exit");
    int choice = sc.nextInt();
    switch (choice) {
        case 1:
            SBaccount sb = new SBaccount();
            try {
                System.out.println("Enter the Average SB amount ");
                double amount = sc.nextDouble();
                System.out.println("Interest gained is : Rs " + sb.calculateInterest(amount));

            } catch (InvalidAmountException e) {
                System.out.println("Exception : Invalid amount");
            }
            break;
        case 2:
            try {
                FDaccount fd = new FDaccount();
                System.out.println("Enter the FD Amount");
                double fAmount = sc.nextDouble();
                System.out.println("Interest gained is: Rs " + fd.calculateInterest(fAmount));
            } catch (InvalidAgeException e) {
                System.out.println("Invalid Age Entered");
            } catch (InvalidAmountException e) {
                System.out.println("Invalid Amount Entered");
            }

            } catch (InvalidDaysException e) {
                System.out.println("Invalid Days Entered");
            }
            break;
        case 3:
            try {
                RDaccount rd = new RDaccount();
                System.out.println("Enter the RD amount");
                double Ramount = sc.nextDouble();
                System.out.println("Interest gained is: Rs " + rd.calculateInterest(Ramount));
            }
            catch (InvalidAgeException e) {
                System.out.println("Invalid Age Entered");
            } catch (InvalidAmountException e) {
                System.out.println("Invalid Amount Entered");
            } catch (InvalidMonthsException e) {
                System.out.println("Invalid Days Entered");
            }
    }
```

```

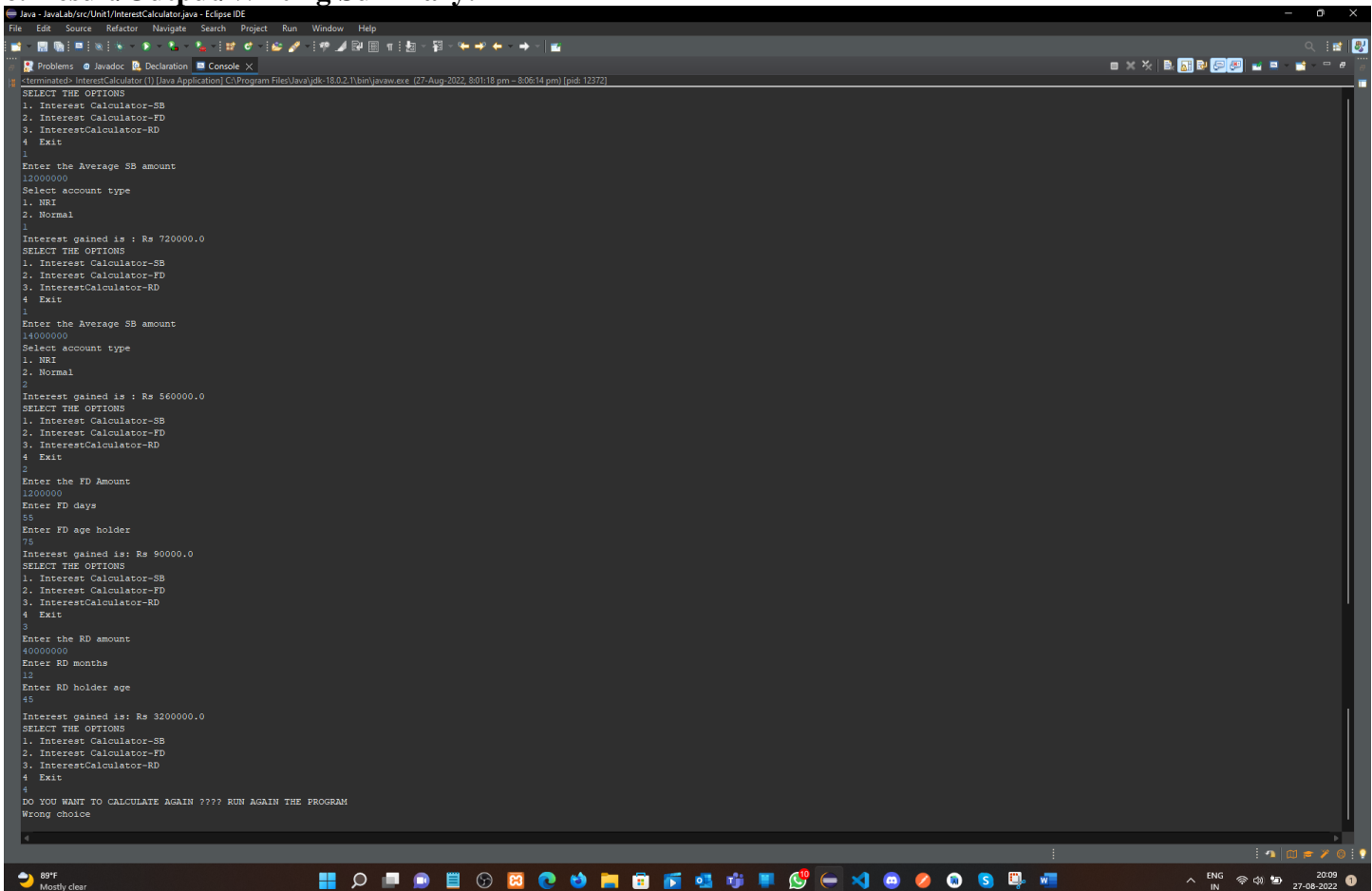
    }
    break;
case 4:
    val=false;
    System.out.println("DO YOU WANT TO CALCULATE AGAIN ???? " + " "
        + "RUN AGAIN THE PROGRAM");
default:
    System.out.println("Wrong choice");
}
}
}
sc.close();
}
}
}

```

## 5. Observations/Discussions/ Complexity Analysis:

Based on the questions here I have created the abstract class named as Account, and then FDaccount, RDaccount and SBaccount class which extends the Account class and then Final class I have created the IntrestCalculator which contains the main method of java program that is based on the question.

## 6. Result/Output/Writing Summary:



```

<terminated> InterestCalculator (1) [Java Application] C:\Program Files\Java\jdk-18.0.2\bin\javaw.exe (27-Aug-2022, 8:01:18 pm - 8:06:14 pm) [pid: 12372]
SELECT THE OPTIONS
1. Interest Calculator-SB
2. Interest Calculator-FD
3. InterestCalculator-RD
4. Exit
1
Enter the Average SB amount
12000000
Select account type
1. NRI
2. Normal
1
Interest gained is : Rs 720000.0
SELECT THE OPTIONS
1. Interest Calculator-SB
2. Interest Calculator-FD
3. InterestCalculator-RD
4. Exit
1
Enter the Average SB amount
14000000
Select account type
1. NRI
2. Normal
2
Interest gained is : Rs 560000.0
SELECT THE OPTIONS
1. Interest Calculator-SB
2. Interest Calculator-FD
3. InterestCalculator-RD
4. Exit
2
Enter the FD Amount
1200000
Enter FD days
55
Enter FD age holder
75
Interest gained is: Rs 90000.0
SELECT THE OPTIONS
1. Interest Calculator-SB
2. Interest Calculator-FD
3. InterestCalculator-RD
4. Exit
3
Enter the RD amount
40000000
Enter RD months
12
Enter RD holder age
45
Interest gained is: Rs 320000.0
SELECT THE OPTIONS
1. Interest Calculator-SB
2. Interest Calculator-FD
3. InterestCalculator-RD
4. Exit
4
DO YOU WANT TO CALCULATE AGAIN ???? RUN AGAIN THE PROGRAM
Wrong choice
4

```

**Learning outcomes (What I have learnt):**

1. Here we have learnt the Concept of Inheritance with the Abstract class
2. And finding the Interest, SB, RD & FD based on the Amount, Citizenship and Age group.

**Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):**

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.			
2.			
3.			

## LAB INDEX

NAME: Vivek Kumar

SUBJECTNAME: Project Based Learning in Java Lab

UID: 21BCS8129

SUBJECTCODE: 20CSP-314

SECTION: WM-20BCS-616/A

Sr. No	Program	Date	Evaluation				Sign
			LW (12)	VV (10)	FW (8)	Total (30)	
1	Create an application to save the employee information using arrays.	09-08-2022					
2	Design and implement a simple inventory control system for a small video rentalstore.	23-08-2022					
3	Create a application to calculate interest for FDs, RDs based on certain conditions using inheritance.	02-09-2022					
4	Create a program to show the usage of Sets of Collection interface.	27-09-2022					
5	Create a program to set view of Keys from Java Hashtable.						
6	Write a Program to perform the basic operations like insert, delete, display and search in list. List contains String object items where these operations are to be performed.						
7	Create a menu based Java application with the following options.1.Add an Employee2.Display All3.Exit If option 1 is selected, the application should gather details of the employee like employee name, employee id, designation and salary and store it in a file. If option 2 is selected, the application should display all the employee details. If option 3 is selected the application should exit.						
8	Create a palindrome creator application for making a longest possible palindrome out of given input string.						
9	Create a Servlet/ application with a facility to print any message on web browser.						
10	Create JSP application for addition, multiplication and division.						



**CHANDIGARH UNIVERSITY  
UNIVERSITY INSTITUTE OF NGINEERING  
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**



<b>Submitted By:</b> Vivek Kumar(21BCS8129)		<b>Submitted To:</b> Neeru Sharma(E12950)	
<b>Subject Name</b>	Project Based Learning in Java Lab		
<b>Subject Code</b>	20CSP-321		
<b>Branch</b>	Computer Science and Engineering		
<b>Semester</b>	5 <sup>th</sup>		

## Experiment - 4

**Student Name: Vivek Kumar****Branch: BE-CSE(LEET)****Semester: 5<sup>th</sup>****Subject Name: Project Based Learning in Java Lab****UID: 21BCS8129****Section/Group: 20BCS-WM-616/A****Date of Performance: 27/09/2022****Subject Code: 20CSP-321**

### 1. Aim/Overview of the practical:

Create a program to show the usage of Sets of Collection interface.

### 2. Task to be done/ Which logistics used:

Write the program to create an application to perform a set manipulation.

### 3. Software Requirements (For programming-based labs):

- JDK-8 or any
- Eclipse-IDE for Java

### 4. Steps for experiment/practical/Code:

```
package unit2;
```

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

```
public class WorkSheet4 {
```

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

```
        Set<Character> mySet1 = new HashSet<Character>();
```

```
            mySet1.add('A');
```

```
            mySet1.add('B');
```

```
            mySet1.add('C');
```

```
            mySet1.add('A');
```

```
            mySet1.add('B');
```

```
        System.out.println("\nmySet1: " + mySet1);
```

```
        List<Character> list = new ArrayList<Character>();
```

```
            list.add('A');
```

```
            list.add('A');
```

```
            list.add('B');
```

```
            list.add('C');
```

```
            list.add('B');
```

```
        Set<Character> mySet2 = new HashSet<Character>(list);
```

```
        System.out.println("\nlist: " + list);
```

```
        System.out.println("\nmySet2: " + mySet2);
```

```
        System.out.println("\nMySet1 matches mySet2: " + mySet1.equals(mySet2));
```

```
mySet2.remove('A');
System.out.println("\nmySet2: " + mySet2);
System.out.println("\nMySet1 matches mySet2: " + mySet1.equals(mySet2));

System.out.println("\nMySet1 contains all the elements: " + mySet1.containsAll(list));
System.out.println("\nMySet2 contains all the elements: " + mySet2.containsAll(list));

System.out.println("\nIterator Implementation");
Iterator<Character> iterator = mySet1.iterator();
while (iterator.hasNext()) {
    System.out.println("Iterator loop: " + iterator.next());
}

System.out.println("\nFor loop Implementation");
for(Object str:mySet1) {
    System.out.println("For each loop "+str);
}

mySet1.clear();
System.out.println("\nmySet1 is Empty: " + mySet1.isEmpty());

System.out.println("\nmySet1 has: " + mySet1.size() + " Elements");
System.out.println("\nmySet2 has: " + mySet2.size() + " Elements");

System.out.println("\nArray Conversion");
Object[] array = mySet1.toArray(new String[mySet2.size()]);
System.out.println("The array:" + Arrays.toString(array));
}
}
```

### 5. Observations/Discussions/ Complexity Analysis:

Here we have created the Set, list and Iterator and Data inserted, performed all the operation of set and Hashset.



## 6. Result/Output/Writing Summary:

```
Console X
<terminated> WorkSheet4 [Java Application] C:\Program Files\Java\jdk-18.0.2.1\bin\javaw.exe (04-Oct-2022, 10:39:32 am – 10:39:32 am) [pid: 12548]

mySet1: [A, B, C]

list: [A, A, B, C, B]

mySet2: [A, B, C]

MySet1 matches mySet2: true

mySet2: [B, C]

MySet1 matches mySet2: false

MySet1 contains all the elements: true

MySet2 contains all the elements: false

Iterator Implementation
Iterator loop: A
Iterator loop: B
Iterator loop: C

For loop Implementation
For each loop A
For each loop B
For each loop C

mySet1 is Empty: true
mySet1 has: 0 Elements
mySet2 has: 2 Elements
The array:[null, null]
```

### Learning outcomes (What I have learnt):

1. Learnt How to create the HashSet and insert the values to it.
2. Set manipulation concept understood.
3. Created list and Imported list in to a set.
4. Learnt the concept of Iterator.
5. Learnt concept of Set to Array Conversion

**Evaluation Grid (To be created per the faculty's SOP and Assessment guidelines):**

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.	Worksheet completion including writing learning objectives/Outcomes. (To be submitted at the end of the day).		
2.	Post-Lab Quiz Result.		
3.	Student Engagement in Simulation/Demonstration/Performance and Controls/Pre-Lab Questions.		
	Signature of Faculty (with Date):	Total Marks Obtained:	

## LAB INDEX

NAME: Vivek Kumar

SUBJECTNAME: Project Based Learning in Java Lab

UID: 21BCS8129

SUBJECTCODE: 20CSP-314

SECTION: WM-20BCS-616/A

Sr. No	Program	Date	Evaluation				Sign
			LW (12)	VV (10)	FW (8)	Total (30)	
1	Create an application to save the employee information using arrays.	09-08-2022					
2	Design and implement a simple inventory control system for a small video rentalstore.	23-08-2022					
3	Create a application to calculate interest for FDs, RDs based on certain conditions using inheritance.	02-09-2022					
4	Create a program to show the usage of Sets of Collection interface.	27-09-2022					
5	Create a program to set view of Keys from Java Hashtable.	27-09-2022					
6	Write a Program to perform the basic operations like insert, delete, display and search in list. List contains String object items where these operations are to be performed.						
7	Create a menu based Java application with the following options.1.Add an Employee2.Display All3.Exit If option 1 is selected, the application should gather details of the employee like employee name, employee id, designation and salary and store it in a file. If option 2 is selected, the application should display all the employee details. If option 3 is selected the application should exit.						
8	Create a palindrome creator application for making a longest possible palindrome out of given input string.						
9	Create a Servlet/ application with a facility to print any message on web browser.						
10	Create JSP application for addition, multiplication and division.						

**CHANDIGARH UNIVERSITY  
UNIVERSITY INSTITUTE OF NGINEERING  
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**



<b>Submitted By:</b> Vivek Kumar(21BCS8129)		<b>Submitted To:</b> Neeru Sharma(E12950)	
<b>Subject Name</b>	Project Based Learning in Java Lab		
<b>Subject Code</b>	20CSP-321		
<b>Branch</b>	Computer Science and Engineering		
<b>Semester</b>	5 <sup>th</sup>		

## Experiment - 5

**Student Name: Vivek Kumar****Branch: BE-CSE(LEET)****Semester: 5<sup>th</sup>****Subject Name: Project Based Learning in Java Lab****UID: 21BCS8129****Section/Group: 20BCS-WM-616/A****Date of Performance: 27/09/2022****Subject Code: 20CSP-321**

### 1. Aim/Overview of the practical:

Create a program to set view of Keys from Java Hashtable.

### 2. Task to be done/ Which logistics used:

Write the program to create an application to perform operation on Hashtable.

### 3. Software Requirements (For programming-based labs):

- JDK-8 or any
- Eclipse-IDE for Java

### 4. Steps for experiment/practical/Code:

```
package unit2;
```

```
import java.util.Enumeration;
```

```
import java.util.Hashtable;
```

```
public class WorkSheet5 {
```

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

```
        // TODO Auto-generated method stub
```

```
        Hashtable<Integer, String> ht = new Hashtable<>();
```

```
        ht.put(1, "Vivek");
```

```
        ht.put(2, "Abhi");
```

```
        ht.put(3, "Amar");
```

```
        ht.put(4, "Punya");
```

```
        ht.put(5, "Vikash");
```

```
        ht.put(6, "Ruh");
```

```
        Enumeration<Integer> keys = ht.keys();
```

```
        Enumeration<String> values = ht.elements();
```

```
        while(keys.hasMoreElements() && values.hasMoreElements()){
```

```
            System.out.println("Key: "+keys.nextElement()+"\tValue: "+values.nextElement());
```

```
        }
```

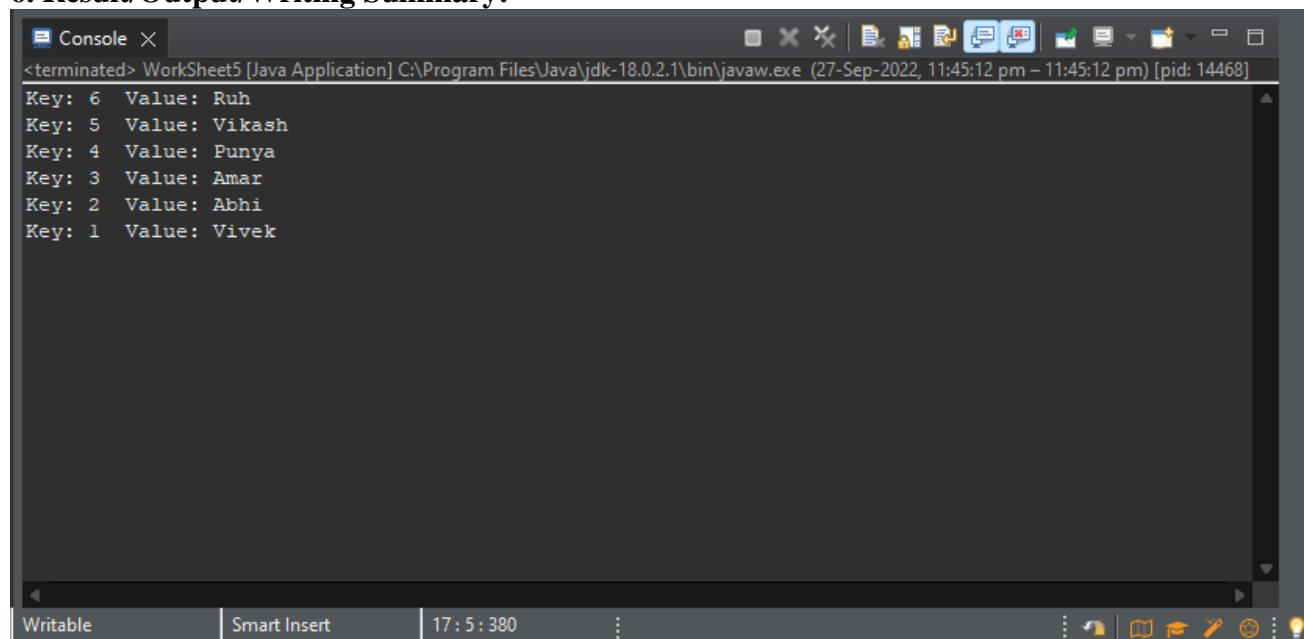
```
    }
```

```
}
```

### 5. Observations/Discussions/ Complexity Analysis:

Here we have created the Hashtable and Enumeration and Data inserted to Hashtable with the key pairs, performed display operation.

## 6. Result/Output/Writing Summary:



```
<terminated> WorkSheet5 [Java Application] C:\Program Files\Java\jdk-18.0.2.1\bin\javaw.exe (27-Sep-2022, 11:45:12 pm - 11:45:12 pm) [pid: 14468]
Key: 6 Value: Ruh
Key: 5 Value: Vikash
Key: 4 Value: Punya
Key: 3 Value: Amar
Key: 2 Value: Abhi
Key: 1 Value: Vivek
```

### Learning outcomes (What I have learnt):

1. Learnt How to create the Hashtable and insert the values to it.
2. Hashtable manipulation concept understood.
3. Learnt the concept of Enumeration.
4. Learnt concept of while and Hashtable iteration.
5. Understood the concept of AND operator '&&'.

### Evaluation Grid (To be created per the faculty's SOP and Assessment guidelines):

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.	Worksheet completion including writing learning objectives/Outcomes. (To be submitted at the end of the day).		
2.	Post-Lab Quiz Result.		
3.	Student Engagement in Simulation/Demonstration/Performance and Controls/Pre-Lab Questions.		
	Signature of Faculty (with Date):	Total Marks Obtained:	

## LAB INDEX

NAME: Vivek Kumar

SUBJECTNAME: Project Based Learning in Java Lab

UID: 21BCS8129

SUBJECTCODE: 20CSP-314

SECTION: WM-20BCS-616/A

Sr. No	Program	Date	Evaluation				Sign
			LW (12)	VV (10)	FW (8)	Total (30)	
1	Create an application to save the employee information using arrays.	09-08-2022					
2	Design and implement a simple inventory control system for a small video rentalstore.	23-08-2022					
3	Create a application to calculate interest for FDs, RDs based on certain conditions using inheritance.	02-09-2022					
4	Create a program to show the usage of Sets of Collection interface.	27-09-2022					
5	Create a program to set view of Keys from Java Hashtable.	27-09-2022					
6	Write a Program to perform the basic operations like insert, delete, display and search in list. List contains String object items where these operations are to be performed.	04-10-2022					
7	Create a menu based Java application with the following options.1.Add an Employee2.Display All3.Exit If option 1 is selected, the application should gather details of the employee like employee name, employee id, designation and salary and store it in a file. If option 2 is selected, the application should display all the employee details. If option 3 is selected the application should exit.						
8	Create a palindrome creator application for making a longest possible palindrome out of given input string.						
9	Create a Servlet/ application with a facility to print any message on web browser.						
10	Create JSP application for addition, multiplication and division.						



**CHANDIGARH UNIVERSITY  
UNIVERSITY INSTITUTE OF NGINEERING  
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**



<b>Submitted By:</b> Vivek Kumar(21BCS8129)		<b>Submitted To:</b> Neeru Sharma(E12950)	
<b>Subject Name</b>	Project Based Learning in Java Lab		
<b>Subject Code</b>	20CSP-321		
<b>Branch</b>	Computer Science and Engineering		
<b>Semester</b>	5 <sup>th</sup>		



## Experiment - 6

**Student Name: Vivek Kumar****Branch: BE-CSE(LEET)****Semester: 5<sup>th</sup>****Subject Name: Project Based Learning in Java Lab****UID: 21BCS8129****Section/Group: 20BCS-WM-616/A****Date of Performance: 27/09/2022****Subject Code: 20CSP-321**

### 1. Aim/Overview of the practical:

Write a Program to perform the basic operations like insert, delete, display and search in list. List contains String object items where these operations are to be performed.

### 2. Task to be done/ Which logistics used:

Write the program to create an application to perform a List manipulation.

### 3. Software Requirements (For programming-based labs):

- JDK-8 or any
- Eclipse-IDE for Java

### 4. Steps for experiment/practical/Code:

```
package unit2;
```

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

```
import java.util.Scanner;
```

```
public class WorkSheet6 {  
    public static List<String> list=new ArrayList<String>();  
    void addItem(String item) {  
        list.add(item);  
    }  
    void displayItem() {  
        if(list.size()>0) {  
            for(String name:list)  
                System.out.println(name);  
        }else  
            System.out.println("List is Empty");  
    }  
    void searchItem(String item) {  
        if(list.size()>0) {  
            if(list.contains(item))  
                System.out.println(item+" is Present");  
            else  
                System.out.println(item+" is not Present");  
        }else  
            System.out.println("List is Empty");  
    }  
}
```

```
}  
void deleteItem(String item) {  
    if(list.size()>0) {  
        if(list.contains(item)) {  
            list.remove(item);  
            System.out.println(item+" is removed");  
        }else  
            System.out.println(item+" is not Present");  
    }else  
        System.out.println("List is Empty");  
}
```

```
public static void main(String[] args) {  
    WorkSheet6 obj = new WorkSheet6();  
    boolean flag=true;  
    String item;  
    int choice;  
    Scanner in = new Scanner(System.in);  
    while(flag) {  
        System.out.println("\nMAIN MENU");  
        System.out.println("*****");  
        System.out.println("1.Insert:");  
        System.out.println("2.Search:");  
        System.out.println("3.Delete:");  
        System.out.println("4.Display:");  
        System.out.println("5.Exit");  
        System.out.println("\nEnter your choice:");  
        choice = in.nextInt();  
        switch(choice)  
        {  
            case 1:  
            {  
                System.out.println("Enter the item: ");  
                item = in.next();  
                obj.addItem(item);  
                break;  
            }  
            case 2:  
            {  
                System.out.println("Enter the item: ");
```

```
        item = in.next();
        obj.searchItem(item);
    break;
    }
    case 3:
    {
        System.out.println("Enter the item: ");
        item = in.next();
        obj.deleteItem(item);
    break;
    }
    case 4:
    {
        System.out.println("\nElement of Lists are: ");
        obj.displayItem();
    break;
    }
    case 5:
    {
        System.out.println("Exiting...!! Thanks for using the application");
        flag=false;
        break;
    }
    default:
    {
        System.out.println("Wrong input!!");
    }
}
}
in.close();
}
```

### 5. Observations/Discussions/ Complexity Analysis:

Here we have created the List, and performed all the operation of list such as insertion, searching, deletion, and traversal.

## 6. Result/Output/Writing Summary:

```
Console X
WorkSheet6 [Java Application] C:\Program Files\Java\jdk-18.0.2.1\bin\javaw.exe (06-Oct-2022, 7:53:01 pm) [pid: 2552]

MAIN MENU
*****
1.Insert:
2.Search:
3.Delete:
4.Display:
5.Exit

Enter your choice:
1
Enter the item:
Vivek

MAIN MENU
*****
1.Insert:
2.Search:
3.Delete:
4.Display:
5.Exit

Enter your choice:
1
Enter the item:
Amar

MAIN MENU
*****
1.Insert:
2.Search:
3.Delete:
4.Display:
5.Exit

Enter your choice:
1
Enter the item:
Abhi

MAIN MENU
*****
1.Insert:
2.Search:
3.Delete:
4.Display:
5.Exit

Enter your choice:
```

```
Console X
WorkSheet6 [Java Application] C:\Program Files\Java\jdk-18.0.2.1\bin\javaw.exe (06-Oct-2022, 7:53:01 pm) [pid: 2552]

MAIN MENU
*****
1.Insert:
2.Search:
3.Delete:
4.Display:
5.Exit

Enter your choice:
4

Element of Lists are:
Vivek
Amar
Abhi

MAIN MENU
*****
1.Insert:
2.Search:
3.Delete:
4.Display:
5.Exit

Enter your choice:
2
Enter the item:
Amar
Amar is Present

MAIN MENU
*****
1.Insert:
2.Search:
3.Delete:
4.Display:
5.Exit

Enter your choice:
3
Enter the item:
Amar
Amar is removed

MAIN MENU
*****
1.Insert:
2.Search:
3.Delete:
4.Display:
5.Exit

Enter your choice:
4

Element of Lists are:
Vivek
Abhi

MAIN MENU
*****
1.Insert:
2.Search:
3.Delete:
4.Display:
5.Exit

Enter your choice:
5
Exiting...!! Thanks for using the application
```

**Learning outcomes (What I have learnt):**

1. Learnt while loop.
2. List manipulation concept understood.
3. Created list and performed all operation of list.
4. Learnt the concept of switch concept.
5. Learnt concept of inbuilt function in list.

**Evaluation Grid (To be created per the faculty's SOP and Assessment guidelines):**

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.	Worksheet completion including writing learning objectives/Outcomes. (To be submitted at the end of the day).		
2.	Post-Lab Quiz Result.		
3.	Student Engagement in Simulation/Demonstration/Performance and Controls/Pre-Lab Questions.		
	Signature of Faculty (with Date):	Total Marks Obtained:	

## LAB INDEX

NAME: Vivek Kumar

SUBJECTNAME: Project Based Learning in Java Lab

UID: 21BCS8129

SUBJECTCODE: 20CSP-314

SECTION: WM-20BCS-616/A

Sr. No	Program	Date	Evaluation				Sign
			LW (12)	VV (10)	FW (8)	Total (30)	
1	Create an application to save the employee information using arrays.	09-08-2022					
2	Design and implement a simple inventory control system for a small video rentalstore.	23-08-2022					
3	Create a application to calculate interest for FDs, RDs based on certain conditions using inheritance.	02-09-2022					
4	Create a program to show the usage of Sets of Collection interface.	27-09-2022					
5	Create a program to set view of Keys from Java Hashtable.	27-09-2022					
6	Write a Program to perform the basic operations like insert, delete, display and search in list. List contains String object items where these operations are to be performed.	04-10-2022					
7	Create a menu based Java application with the following options.1.Add an Employee2.Display All3.Exit If option 1 is selected, the application should gather details of the employee like employee name, employee id, designation and salary and store it in a file. If option 2 is selected, the application should display all the employee details. If option 3 is selected the application should exit.	14-10-2022					
8	Create a palindrome creator application for making a longest possible palindrome out of given input string.						
9	Create a Servlet/ application with a facility to print any message on web browser.						
10	Create JSP application for addition, multiplication and division.						

**CHANDIGARH UNIVERSITY  
UNIVERSITY INSTITUTE OF NGINEERING  
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**



<b>Submitted By:</b> Vivek Kumar(21BCS8129)		<b>Submitted To:</b> Neeru Sharma(E12950)
<b>Subject Name</b>	Project Based Learning in Java Lab	
<b>Subject Code</b>	20CSP-321	
<b>Branch</b>	Computer Science and Engineering	
<b>Semester</b>	5 <sup>th</sup>	



## Experiment - 7

**Student Name: Vivek Kumar****Branch: BE-CSE(LEET)****Semester: 5<sup>th</sup>****Subject Name: Project Based Learning in Java Lab****UID: 21BCS8129****Section/Group: 20BCS-WM-616/A****Date of Performance: 14/10/2022****Subject Code: 20CSP-321**

### 1. Aim/Overview of the practical:

Create a menu-based Java application with the following options. 1. Add an Employee 2. Display All 3. Exit  
If option 1 is selected, the application should gather details of the employee like employee name, employee id, designation and salary and store it in a file. If option 2 is selected, the application should display all the employee details. If option 3 is selected the application should exit.

### 2. Task to be done/ Which logistics used:

Write the program to create an application to perform a File manipulation.

### 3. Software Requirements (For programming-based labs):

- JDK-8 or any
- Eclipse-IDE for Java

### 4. Steps for experiment/practical/Code:

```
package unit2;
```

```
import java.io.File;  
import java.io.FileInputStream;  
import java.io.FileOutputStream;  
import java.io.IOException;  
import java.io.ObjectInputStream;  
import java.io.ObjectOutputStream;  
import java.io.Serializable;  
import java.util.ArrayList;  
import java.util.Scanner;
```

```
@SuppressWarnings("serial")  
class Employee implements Serializable {  
    int id;  
    String name;  
    float salary;  
    long contact_no;  
    String email_id;
```

```
    public Employee(  
        int id,  
        String name,  
        float salary,  
        long contact_no,  
        String email_id
```

```
) {
    this.id = id;
    this.name = name;
    this.salary = salary;
    this.contact_no = contact_no;
    this.email_id = email_id;
}

public String toString() {
    return (
        "Employee Details :\n" +
        " ID: " +
        this.id +
        " Name: " +
        this.name +
        " Salary: " +
        this.salary +
        " Contact No: " +
        this.contact_no +
        " Email-id: " +
        this.email_id
    );
}

}

public class WorkSheet7 {

    static void display(ArrayList<Employee> al) {
        System.out.println("\n-----Employee List-----\n");
        System.out.println(
            String.format(
                "%-10s%-15s%-10s%-20s%-10s",
                "ID",
                "Name",
                "salary",
                "contact-no",
                "Email-Id"
            )
        );
        for (Employee e : al) {
            System.out.println(
                String.format(
                    "%-10s%-15s%-10s%-15s%-10s",
                    e.id,
                    e.name,
                    e.salary,
```

```
        e.contact_no,  
        e.email_id  
    )  
    );  
}  
}
```

```
@SuppressWarnings("unchecked")  
public static void main(String[] args) {  
    int id;  
    String name;  
    float salary;  
    long contact_no;  
    String email_id;
```

```
    Scanner sc = new Scanner(System.in);  
    ArrayList<Employee> al = new ArrayList<Employee>();
```

```
    File f = null;  
    FileInputStream fis = null;  
    ObjectInputStream ois = null;  
    FileOutputStream fos = null;  
    ObjectOutputStream oos = null;  
    try {  
        f =  
            new File(  
                "R:\\VnjVibhash\\Assignments\\CU-Assignments\\5th  
Sem\\Java\\JavaLab\\src\\EmployeeDataList.txt"  
            );  
        if (f.exists()) {  
            fis = new FileInputStream(f);  
            ois = new ObjectInputStream(fis);  
            al = (ArrayList<Employee>) ois.readObject();  
        }  
    } catch (Exception exp) {  
        System.out.println(exp);  
    }  
    do {  
        System.out.println(  
            "\n*****Welcome to the Employee Management System*****\n"  
        );  
        System.out.println(  
            "1). Add Employee to the DataBase\n" +  
            "2). Search for Employee\n" +  
            "3). Edit Employee details\n" +  
            "4). Delete Employee Details\n" +
```

```
"5). Display all Employees working in this company\n" +
"6). EXIT\n"
);
System.out.println("Enter your choice : ");
int ch = sc.nextInt();

switch (ch) {
case 1:
    System.out.println("\nEnter the following details to ADD list:\n");
    System.out.println("Enter ID :");
    id = sc.nextInt();
    System.out.println("Enter Name :");
    name = sc.next();
    System.out.println("Enter Salary :");
    salary = sc.nextFloat();
    System.out.println("Enter Contact No :");
    contact_no = sc.nextLong();
    System.out.println("Enter Email-ID :");
    email_id = sc.next();
    al.add(new Employee(id, name, salary, contact_no, email_id));
    display(al);
    break;
case 2:
    System.out.println("Enter the Employee ID to search :");
    id = sc.nextInt();
    int i = 0;
    for (Employee e : al) {
        if (id == e.id) {
            System.out.println(e + "\n");
            i++;
        }
    }
    if (i == 0) {
        System.out.println(
            "\nEmployee Details are not available, Please enter a valid ID!!"
        );
    }
    break;
case 3:
    System.out.println("\nEnter the Employee ID to EDIT the details");
    id = sc.nextInt();
    int j = 0;
    for (Employee e : al) {
        if (id == e.id) {
            j++;
            do {
```

```
int ch1 = 0;
System.out.println(
    "\nEDIT Employee Details :\n" +
    "1). Employee ID\n" +
    "2). Name\n" +
    "3). Salary\n" +
    "4). Contact No.\n" +
    "5). Email-ID\n" +
    "6). GO BACK\n"
);
System.out.println("Enter your choice : ");
ch1 = sc.nextInt();
switch (ch1) {
    case 1:
        System.out.println("\nEnter new Employee ID:");
        e.id = sc.nextInt();
        System.out.println(e + "\n");
        break;
    case 2:
        System.out.println("Enter new Employee Name:");
        e.name = sc.nextLine();
        System.out.println(e + "\n");
        break;
    case 3:
        System.out.println("Enter new Employee Salary:");
        e.salary = sc.nextFloat();
        System.out.println(e + "\n");
        break;
    case 4:
        System.out.println("Enter new Employee Contact No. :");
        e.contact_no = sc.nextLong();
        System.out.println(e + "\n");
        break;
    case 5:
        System.out.println("Enter new Employee Email-ID :");
        e.email_id = sc.next();
        System.out.println(e + "\n");
        break;
    case 6:
        j++;
        break;
    default:
        System.out.println(
            "\nEnter a correct choice from the List : "
        );
        break;
}
```

```
        }
    } while (j == 1);
}
}
if (j == 0) {
    System.out.println(
        "\nEmployee Details are not available, Please enter a valid ID!!"
    );
}

break;
case 4:
    System.out.println(
        "\nEnter Employee ID to DELETE from the Databse : "
    );
    id = sc.nextInt();
    int k = 0;
    try {
        for (Employee e : al) {
            if (id == e.id) {
                al.remove(e);
                display(al);
                k++;
            }
        }
        if (k == 0) {
            System.out.println(
                "\nEmployee Details are not available, Please enter a valid ID!!"
            );
        }
    } catch (Exception ex) {
        System.out.println(ex);
    }
    break;
case 5:
    try {
        al = (ArrayList<Employee>) ois.readObject();
    } catch (ClassNotFoundException e2) {
        System.out.println(e2);
    } catch (Exception e2) {
        System.out.println(e2);
    }
    display(al);
    break;
case 6:
    try {
```

```
fos = new FileOutputStream(f);
oos = new ObjectOutputStream(fos);
oos.writeObject(al);
} catch (IOException e1) {
    e1.printStackTrace();
} catch (Exception e2) {
    e2.printStackTrace();
} finally {
    try {
        fis.close();
        ois.close();
        fos.close();
        oos.close();
    } catch (Exception e1) {
        e1.printStackTrace();
    }
}
System.out.println(
    "\nYou have chosen EXIT !! Saving Files and closing the tool."
);
sc.close();
System.exit(0);
break;
default:
    System.out.println("\nEnter a correct choice from the List :");
    break;
}
} while (true);
}
}
```

### 5. Observations/Discussions/ Complexity Analysis:

Here we have created the File objects, and performed all the operation of file such as insertion, searching, deletion, and finally stored in the File.

## 6. Result/Output/Writing Summary:

```

Console X
WorkSheet7 [Java Application] C:\Program Files\Java\jdk-18.0.2.1\bin\javaw.exe (19-Oct-2022, 8:28:48 pm) [pid: 11068]

*****Welcome to the Employee Management System*****

1). Add Employee to the DataBase
2). Search for Employee
3). Edit Employee details
4). Delete Employee Details
5). Display all Employees working in this company
6). EXIT

Enter your choice :
1

Enter the following details to ADD list:

Enter ID :
104
Enter Name :
Manish
Enter Salary :
55000
Enter Contact No :
8877665544
Enter Email-ID :
manish@gmail.com

-----Employee List-----

ID      Name      salary    contact-no    Email-Id
103     Vivek     100000.0  7762974716   vnjvibhash@gmail.com
102     Amarjeet  100000.0  8765439277   amarjeetkr@gmail.com
104     Manish    55000.0   8877665544   manish@gmail.com

*****Welcome to the Employee Management System*****

1). Add Employee to the DataBase
2). Search for Employee
3). Edit Employee details
4). Delete Employee Details
5). Display all Employees working in this company
6). EXIT

Enter your choice :

```



```
Console X
WorkSheet7 [Java Application] C:\Program Files\Java\jdk-18.0.2.1\bin\javaw.exe (19-Oct-2022, 8:28:48 pm) [pid: 11068]
Enter your choice :
3

Enter the Employee ID to EDIT the details
104

EDIT Employee Details :
1). Employee ID
2). Name
3). Salary
4). Contact No.
5). Email-ID
6). GO BACK

Enter your choice :
1

Enter new Employee ID:
101
Employee Details :
ID: 101 Name: Manish Salary: 55000.0 Contact No: 8877665544 Email-id: manish@gmail.com

EDIT Employee Details :
1). Employee ID
2). Name
3). Salary
4). Contact No.
5). Email-ID
6). GO BACK

Enter your choice :
6

*****Welcome to the Employee Management System*****

1). Add Employee to the DataBase
2). Search for Employee
3). Edit Employee details
4). Delete Employee Details
5). Display all Employees working in this company
6). EXIT

Enter your choice :
```

```

Console X
WorkSheet7 [Java Application] C:\Program Files\Java\jdk-18.0.2.1\bin\javaw.exe (19-Oct-2022, 8:28:48 pm) [pid: 11068]

Enter your choice :
2
Enter the Employee ID to search :
102
Employee Details :
ID: 102 Name: Amarjeet Salary: 100000.0 Contact No: 8765439277 Email-id: amarjeetkr@gmail.com

*****Welcome to the Employee Management System*****

1). Add Employee to the DataBase
2). Search for Employee
3). Edit Employee details
4). Delete Employee Details
5). Display all Employees working in this company
6). EXIT

Enter your choice :
4

Enter Employee ID to DELETE from the Database :
103

-----Employee List-----

ID      Name      salary    contact-no    Email-Id
102     Amarjeet    100000.0   8765439277    amarjeetkr@gmail.com
101     Manish      55000.0    8877665544    manish@gmail.com
java.util.ConcurrentModificationException

*****Welcome to the Employee Management System*****

1). Add Employee to the DataBase
2). Search for Employee
3). Edit Employee details
4). Delete Employee Details
5). Display all Employees working in this company
6). EXIT

Enter your choice :
5
java.io.EOFException

-----Employee List-----

ID      Name      salary    contact-no    Email-Id
102     Amarjeet    100000.0   8765439277    amarjeetkr@gmail.com
101     Manish      55000.0    8877665544    manish@gmail.com

```

```
*****Welcome to the Employee Management System*****

1). Add Employee to the DataBase
2). Search for Employee
3). Edit Employee details
4). Delete Employee Details
5). Display all Employees working in this company
6). EXIT

Enter your choice :
6

You have chosen EXIT !! Saving Files and closing the tool.
```

## Stored File Output:

```
EmployeeDataList - Notepad
File Edit View

~i @sr @java.util.ArrayListx@0@Ca@ @I @sizep @w@ @sr @unit2.Employee-@é0:L@ @J
contact_noI @idF @salaryL @email_idt @Ljava/lang/String;L @nameq ~ @xp @
úý- fGÁP t @amarjeetkr@gmail.comt @Amarjeetsq ~ @ @&m@ eGVØ t @manish@gmail.comt @Manishx

Ln 1, Col 1 | 100% | Unix (LF) | ANSI
```

## Learning outcomes (What I have learnt):

1. Learnt while loop.
2. File manipulation concept understood.
3. Created file and performed all operation of file.
4. Learnt the concept of switch concept.
5. Learnt concept of inbuilt function in file such as FileOutputStream & FileInputStream.

**Evaluation Grid (To be created per the faculty's SOP and Assessment guidelines):**

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.	Worksheet completion including writing learning objectives/Outcomes. (To be submitted at the end of the day).		
2.	Post-Lab Quiz Result.		
3.	Student Engagement in Simulation/Demonstration/Performance and Controls/Pre-Lab Questions.		
	Signature of Faculty (with Date):	Total Marks Obtained:	

## LAB INDEX

NAME: Vivek Kumar

SUBJECTNAME: Project Based Learning in Java Lab

UID: 21BCS8129

SUBJECTCODE: 20CSP-314

SECTION: WM-20BCS-616/A

Sr. No	Program	Date	Evaluation				Sign
			LW (12)	VV (10)	FW (8)	Total (30)	
1	Create an application to save the employee information using arrays.	09-08-2022					
2	Design and implement a simple inventory control system for a small video rentalstore.	23-08-2022					
3	Create a application to calculate interest for FDs, RDs based on certain conditions using inheritance.	02-09-2022					
4	Create a program to show the usage of Sets of Collection interface.	27-09-2022					
5	Create a program to set view of Keys from Java Hashtable.	27-09-2022					
6	Write a Program to perform the basic operations like insert, delete, display and search in list. List contains String object items where these operations are to be performed.	04-10-2022					
7	Create a menu-based Java application with the following options.1. Add an Employee2.Display All3.Exit If option 1 is selected, the application should gather details of the employee like employee name, employee id, designation and salary and store it in a file. If option 2 is selected, the application should display all the employee details. If option 3 is selected the application should exit.	14-10-2022					
8	Create a palindrome creator application for making a longest possible palindrome out of given input string.	01-11-2022					
9	Create a Servlet/ application with a facility to print any message on web browser.						
10	Create JSP application for addition, multiplication and division.						



**CHANDIGARH UNIVERSITY  
UNIVERSITY INSTITUTE OF NGINEERING  
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**



<b>Submitted By:</b> Vivek Kumar(21BCS8129)		<b>Submitted To:</b> Neeru Sharma(E12950)	
<b>Subject Name</b>	Project Based Learning in Java Lab		
<b>Subject Code</b>	20CSP-321		
<b>Branch</b>	Computer Science and Engineering		
<b>Semester</b>	5 <sup>th</sup>		

## Experiment - 8

**Student Name: Vivek Kumar****Branch: BE-CSE(LEET)****Semester: 5<sup>th</sup>****Subject Name: Project Based Learning in Java Lab****UID: 21BCS8129****Section/Group: 20BCS-WM-616/A****Date of Performance: 01/11/2022****Subject Code: 20CSP-321**

### 1. Aim/Overview of the practical:

Create a palindrome creator application for making a longest possible palindrome out of given input string.

### 2. Task to be done/ Which logistics used:

Write the program to create an application to form a largest palindrome from given String.

### 3. Software Requirements (For programming-based labs):

- JDK-8 or any
- Eclipse-IDE for Java

### 4. Steps for experiment/practical/Code:

package unit2;

import java.util.HashMap;

import java.util.Map;

import java.util.Scanner;

public class WorkSheet8 {

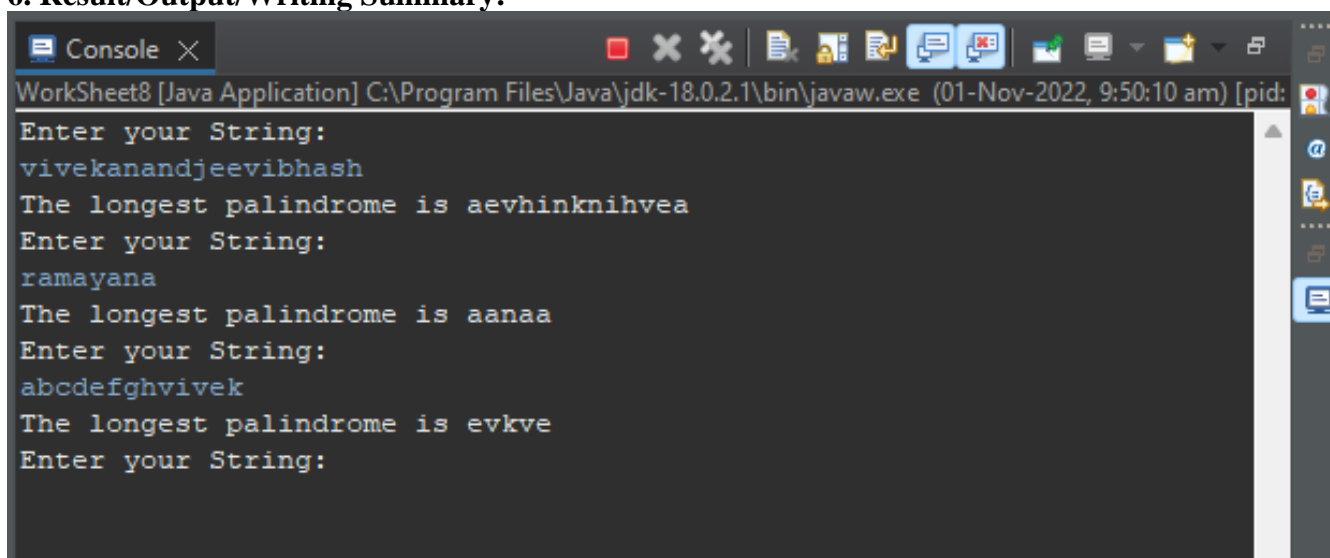
```
    public static String longestPalindrome(String str) {
        if (str == null || str.length() == 0) {
            return str;
        }
        Map<Character, Integer> freq = new HashMap<>();
        for (char ch: str.toCharArray()) {
            freq.put(ch, freq.getOrDefault(ch, 0) + 1);
        }
        String mid_char = "";
        StringBuilder left = new StringBuilder();
        for (var entry: freq.entrySet()){
            char ch = entry.getKey();
            int count = entry.getValue();
            if (count % 2 == 1) {
                mid_char = String.valueOf(ch);
            }
            left.append(String.valueOf(ch).repeat(count / 2));
        }
        StringBuilder right = new StringBuilder(left).reverse();
        return ("" + left + mid_char + right);
    }
```

```
public static void main(String args[]) {  
    Scanner in = new Scanner(System.in);  
    System.out.println("Enter your String: ");  
    String str = in.next();  
    System.out.println("The longest palindrome is " + longestPalindrome(str));  
    in.close();  
}  
  
}
```

### 5. Observations/Discussions/ Complexity Analysis:

Here we have created the palindrome function to performed an operation on a String to create a largest possible palindrome.

### 6. Result/Output/Writing Summary:



```
WorkSheet8 [Java Application] C:\Program Files\Java\jdk-18.0.2.1\bin\javaw.exe (01-Nov-2022, 9:50:10 am) [pid:  
Enter your String:  
vivekanandjeevibhash  
The longest palindrome is aevhinknihvea  
Enter your String:  
ramayana  
The longest palindrome is aanaa  
Enter your String:  
abcdefghijklhvivek  
The longest palindrome is evkve  
Enter your String:
```

### Learning outcomes (What I have learnt):

1. Learnt the concept of palindrome.
2. Learnt the concept of StringBuilder ().
3. Learnt the concept of HashMap ().
4. Learnt the concept of StringBuilder Manipulation such as Reverse.
5. Successfully executed the code and completed the Worksheet.



**Evaluation Grid (To be created per the faculty's SOP and Assessment guidelines):**

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.	Worksheet completion including writing learning objectives/Outcomes. (To be submitted at the end of the day).		
2.	Post-Lab Quiz Result.		
3.	Student Engagement in Simulation/Demonstration/Performance and Controls/Pre-Lab Questions.		
	Signature of Faculty (with Date):	Total Marks Obtained:	

## LAB INDEX

NAME: Vivek Kumar

SUBJECTNAME: Project Based Learning in Java Lab

UID: 21BCS8129

SUBJECTCODE: 20CSP-314

SECTION: WM-20BCS-616/A

Sr. No	Program	Date	Evaluation				Sign
			LW (12)	VV (10)	FW (8)	Total (30)	
1	Create an application to save the employee information using arrays.	09-08-2022					
2	Design and implement a simple inventory control system for a small video rentalstore.	23-08-2022					
3	Create a application to calculate interest for FDs, RDs based on certain conditions using inheritance.	02-09-2022					
4	Create a program to show the usage of Sets of Collection interface.	27-09-2022					
5	Create a program to set view of Keys from Java Hashtable.	27-09-2022					
6	Write a Program to perform the basic operations like insert, delete, display and search in list. List contains String object items where these operations are to be performed.	04-10-2022					
7	Create a menu-based Java application with the following options.1. Add an Employee2.Display All3.Exit If option 1 is selected, the application should gather details of the employee like employee name, employee id, designation and salary and store it in a file. If option 2 is selected, the application should display all the employee details. If option 3 is selected the application should exit.	14-10-2022					
8	Create a palindrome creator application for making a longest possible palindrome out of given input string.	01-11-2022					
9	Create a Servlet/ application with a facility to print any message on web browser.	10/11/2022					
10	Create JSP application for addition, multiplication and division.						



**CHANDIGARH UNIVERSITY  
UNIVERSITY INSTITUTE OF NGINEERING  
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**



<b>Submitted By:</b> Vivek Kumar(21BCS8129)		<b>Submitted To:</b> Neeru Sharma(E12950)
<b>Subject Name</b>	Project Based Learning in Java Lab	
<b>Subject Code</b>	20CSP-321	
<b>Branch</b>	Computer Science and Engineering	
<b>Semester</b>	5 <sup>th</sup>	

## Experiment - 9

**Student Name: Vivek Kumar****Branch: BE-CSE(LEET)****Semester: 5<sup>th</sup>****Subject Name: Project Based Learning in Java Lab****UID: 21BCS8129****Section/Group: 20BCS-WM-616/A****Date of Performance: 01/11/2022****Subject Code: 20CSP-321**

### 1. Aim/Overview of the practical:

Create a program that uses JSP and html to insert, edit, delete or view employee data from a database. You have to use JSP bean class to set values for employee objects.

### 2. Task to be done/ Which logistics used:

Create a program that uses JSP and html to insert, edit, delete or view employee data from a database. You have to use JSP bean class to set values for employee objects.

### 3. Apparatus / Simulator Used:

- Eclipse IDE - (Java)
- NetBeans.
- JDK-8 or any.

### 4. Programs/ Code:

#### Index.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
<style>
    h1{
        margin: auto;
        width: 60%;
        padding-left: 100px;
    }
    body {
        background:orange;
        color: white;
    }
    .fall {
        background: grey;
        color: white;
        border: 2px solid blue;
        width: 45%;
        margin:auto;
    }
    .input-group{
        position: relative;
```

```

        margin: 23px;
        left: 175px;
    }
    button{
        position: relative;
        left: 35%
    }
    button:hover{
        cursor:pointer;
        background: black;
        color: white;
    }
    h3{
        margin: auto;
        width: 40%;
        padding: 5px;
    }
</style>
</head>
<body>
    <h1>Welcome the AP Auctions, Please Enter the Bid</h1>
    <form class="fall" name="register_form"
    action="<%=request.getContextPath()%>/functions" method="post" >
        <div class="input-group mb-2">
            <input type="text" name="name" id="name" onkeypress="return
            /[0-9a-zA-Z]/i.test(event.key)" class="form-control input_user" required="required"
            placeholder="Item ID">
        </div>
        <p class="error" id="name_id"></p>
        <div class="input-group mb-2">
            <div class="input-group-append">
                <span class="input-group-text"><i class="fas fa-
user"></i></span>
            </div>
            <input type="text" name="branch" id="branch" onkeypress="return
            /[0-9a-zA-Z]/i.test(event.key)" class="form-control input_user" required="required"
            placeholder="Item
Name">
        </div>
        <p class="error" id="branch_id"></p>
        <div class="input-group mb-2">
            <div class="input-group-append">
                <span class="input-group-text"><i class="fas fa-
user"></i></span>
            </div>
            <input type="text" name="course" id="course"
onkeypress="return /[0-9a-zA-Z]/i.test(event.key)" class="form-control input_user"
required="required"
            placeholder="Your
Name">
        </div>
        <p class="error" id="course_id"></p>
        <div class="input-group mb-2">

```

```

        <div class="input-group-append">
            <span class="input-group-text"><i class="fas fa-
user"></i></span>
        </div>
        <input type="email" name="email" id="email" onkeypress="return
/[0-9a-zA-Z.@_]/i.test(event.key)" class="form-control input_user" required="required"
placeholder="Email">
    </div>
    <p class="error" id="email_id"></p>

    <div class="input-group mb-2">
        <div class="input-group-append">
            <span class="input-group-text"><i class="fas fa-
user"></i></span>
        </div>
        <input type="number" id="number"
onKeyPress="if(this.value.length==10) return false;" name="number" class="form-control
input_user" required="required"
placeholder="Amount
Bid">
    </div>

    <p class="error"
id="number_id"></p>
    <button type="submit" onclick=hide()>Submit Bid</button>
</form>
<script>
document.getElementById("name").addEventListener("keypress", myname);
document.getElementById("name").addEventListener("mouseover", myname);
function myname() {
    var name= document.register_form.name.value;
    var paragraph = document.getElementById("name_id");
    if(name.length==0){
        paragraph.textContent="Please enter your Id !!";
        document.register_form.name.focus();
    }
    if(name.length>5 && name.length!=0){
        paragraph.textContent="\ID should not be greater than 5
character !!!";
        document.register_form.name.focus();
    }
    if(name.length<5){
        paragraph.textContent="";
    }
}

document.getElementById("email").addEventListener("keypress", myemail);
document.getElementById("email").addEventListener("mouseover", myemail);
function myemail() {
    var email= document.register_form.email.value;
    var paragraph = document.getElementById("email_id");
    if(email.length==0){
        paragraph.textContent="Please enter your email !!";
        document.register_form.email.focus();
    }
}

```

```

        else if(!(/^\w+([\.-]?\w+)*@\w+([\.-
]? \w+)*(\.\w{2,3})+$/).test(email))) {
            paragraph.textContent="Enter valid email !!!";
            document.register_form.email.focus();
        }
        else {
            paragraph.textContent="";
        }
    }

document.getElementById("branch").addEventListener("keypress", mybranch);
document.getElementById("branch").addEventListener("mouseover", mybranch);
function mybranch() {
    var branch= document.register_form.branch.value;
    var paragraph = document.getElementById("branch_id");
    if(branch.length<1){
        paragraph.textContent="Please enter your Item Name !!";
        document.register_form.branch.focus();
    }
    else {
        paragraph.textContent="";
    }
}

document.getElementById("course").addEventListener("keypress", mycourse);
document.getElementById("course").addEventListener("mouseover", mycourse);
function mycourse() {
    var course= document.register_form.course.value;
    var paragraph = document.getElementById("course_id");
    if(course.length<1){
        paragraph.textContent="Please enter your Name !!";
        document.register_form.course.focus();
    }
    else {
        paragraph.textContent="";
    }
}

document.getElementById("number").addEventListener("keypress", mynumber);
document.getElementById("number").addEventListener("mouseover", mynumber);
function mynumber() {
    var number= document.register_form.number.value;
    var paragraph = document.getElementById("number_id");
    if(number.length==0){
        paragraph.textContent="Please enter your Amount Bid !!!";
        document.register_form.email.focus();
    }
    else if(!(/^[6-9]\d{9}$/).test(number)){
        paragraph.textContent="Enter valid Amount Bid !!!";
        document.register_form.number.focus();
    }
    else {
        paragraph.textContent="";
    }
}

```



```

</script>
<h3>Bid Submitted</h3>
<h3>Your bid is now active. if your bid is successful, you will be notified within
24HRS</h3>
<h3> Item id:  <%=request.getAttribute("ans") %></h3> <br>
<h3> Item Name: <%=request.getAttribute("ans1") %></h3> <br/>
<h3> Your Name <%=request.getAttribute("ans2") %></h3><br/>
<h3> Email:<%=request.getAttribute("ans3") %></h3><br/>
<h3> Your Amount: <%=request.getAttribute("ans4") %></h3><br/>
</body>
</html>

```

### **Functions.java**

```

package unit3;
import java.util.*;
import java.io.IOException;
import java.io.PrintWriter;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 * Servlet implementation class functions
 */
@WebServlet(name="functions",urlPatterns={"/functions"})
public class functions extends HttpServlet {
    protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        String a=request.getParameter("name");
        String b=request.getParameter("branch");
        String c=request.getParameter("course");
        String d=request.getParameter("email");
        String e=request.getParameter("number");
        try {

            String ans=a;
            String ans1=b;
            String ans2=c;
            String ans3=d;

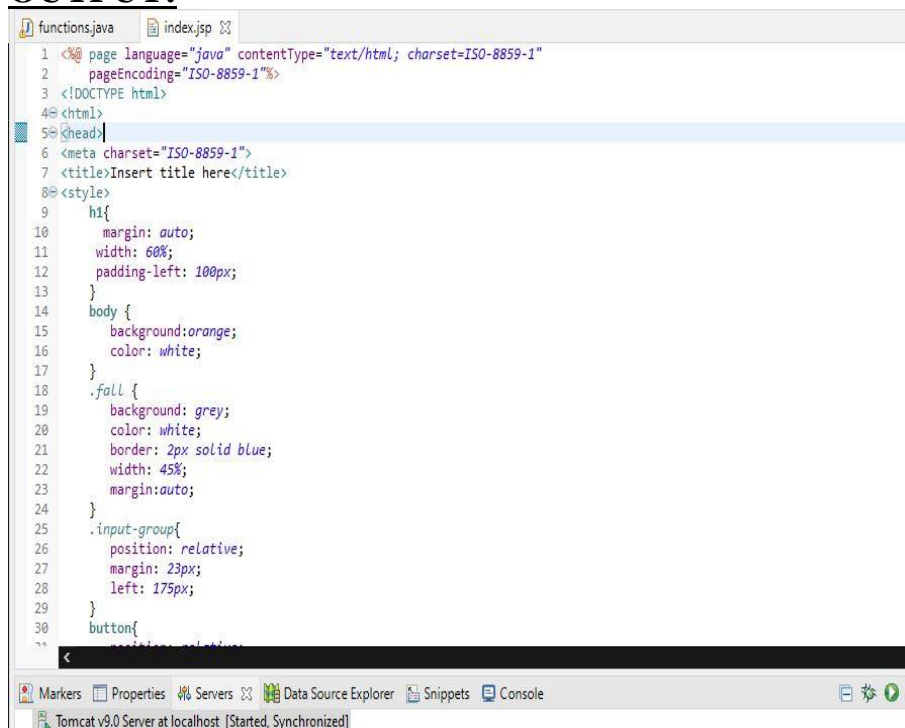
```



```
String ans4=e;
request.setAttribute("ans",ans);
request.setAttribute("ans1",ans1);
request.setAttribute("ans2",ans2);
request.setAttribute("ans3",ans3);
request.setAttribute("ans4",ans4);
```

```
request.getRequestDispatcher("index.jsp").forward(request,response);
} catch(Exception e1) {
    System.out.println(e1);
}
}
```

## OUTPUT:



```
1 <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
2   pageEncoding="ISO-8859-1"%>
3 <!DOCTYPE html>
4 <html>
5 <head>
6   <meta charset="ISO-8859-1">
7   <title>Insert title here</title>
8   <style>
9     h1{
10      margin: auto;
11      width: 60%;
12      padding-left: 100px;
13    }
14    body {
15      background:orange;
16      color: white;
17    }
18    .fall {
19      background: grey;
20      color: white;
21      border: 2px solid blue;
22      width: 45%;
23      margin:auto;
24    }
25    .input-group{
26      position: relative;
27      margin: 23px;
28      left: 175px;
29    }
30    button{
```

Item ID

Item Name

Please enter your Item Name !!

Your Name

Please enter your Name !!

Email

Please enter your email !!

Amount Bid

**Bid Submitted**

Your bid is now active. if your bid is successful, you will be notified within 24HRS

Item id: 1234

Item Name: rolex

Your Name: CHIRAG BITHER

Email: chiragbither02@gmail.com

Your Amount: 10000

## Learning outcomes (What I have learnt):

1. Learn About the servlet.
2. Learn about jsp and dynamic web project.
3. Learn about the tomcat server and its integrations with the java.

## Evaluation Grid (To be created per the faculty's SOP and Assessment guidelines):

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.	Worksheet completion including writing learning objectives/Outcomes. (To be submitted at the end of the day).		
2.	Post-Lab Quiz Result.		
3.	Student Engagement in Simulation/Demonstration/Performance and Controls/Pre-Lab Questions.		
	Signature of Faculty (with Date):	Total Marks Obtained:	

## LAB INDEX

NAME: Vivek Kumar

SUBJECTNAME: Project Based Learning in Java Lab

UID: 21BCS8129

SUBJECTCODE: 20CSP-314

SECTION: WM-20BCS-616/A

Sr. No	Program	Date	Evaluation				Sign
			LW (12)	VV (10)	FW (8)	Total (30)	
1	Create an application to save the employee information using arrays.	09-08-2022					
2	Design and implement a simple inventory control system for a small video rentalstore.	23-08-2022					
3	Create a application to calculate interest for FDs, RDs based on certain conditions using inheritance.	02-09-2022					
4	Create a program to show the usage of Sets of Collection interface.	27-09-2022					
5	Create a program to set view of Keys from Java Hashtable.	27-09-2022					
6	Write a Program to perform the basic operations like insert, delete, display and search in list. List contains String object items where these operations are to be performed.	04-10-2022					
7	Create a menu-based Java application with the following options.1. Add an Employee2.Display All3.Exit If option 1 is selected, the application should gather details of the employee like employee name, employee id, designation and salary and store it in a file. If option 2 is selected, the application should display all the employee details. If option 3 is selected the application should exit.	14-10-2022					
8	Create a palindrome creator application for making a longest possible palindrome out of given input string.	01-11-2022					
9	Create a Servlet/ application with a facility to print any message on web browser.	10/11/2022					
10	Create JSP application for addition, multiplication and division.	10/11/2022					



**CHANDIGARH UNIVERSITY  
UNIVERSITY INSTITUTE OF NGINEERING  
DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**



<b>Submitted By:</b> Vivek Kumar(21BCS8129)		<b>Submitted To:</b> Neeru Sharma(E12950)	
<b>Subject Name</b>	Project Based Learning in Java Lab		
<b>Subject Code</b>	20CSP-321		
<b>Branch</b>	Computer Science and Engineering		
<b>Semester</b>	5 <sup>th</sup>		

## Experiment - 10

**Student Name: Vivek Kumar**

**Branch: BE-CSE(LEET)**

**Semester: 5<sup>th</sup>**

**Subject Name: Project Based Learning in Java Lab**

**UID: 21BCS8129**

**Section/Group: 20BCS-WM-616/A**

**Date of Performance: 01/11/2022**

**Subject Code: 20CSP-321**

### 1. Aim/Overview of the practical:

Create a program that uses XML and html to create a DOM parser.

### 2. Task to be done/ Which logistics used:

Create a program that uses XML and html to create a DOM parser.

### 3. Apparatus / Simulator Used:

- Eclipse IDE - (Java)
- NetBeans.
- JDK-8 or any.

### 4. Programs/ Code:

**Code:**

**Index.jsp**

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Calculator</title>
<style>
body{
    background: black;
    color: white;
}
h1{
    text-align: center;
}
.Paramter{
    border: 2px solid white;background: blue;
    padding: 5px;
    max-width: 500px;
    margin: auto;
    font-size: 19px;
}
button{
    position: relative;
```

```

    left: 170px;
    margin: 10px; width: 60px; height: 30px;
    cursor: pointer; border-radius: 5px;
}
button:hover{
    background: orange;
}
</style>
</head>
<body>
    <br/>
    <div class="Paramter">
        <form name="funcitons"
action="<%=request.getContextPath()%>/functions" method="post" >
            <h1>Mathematical Operation</h1>
            <input type="radio" id="add" name="fun" value="+">
Addition <br/>
            <input type="radio" id="mul" name="fun" value="*">
Multiplication <br/>
            <input type="radio" id="sub" name="fun" value="-">
Subtraction <br/><br/>
            Enter the First Value: <input type="number"
name="fst"><br/><br/>
            Enter the Second Value: <input type="number"
name="snd"><br/>
            <button type="submit">Submit</button>
            <button value="Reset">Reset</button>
        </form>
        <h1>Ans = <%=request.getAttribute("ans") %></h1>
    </div>

</body>
</html>

```

### Functions.java

```

package unit3;

import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 * Servlet implementation class functions

```

```
*/
@WebServlet(name="functions",urlPatterns={"/functions"})
public class functions extends HttpServlet {
    protected void doPost(HttpServletRequest request,
        HttpServletResponse response) throws ServletException, IOException {
        String a=request.getParameter("fst");
        String b=request.getParameter("snd");
        String fun=request.getParameter("fun");
        try {
            System.out.println(a+fun+b);
            int i1=Integer.parseInt(a);
            int i2=Integer.parseInt(b);
            int ans=0;
            if(fun.equals("+")) {
                ans=i1+i2;
            }else if(fun.equals("-")) {
                ans=i1-i2;
            }else if(fun.equals("*")) {
                ans=i1*i2;
            }
            // System.out.println(ans);
            request.setAttribute("ans", ans);

            request.getRequestDispatcher("index.jsp").forward(request,response);
        }catch(Exception e) {
            System.out.println(e);
        }
    }
}
```



**OUTPUT:**

## Mathematical Operation

- ☐ Addition
- ☐ Multiplication
- ☐ Subtraction

Enter the First Value:

Enter the Second Value:

**Ans = null**

## Mathematical Operation

- ☐ Addition
- ☐ Multiplication
- ☐ Subtraction

Enter the First Value:

Enter the Second Value:

**Ans = 47**



**Learning outcomes (What I have learnt):**

1. Learn About the servlet.
2. Learn about jsp and dynamic web project.
3. Learn about the tomcat server and its integrations with the java.

**Evaluation Grid (To be created per the faculty's SOP and Assessment guidelines):**

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.	Worksheet completion including writing learning objectives/Outcomes. (To be submitted at the end of the day).		
2.	Post-Lab Quiz Result.		
3.	Student Engagement in Simulation/Demonstration/Performance and Controls/Pre-Lab Questions.		
	Signature of Faculty (with Date):	Total Marks Obtained:	