**Ex No : 1 Simple Java Program**

**Date :**

**Aim:**

To find the minimum of 2 numbers through command line argument, To find sum and product of a given digit, To convert the number of days into months and days, To display Greet message according to marks obtained by the student, To Display the Pyramid Pattern

**1 a. Find Minimum of 2 nos. And get the input through command line argument.**

**Algorithm:**

1. Import the necessary package 'exercise1' and 'java.util.Scanner'.

2. Declare two integer variables 'a' and 'b'.

3. Create a Scanner object 'obj' to read input from the user.

4. Read the first command-line argument (args[0]) and convert it to an integer, storing it in 'a'.

5. Print the value of 'a'.

6. Read the second command-line argument (args[1]) and convert it to an integer, store it in 'b'.

7. Print the value of 'b'.

8. Compare 'a' and 'b': If 'a' is less than 'b', print "A is Minimum". Otherwise, print "B is minimum".

9. Close the Scanner object.

**Program:**

package exercise1;

import java.util.Scanner;

public class MinCommandLineArg {

public static void main(String[] args) {

int a,b;

Scanner obj = new Scanner(System.in);

a = Integer.parseInt(args[0]);

System.out.println("Value of A is : " + a);

b = Integer.parseInt(args[1]);

System.out.println("Value of B is :" + b);

if(a<b){System.out.print("A is Minimum");}

else{System.out.println("B is minimum");}

obj.close();

}

}

**Output:**

Command line argument passed : 10 20

Value of A is : 10

Value of B is : 20

A is Minimum

**1 b. Write a program to find SUM AND PRODUCT of a given Digit**

**Algorithm:**

1. Import the necessary package 'exercise1' and 'java.util.Scanner'.

2. Declare two integer variables 'a' and 'b'.

3. Create a Scanner object 'obj' to read input from the user.

4. Prompt the user to enter the value of 'a'.

5. Read an integer from the user using the Scanner object and store it in 'a'.

6. Prompt the user to enter the value of 'b'.

7. Read an integer from the user using the Scanner object and store it in 'b'.

8. Calculate the sum of 'a' and 'b' and print it.

9. Calculate the product of 'a' and 'b' and print it.

10. Close the Scanner object.

**Program:**

package exercise1;

import java.util.Scanner;

public class SumProduct {

public static void main(String[] args) {

int a,b;

Scanner obj = new Scanner(System.in);

System.out.print("Enter the value of a : ");

a = obj.nextInt();

System.out.print("Enter the value of b : ");

b = obj.nextInt();

System.out.println("Sum of the values : " + (a+b));

System.out.println("Product of the values : " + (a\*b));

}

}

**Output:**

Enter the value of a : 10

Enter the value of b : 20

Sum of the values : 30

Product of the values : 200

**1 c. Write a program to convert given no. of days into months and days.**

**(Assume that each month is of 30 days)**

**Example:**

**Input - 69**

**Output - 69 days = 2 Month and 9 days**

**Algorithm:**

1. Import the necessary package 'exercise1' and 'java.util.Scanner'.

2. Declare an integer variable 'days'.

3. Create a Scanner object 'obj' to read input from the user.

4. Prompt the user to enter the number of days.

5. Read an integer from the user using the Scanner object and store it in 'days'.

6. Declare two integer variables 'months' and 'RDays' to store the calculated values.

7. Calculate the number of months by dividing 'days' by 30 and store it in 'months'.

8. Print the calculated number of months.

9. Calculate the remaining days by subtracting the total days represented by 'months' from 'days', and store it in 'RDays'.

10. Print the calculated number of remaining days.

11. Close the Scanner object.

**Program:**

package exercise1;

import java.util.Scanner;

public class DaysToMonths {

public static void main(String[] args) {

int days;

Scanner obj = new Scanner(System.in);

System.out.print("Enter the number of Days : ");

days = obj.nextInt();

int months,RDays;

months = days / 30;

System.out.print( months +" Months and");

RDays = days - (months\*30);

System.out.println( RDays +" Days");

}

}

**Output:**

Enter the number of Days : 69

2 Months and 9 Days

**1 d. Write a program to display a greet message according to Marks obtained by student using switch case.Condition: 10 (Excellent)**

**9-8 (very good)**

**7-6 (Good)**

**5-3(poor)**

**2-0(very poor)**

**Algorithm:**

1. Import the necessary package 'exercise1' and 'java.util.Scanner'.

2. Declare an integer variable 'mark'.

3. Create a Scanner object 'obj' to read input from the user.

4. Prompt the user to enter the mark of the student (1 – 10).

5. Read an integer from the user using the Scanner object and store it in 'mark'.

6. Use conditional statements to evaluate the value of 'mark' and determine the student's performance:

a. If 'mark' is equal to 10, print "Excellent".

b. Else if 'mark' is between 8 and 9 (inclusive), print "Very Good".

c. Else if 'mark' is between 6 and 7 (inclusive), print "Good".

d. Else if 'mark' is between 3 and 5 (inclusive), print "Poor".

e. Else if 'mark' is between 0 and 2 (inclusive), print "Very Poor".

f. If none of the above conditions are met, print "Invalid Entry".

7. Close the Scanner object**.**

**Program:**

package exercise1;

import java.util.Scanner;

public class SwitchCase {

public static void main(String[] args) {

int mark;

Scanner obj = new Scanner(System.in);

System.out.print("Enter the Mark of the Student ( 1 – 10 ) : ");

mark = obj.nextInt();

if(mark==10){System.out.println("Excellent");}

else if ((mark<=9)&&(mark>=8)){System.out.println("Very Good");}

else if ((mark<=7)&&(mark>=6)){System.out.println("Good");}

else if ((mark<=5)&&(mark>=3)){System.out.println("Poor");}

else if ((mark<=2)&&(mark>=0)){System.out.println("Very Poor");}

else{System.out.print("Invalud Entry");}

}

}

**Output:**

Case I:

Enter the Mark of the Student ( 1 – 10 ) : 10

Excellent

Case II:

Enter the Mark of the Student ( 1 – 10 ) : 4

Poor

**1 e.**

****

**Algorithm:**

1. Import the necessary package 'exercise1' and 'java.util.Scanner'.

2. Declare three integer variables 'i', 'j', and 'n'.

3. Create a Scanner object 'obj' to read input from the user.

4. Prompt the user to enter the value of 'n'.

5. Read an integer from the user using the Scanner object and store it in 'n'.

6. Use a nested loop to print the pattern:

a. Outer loop runs from 0 to 'n-1' (inclusive):

- Initialize 'i' to 0.

- Continue loop while 'i' is less than 'n'.

- Increment 'i' after each iteration.

b. Inner loop 1 prints spaces before asterisks:

- Initialize 'j' to '2 \* (n - i)'.

- Continue loop while 'j' is greater than or equal to 0.

- Decrement 'j' after each iteration.

- Print a space.

c. Inner loop 2 prints asterisks:

- Initialize 'j' to 0.

- Continue loop while 'j' is less than or equal to 'i'.

- Increment 'j' after each iteration.

- Print an asterisk followed by a space.

d. Print a newline to move to the next row after printing spaces and asterisks.

7. Close the Scanner object.

**Program:**

package exercise1;

import java.util.Scanner;

public class Pattern {

public static void main(String[] args) {

int i,j,n;

Scanner obj = new Scanner(System.in);

System.out.print("Enter the value of n : ");

n = obj.nextInt();

for(i=0; i<n; i++)

{

for(j=2\*(n-i); j>=0; j--) { System.out.print(" "); }

for(j=0; j<=i; j++) { System.out.print("\* "); }

System.out.println(){

} }

**Output:**

Enter the value of n: 5

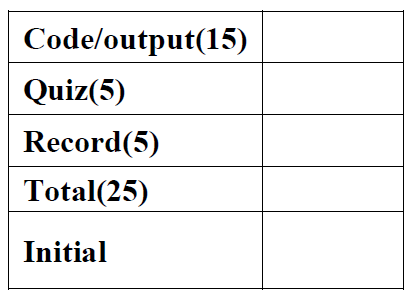
\*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*



**Result :**

Thus the program for finding the minimum of 2 numbers through command line argument, finding sum and product of a given digit, converting the number of days into months and days, displaying Greet message according to marks obtained by the student, and Displaying the Pyramid Pattern have been executed successfully and output is obtained using JAVA programming language

**Ex No : 2 Arrays and classes**

**Date : 30.08.2023**

**Aim:**

2.a) To Develop a Java application to Calculate or set Employee salary and compute the salary amount using the following conditions

2.b)To Develop a Java application to generate Electricity bill and to compute the bill amount using the following given tariff conditions

**2 a. To Develop a Java application to Calculate or set Employee salary and compute the salary amount using the following conditions**

**Algorithm:**

1. Create a class named `Bill`:

1.1. Declare instance variables `ConsumerNo`, `ConsumerName`, `prevMonthReading`, `currMonthReading`, and `type`.

1.2. Create a constructor that takes parameters to initialize these variables.

1.3. Create a method `calculateBill` to calculate the bill amount based on consumption and type of consumer:

1.3.1. Calculate units consumed: `unitsConsumed = currMonthReading - prevMonthReading`.

1.3.2. Initialize `billAmount` to 0.

1.3.3. If the type is "domestic":

1.3.3.1. Implement tariff calculation based on units consumed:

- First 100 units: Rs. 1 per unit

- 101-200 units: Rs. 2.5 per unit

- 201-500 units: Rs. 4 per unit

- 501 units and above: Rs. 6 per unit

1.3.4. Else, if the type is "commercial":

1.3.4.1. Implement tariff calculation based on units consumed:

- First 100 units: Rs. 2 per unit

- 101-200 units: Rs. 4.5 per unit

- 201-500 units: Rs. 6 per unit

- 501 units and above: Rs. 7 per unit

1.3.5. Return the calculated `billAmount`.

2. Create the `ElectricBill` class:

2.1. Create a `Scanner` object for input.

2.2. Ask the user for the number of customers `n`.

2.3. Create an array of `Bill` objects with size `n`.

2.4. For each customer (loop `i` from 0 to `n-1`):

2.4.1. Ask for customer details:

- Ask for customer number and store it in `C\_No`.

- Ask for customer name and store it in `C\_Name`.

- Ask for previous month reading and store it in `P\_Amt`.

- Ask for current month reading and store it in `C\_Amt`.

- Ask for the type of consumer (domestic/commercial) and store it in `type\_Of\_Consumer`.

2.4.2. Create a `Bill` object using the provided values and add it to the `newbill` array at index `i`.

2.5. Display the electricity bills:

2.5.1. Iterate through the `newbill` array (loop `i` from 0 to `n-1`):

2.5.1.1. Calculate the bill amount for the current customer using the `calculateBill` method.

2.5.1.2. Display the customer number, name, and calculated bill amount.

3. Close the `Scanner`.

**Program:**

import java.util.Scanner;

class Bill{

int CustomerNo;

String CustomerName;

int prevMonthReading;

int currMonthReading;

String type;

public Bill(int CustomerNo,String CustomerName, int prevMonthReading,int currMonthReading,String type){

this.CustomerNo = CustomerNo ;

this.CustomerName = CustomerName ;

this.prevMonthReading = prevMonthReading;

this.currMonthReading = currMonthReading;

this.type = type;

}

public double calculateBill(){

int unitsConsumed = currMonthReading - prevMonthReading;

double billAmount = 0;

if(type.equalsIgnoreCase("domestic")){

if (unitsConsumed <= 100) {

billAmount = unitsConsumed;

} else if ((unitsConsumed > 101 ) && (unitsConsumed <= 200)) {

billAmount = 100 + (unitsConsumed - 100) \* 2.5;

} else if ((unitsConsumed > 201 ) && (unitsConsumed <= 500)) {

billAmount = 100 + 100 \* 2.5 + (unitsConsumed - 200) \* 4;

} else {

billAmount = 100 + 100 \* 2.5 + 300 \* 4 + (unitsConsumed - 500) \* 6;

}

}

else if (type.equalsIgnoreCase("commercial")) {

if (unitsConsumed <= 100) {

billAmount = unitsConsumed \* 2;

} else if ((unitsConsumed > 101 ) && (unitsConsumed <= 200)) {

billAmount = 100 + (unitsConsumed - 100) \* 4.5;

} else if ((unitsConsumed > 201 ) && (unitsConsumed <= 500)) {

billAmount = 100 + 100 \* 4.5 + (unitsConsumed - 200) \* 6;

} else {

billAmount = 100 + 100 \* 4.5 + 300 \* 6 + (unitsConsumed - 500) \* 7;

}

}

return billAmount;

}

}

public class ElectricBill {

public static void main(String[] args) {

Scanner scan = new Scanner(System.in);

System.out.print("Enter the number of Customers : ");

int n = scan.nextInt();

Bill[] newbill = new Bill[n];

for (int i=0;i<n;i++) {

System.out.print("Enter the Customer Number : ");

int C\_No = scan.nextInt();

System.out.print("Enter the name of the Customer : ");

String C\_Name = scan.next();

System.out.print("Enter previous month reading : ");

int P\_Amt = scan.nextInt();

System.out.print("Enter this month reading : ");

int C\_Amt = scan.nextInt();

System.out.print("Enter the type of Customer (domestic/commercial) : ");

String type\_Of\_Customer = scan.next();

newbill[i] = new Bill(C\_No,C\_Name,P\_Amt,C\_Amt,type\_Of\_Customer);

}

System.out.println();

System.out.println("Electricity Bills : ");

{

for (int i = 0; i < n; i++) {

double billAmount = newbill[i].calculateBill();

System.out.println("Customer " + (i + 1) + " - " + newbill[i].CustomerName + ": Rs. " + billAmount);

}

}

}

}

**Output:**

Enter the number of Customers : 2

Enter the CustomerNumber : 101

Enter the name of the Customer: Customer1

Enter previous month reading : 300

Enter this month reading : 650

Enter the type of Customer (domestic/commercial) : domestic

Enter the CustomerNumber : 102

Enter the name of the Customer: Customer2

Enter previous month reading : 300

Enter this month reading : 650

Enter the type of Customer (domestic/commercial) : commercial

Electricity Bills :

Customer 1 - Customer1: Rs. 950.0

Customer 2 - Customer2: Rs. 1450.0

**2 b.** **Develop a Java application to generate Electricity bill. Create a class with the**

**Following members: Consumer no., consumer name, previous month reading,**

**current month reading, type of EB connection (i.e domestic or commercial). Compute**

**the bill amount using the following tariff,**

**If the type of the EB connection is domestic, calculate the amount to be paid as follows:**

* **First 100 units-Rs.1 per unit**
* **101-200 units - Rs. 2.50 per unit**
* **201 -500 units - Rs. 4 per unit**
* **501 units - Rs. 6 per unit**

**If the type of the EB connection is commercial, calculate the amount to be paid as follows:**

* **First 100 units - Rs. 2 per unit**
* **101-200 units - Rs. 4.50 per unit**
* **201-500 units-Rs.6 per unit**
* **501 units - Rs. 7 per unit**

**Algorithm:**

1. Create a class named `details`:

1.1. Declare instance variables `Empid`, `salary`, `F\_Name`, and `L\_Name`.

1.2. Create a constructor to initialize these variables.

1.3. Create getter and setter methods for `Empid`, `F\_Name`, `L\_Name`, `salary`.

1.4. Create methods to calculate `getAnnualSalary`, `raiseSalary`, and `getName`.

1.5. Create a `toString` method to return a formatted string representation.

2. Create the `Employee` class:

2.1. Create a `Scanner` object for input.

2.2. Ask the user for the number of employees `n`.

2.3. Create an array of `details` objects with size `n`.

2.4. For each employee:

2.4.1. Ask for employee details: ID, first name, last name, and salary.

2.4.2. Create a `details` object with the entered details and add it to the array.

2.5. Create a loop to handle menu options:

2.5.1. Display menu options: Set Salary, Raise Salary, Show Employee Details, Exit.

2.5.2. Ask the user for their choice.

2.5.3. Use a switch based on the choice:

- Case 1:

- Ask for employee ID and new salary.

- Find the employee by ID and update the salary.

- Case 2:

- Ask for employee ID and raise percentage.

- Find the employee by ID and raise the salary.

- Case 3:

- Display details of all employees.

- Case 4:

- Exit the loop.

2.6. Close the `Scanner`.

**Program:**

package employee;

import java.util.Scanner;

class details{

int Empid,salary;

String F\_Name,L\_Name;

details(int id,String fn, String ln,int sal){

Empid = id;

F\_Name = fn;

L\_Name = ln;

salary = sal;

}

public int getID(){return Empid;}

public String getFirstName(){return F\_Name;}

public String getLastName(){return L\_Name;}

public String getName(){return F\_Name+" "+L\_Name;}

public int getSalary(){return salary;}

public void setSalary(int sa){salary=sa;}

public int getAnnualSalary(){return salary = salary \* 12;}

public int raiseSalary(int percent) {return salary = (int) (salary \* (1 + percent / 100.0));}

public String toString(){

return "EMPID : "+ Empid +" Name :"+F\_Name+" "+L\_Name+" Salary :"+salary;

}

}

class Employee {

public static void main(String[] args) {

Scanner s = new Scanner(System.in);

System.out.print("Enter the number of Employees: ");

int n = s.nextInt();

details[] emp = new details[n];

int Empid, salary;

String F\_Name, L\_Name;

for (int i = 0; i < n; i++) {

System.out.print("Enter the Employee ID: ");

Empid = s.nextInt();

System.out.print("Enter First Name: ");

F\_Name = s.next();

System.out.print("Enter Last Name: ");

L\_Name = s.next();

System.out.print("Enter Salary: ");

salary = s.nextInt();

emp[i] = new details(Empid, F\_Name, L\_Name, salary);

}

boolean continueUpdating = true;

while (continueUpdating) {

System.out.println("\nOptions:");

System.out.println("1. Set Salary");

System.out.println("2. Raise Salary");

System.out.println("3. Show Employee Details");

System.out.println("4. Exit");

System.out.print("Enter your choice: ");

int choice = s.nextInt();

switch (choice) {

case 1:

System.out.print("Enter the Employee ID: ");

int target = s.nextInt();

System.out.print("Enter the amount to set Salary: ");

int setsal = s.nextInt();

for (int i = 0; i < emp.length; i++) {

if (target == emp[i].Empid) {

emp[i].setSalary(setsal);

System.out.println("Updated Salary: " + emp[i].getSalary());

}

}

break;

case 2:

System.out.print("Enter the Employee ID: ");

target = s.nextInt();

System.out.print("Enter the percentage to be raised: ");

int percent = s.nextInt();

for (details empDetail : emp) {

if (target == empDetail.Empid) {

int raisedSalary = empDetail.raiseSalary(percent);

System.out.println("The salary raised to: " + raisedSalary);

}

}

break;

case 3:

for (int i = 0; i < n; i++) {

System.out.println(emp[i]);

}

break;

case 4:

continueUpdating = false;

System.out.println("Exiting...");

break;

default:

System.out.println("Invalid choice");

break;

}

}

}

}

**Output:**

Enter the number of Employees: 1

Enter the Employee ID: 1221

Enter First Name: AAA

Enter Last Name: BBB

Enter Salary: 12000

Options:

1. Set Salary

2. Raise Salary

3. Show Employee Details

4. Exit

Enter your choice: 1

Enter the Employee ID: 1221

Enter the amount to set Salary: 14000

Updated Salary: 14000

Options:

1. Set Salary

2. Raise Salary

3. Show Employee Details

4. Exit

Enter your choice: 2

Enter the Employee ID: 1221

Enter the percentage to be raised: 5

The salary raised to: 14700

Options:

1. Set Salary

2. Raise Salary

3. Show Employee Details

4. Exit

Enter your choice: 3

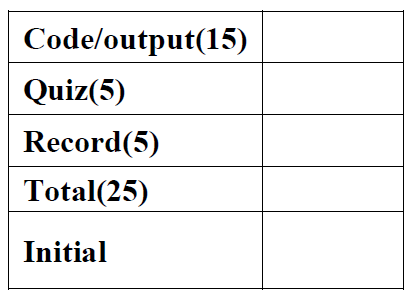
EMPID : 1221 Name :AAA BBB Salary :14700

Options:

1. Set Salary

2. Raise Salary

3. Show Employee Details

4. Exit

Enter your choice: 4

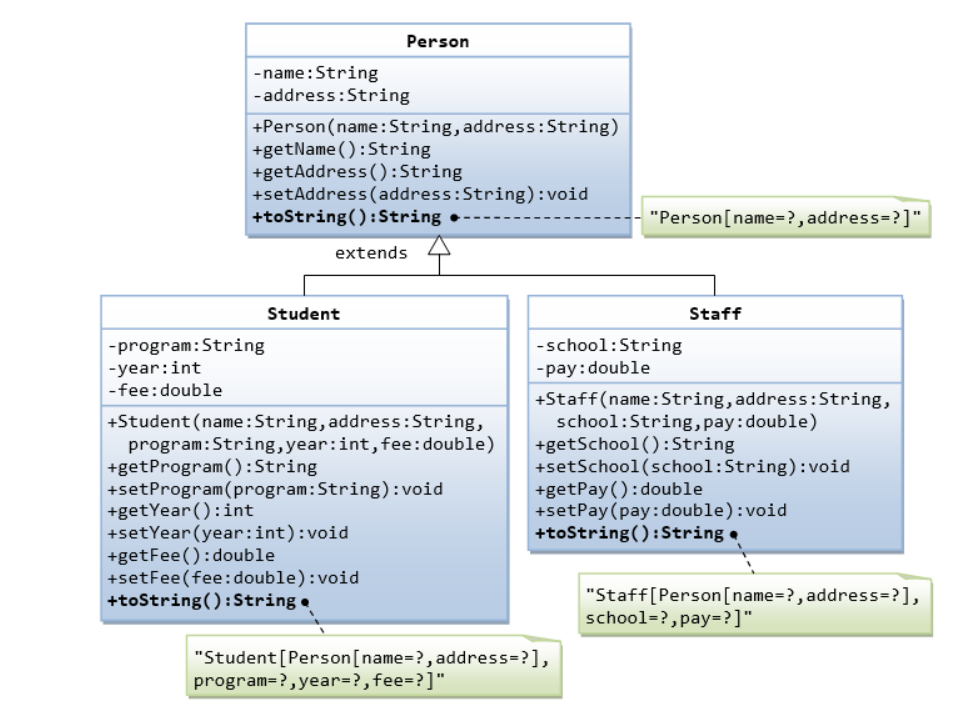
Exiting...

**Result :**

Thus the program To Develop a Java application to Calculate or set Employee salary and compute the salary amount using the following conditions and To Develop a Java application to generate Electricity bill and to compute the bill amount using the following given tariff conditions have been executed successfully.

**Ex No : 3 Inheritance**

**Date : 06.06.2023**



**Aim:**

To Develop a Java application that utilizes inheritance to manage and manipulate Student and Staff details within the Person class, incorporating the specified conditions for implementation.

**Algorithm:**

1. Start the program.

2. Initialize variables:

- number\_of\_persons to store the number of persons.

- S\_name, S\_add, prg, scl to store user inputs.

- fe, pa to store fee and pay.

3. Create a Scanner object for user input.

4. Prompt the user to enter the number of persons.

5. Create arrays for Person, Student, and Staff objects to store information for each person.

6. Loop through each person:

a. Prompt the user to choose between Student and Staff.

b. Depending on the choice:

- Initialize Person, Student, or Staff objects with empty values.

- Prompt the user for name and address and update the respective objects.

- If it's a Student:

i. Prompt the user for program, year, and fee.

ii. Update the Student object with these values.

iii. Print the Student's information.

iv. Ask the user if they want to update the information:

- If yes, prompt for which field to update (Program/Year/Fee).

- Update the chosen field and print the updated information.

- If it's a Staff member:

i. Prompt the user for school and pay.

ii. Update the Staff object with these values.

iii. Print the Staff member's information.

iv. Ask the user if they want to update the information:

- If yes, prompt for which field to update (School/Pay).

- Update the chosen field and print the updated information.

c. Repeat the loop for the next person.

7. End the program.

**Program:**

import java.util.Scanner;

public class Exercise3 {

public static void main(String[] args) {

int number\_of\_persons,yr;

String S\_name,S\_add,prg,scl;

double fe,pa;

Scanner scan =new Scanner(System.in);

System.out.print("Enter the number of Persons : ");

number\_of\_persons= scan.nextInt();

Person[] newPerson = new Person[number\_of\_persons];

Student[] newStudent = new Student[number\_of\_persons];

Staff[] newStaff = new Staff[number\_of\_persons];

for (int i = 0; i < number\_of\_persons; i++) {

int choice;

System.out.println("1.Student\n2.Staff");

System.out.print("Enter the Person type :");

choice= scan.nextInt();

newPerson[i] = new Person("", "");

newStudent[i] = new Student("", "", "", 0, 0.0);

newStaff[i] = new Staff("", "", "", 0.0);

switch (choice){

case 1:

S\_name = newPerson[i].getName();

S\_add = newPerson[i].getAddress();

prg = newStudent[i].getProgram();

yr = newStudent[i].getYear();

fe =newStudent[i].getFee();

newStudent[i] = new Student(S\_name,S\_add,prg,yr,fe);

System.out.println(newStudent[i]);

System.out.println("1 to Continue || Enter 0 to update");

int select = scan.nextInt();

if (select==0) {

System.out.print("To Set\n1.Program\n2.Year\n3.Fee\n");

int choice2 = scan.nextInt();

switch (choice2) {

case 1:

System.out.println("Enter the program to update : ");

String NewProgram = scan.nextLine();

newStudent[i].setProgram(NewProgram);

System.out.println(newStudent[i]);

break;

case 2:

System.out.println("Enter the Year to update : ");

int NewYear = scan.nextInt();

newStudent[i].setYear(NewYear);

System.out.println(newStudent[i]);

break;

case 3:

System.out.println("Enter the Fee to update : ");

double NewFee = scan.nextDouble();

newStudent[i].setFee(NewFee);

System.out.println(newStudent[i]);

break;

default:

break;

}

}

break;

case 2:

S\_name = newPerson[i].getName();

S\_add = newPerson[i].getAddress();

scl = newStaff[i].getSchool();

pa =newStaff[i].getPay();

newStaff[i] = new Staff(S\_name,S\_add,scl,pa);

System.out.println(newStaff[i]);

System.out.println("1 to Continue || Enter 0 to update");

select = scan.nextInt();

if (select==0) {

System.out.print("To Set\n1.School\n2.Pay\n");

int choice2 = scan.nextInt();

switch (choice2) {

case 1:

System.out.println("Enter the School to update : ");

String NewSchool = scan.nextLine();

newStaff[i].setSchool(NewSchool);

System.out.println(newStaff[i]);

break;

case 2:

System.out.println("Enter the Pay to update : ");

int NewPay = scan.nextInt();

newStaff[i].setPay(NewPay);

System.out.println(newStaff[i]);

break;

default:

break;

}

}

break;

default:

break;

}

}

}

}

class Person{

Scanner obj = new Scanner(System.in);

String name;

String address;

public Person(String name,String address){

this.name=name;

this.address=address;

}

String getName(){

System.out.print("Enter Name : ");

name = obj.nextLine();

return name;

}

String getAddress(){

System.out.print("Enter Address : ");

address = obj.nextLine();

return address ;

}

}

class Student extends Person{

String program;

int year;

double fee;

Student(String name,String address,String program,int year,double fee){

super(name, address);

this.program=program;

this.year=year;

this.fee=fee;

}

String getProgram(){

System.out.print("Enter the Program : ");

program = obj.nextLine();

return program;

}

void setProgram(String prg){

this.program=prg;

}

int getYear(){

System.out.print("Enter the Year : ");

year = obj.nextInt();

return year;

}

void setYear(int yr){

this.year =yr;

}

double getFee(){

System.out.print("Enter the Fees : ");

fee = obj.nextDouble();

return fee;

}

void setFee(double fe){

this.fee=fe;

}

@Override

public String toString() {

return "Name : "+name+"\nAddress : "+address+"\nProgram : "+program+"\nYears : "+year+"\nFees : "+fee;

}

}

class Staff extends Person{

String school;

double pay;

Staff(String name,String address,String school,double pay){

super(name, address);

this.school=school;

this.pay=pay;

}

String getSchool(){

System.out.print("Enter the School Name : ");

school = obj.nextLine();

return school;

}

double getPay(){

System.out.print("Enter the Basic Pay : ");

pay = obj.nextDouble();

return pay;

}

void setSchool(String scl){

this.school=scl;

}

void setPay(double pa){

this.pay=pa;

}

@Override

public String toString() {

return "Name : "+name+"\nAddress : "+address+"\nSchool : "+school+"\nPay : "+pay;

}

}

**Output:**

Enter the number of Persons : 2

1.Student

2.Staff

Enter the Person type :1

Enter Name : Navin Kumaran

Enter Address : 123 Main St

Enter the Program : Computer Science

Enter the Year : 2

Enter the Fees : 86000

1 to Continue || Enter 0 to update

0

To Set

1.Program

2.Year

3.Fee

1

Enter the program to update : Information Technology

Name : Navin Kumaran

Address : 123 Main St

Program : Information Technology

Years : 2

Fees : 86000

1.Student

2.Staff

Enter the Person type :2

Enter Name : Jane Doe

Enter Address : 456 Elm St

Enter the School Name : XYZ School

Enter the Basic Pay : 25000

1 to Continue || Enter 0 to update

0

To Set

1.School

2.Pay

2

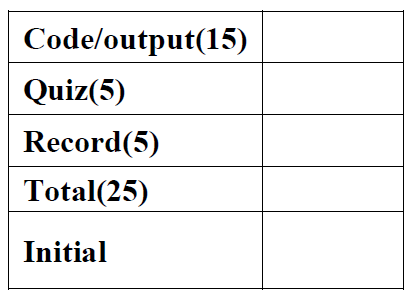
Enter the Pay to update : 27000

Name : Jane Doe

Address : 456 Elm St

School : XYZ School

Pay : 27000.0



**Result** :

Thus, The program for managing and manipulating Student and Staff details within the Person class, while adhering to the specified conditions, has been successfully executed.