# COSC 12043

## Object Oriented Programming

**(Academic Year 2019/2020)**

## Project Report

## "School Management System"

## Group No.09

PS/2019/279 - Arshana Sagadevan
PS/2019/051 - Pitipanage Malaka Dilshan
PS/2019/181 - Sashini Udyani Jayawardhana
PS/2019/233 - Daluwakgodage Dulakshi Anuradha
PS/2019/195 - Wathsala wimansani Ranasingha

# Introduction

We have developed an OOP-based system for a school management system, as group No.09.

School management systems become an integral a part of the upper education system. We create it for to make the works easier to management system for the principal and teachers. The executive aspects of such systems could include class rosters and therefore the ability to record students' grades. With relevance the teaching aspects, however, it might include Servant details, Tenders information and treasury information as well.

This project **"School Management System"** provides us a simple interface for maintenance of student information. It can be used by educational institutes or colleges to maintain the records of students easily. Achieving this objective is difficult using a manual system as the information is scattered, can be redundant and collecting relevant information may be very time consuming. **All these problems are solved using this project.**

**Throughout the project the focus has been on presenting information in an easy and intelligible manner. The project is very useful for those who want**

**to know about School Management Systems and want to develop program based on the same concept.**

The project provides facilities like registration and profile creation of students thus reducing paperwork and automating the record generation process in an educational institution.

# Acknowledgement

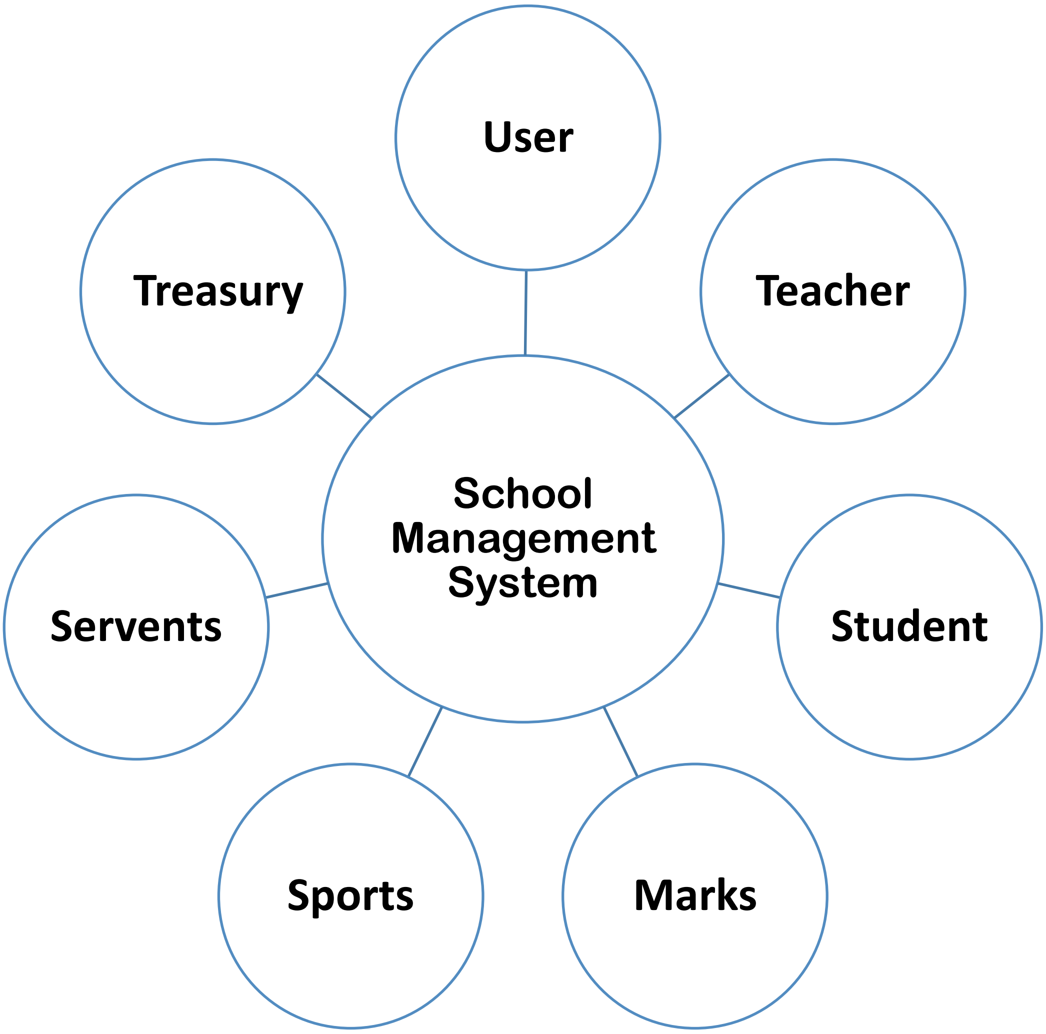
We take this opportunity to express our sincere gratitude to all those who helped us in various capacities in undertaking this project and devising the report.

We are privileged to express our sense of gratitude to our respected **Dr. B.M.Thosini Kumarika** whose unparalleled knowledge, moral and judgment along with know-how, was an immense support in completing the project.

We take this opportunity also to thank our friends and contemporaries for their co-operation and compliance.

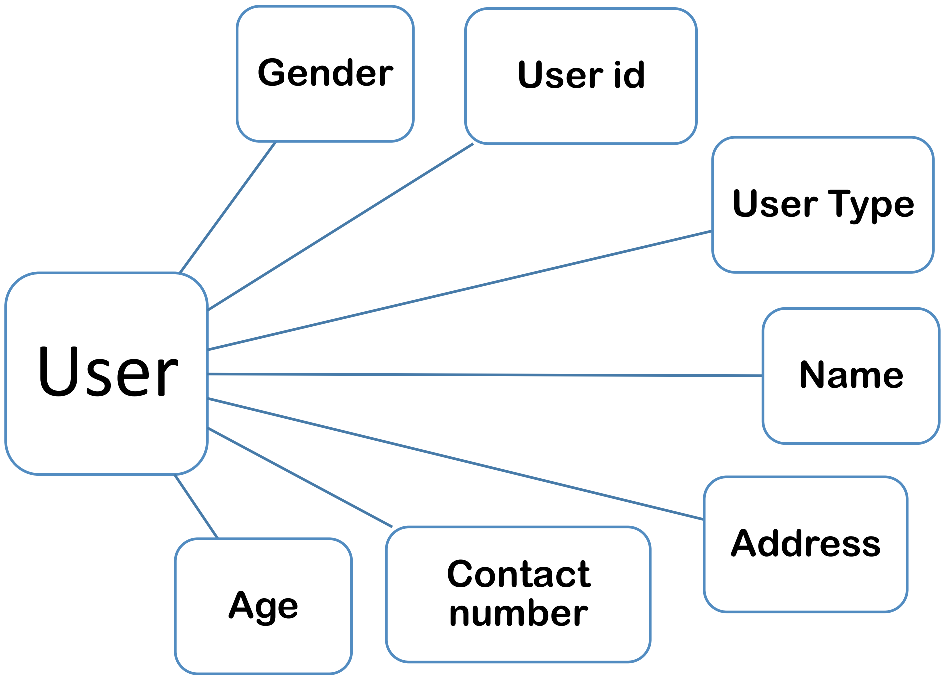
# Classes and their attributes and methods

**Here is an overview of the classes included in our system.**



This is our basic class plan and we have improved this design with several classes, methods, attributes and some applications of OOP concepts. Those things will be discussed further.

# Class 01 – User Class



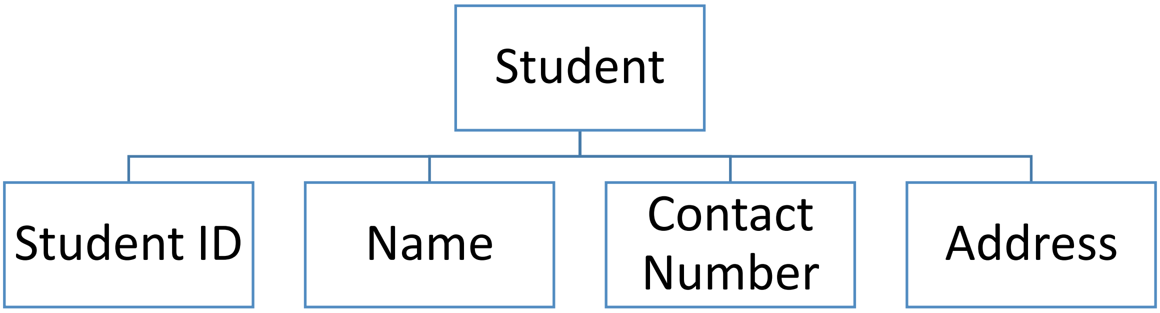
User class represents the users of the system. Teacher, Student, Servant are some users of the proposed system.

User class has following structure:

```
class User{

        Properties:

                private String userType;

                private Integer userId;

                private String name;

                private String address;

                private String contactNumber;

                private Integer age;

                private String gender;

                private String username;

                private String password;
        Behaviors:
                public HashMap<String, String> createProfile(){}
                public boolean login(){}
                public void viewProfile(User user){}
                public void editProfile(User user){}
}
```

- All these attributes and methods are inherited by child classes. Firstly, user must create a profile.
- In order to log in to the system user has to give correct username and password.
- If the username and password are incorrect, they cannot log in to the system. After login to the system, they can view their profile.
- Then if they want, they can update their details in the update profile section.

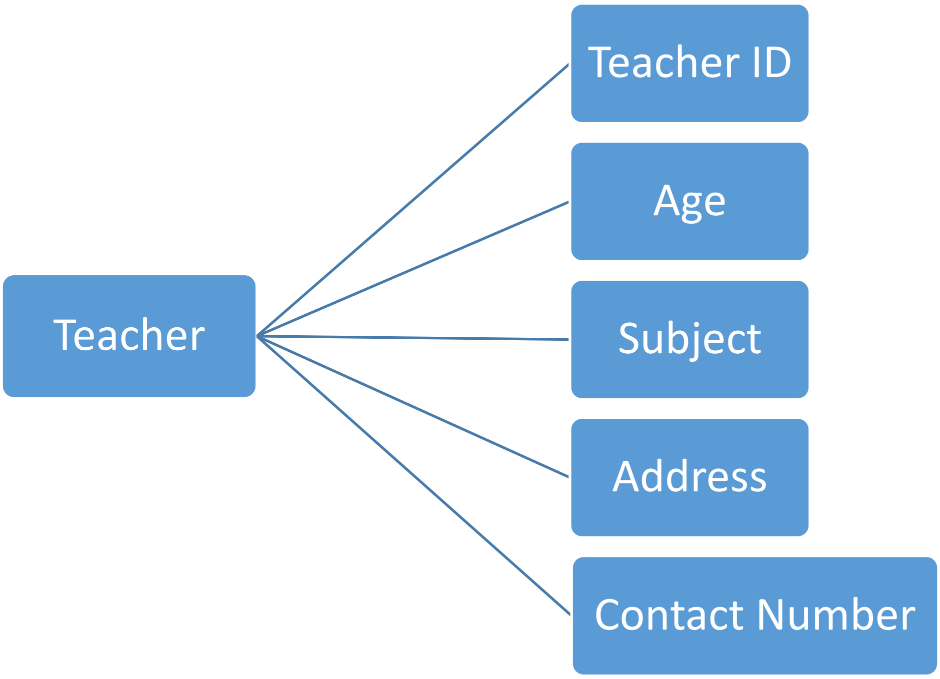
## Class 02 – Student Class



- The student class has four properties namely Student ID, Name, Contact number, Address.
- All these properties have respective methods to get and set object values. Used ArrayList for collection framework.
- Use exception handling to avoid from the exceptions occurs when run time.
- The purpose of this class is to maintain student details.
- All these attributes and methods are inherited by child classes.

- Use exception handling to avoid from the exceptions occurs when run time.

## Class 03 – Teacher Class



Teacher

Teacher ID

Age

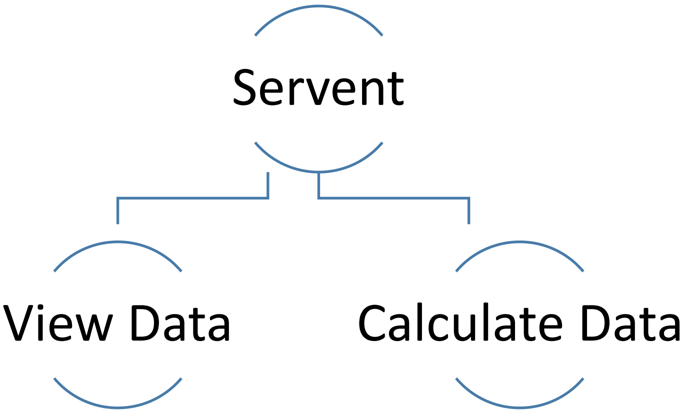
Subject

Address

Contact Number

- The teacher class has four properties namely Teacher ID, Name, Contact number, Address.
- All these properties have respective methods to get and set object values. Used ArrayList for collection framework.
- Use exception handling to avoid from the exceptions occurs when run time.
- The purpose of this class is to maintain student details.
- All these attributes and methods are inherited by child classes.
- Use exception handling to avoid from the exceptions occurs when run time.

# Class 04 – Servant Class



- This class is designed to retain details of school workers and calculate their monthly salary.

- This required the names, age, daily salary, allowances, etc. of the employees employed as data.

- There are two methods in servant class.

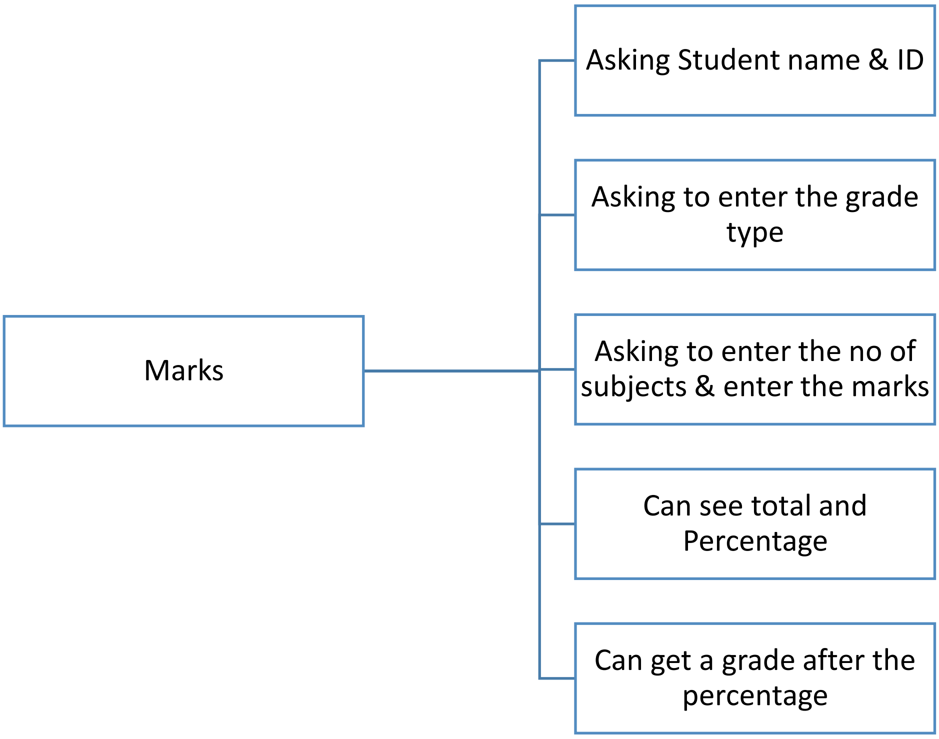
1) Salary

- When calculate salary you should enter occupation, working hours and over time hours. (Salary() )
- Then you can calculate salary according to the relevant job.
- Use exception handling to avoid from the exceptions occurs when run time.
- Used ArrayList for collection framework.

2) View Data

- In this method you can enter employee id and then can view his data.(Name, age and occupation )
- Use exception handling to avoid from the exceptions occurs when run time.
- Used ArrayList for collection framework.

# Class 05 – Marks Class


Marks
Asking Student name & ID
Asking to enter the grade type
Asking to enter the no of subjects & enter the marks
Can see total and Percentage
Can get a grade after the percentage

In this class a teacher can add the student subjects and marks of students. Finally can see the total of the marks and percentage. There are a 4 types of grade.

They are:

- Grade 1 to 5;
- Grade 6 to 9;
- Grade 10 & 11;
- Grade 12 & 13;

It's easy to accesses the student marks by dividing the grades. For this class there were used arraies for the ender the number of subjects and if else statement.

Accessing array elements using the for loop
Initialize the array
Ask the user to enter the Array Size
Taking input from user
Function that returns the average of an array.

**These are some of statement in my class**

## Class 06 – Sports Class

• This sports class is designed to collect data related to each sport.

• Accordingly, the coaches, teachers in charge and training dates and times related to each sport are collected as data.

•There are two main methods used here.

1) Input data

• What is done here is to add a new set of information to the system.

• Where the ArrayList concept is applied.

• Here you can enter information related to a new sport and view it.

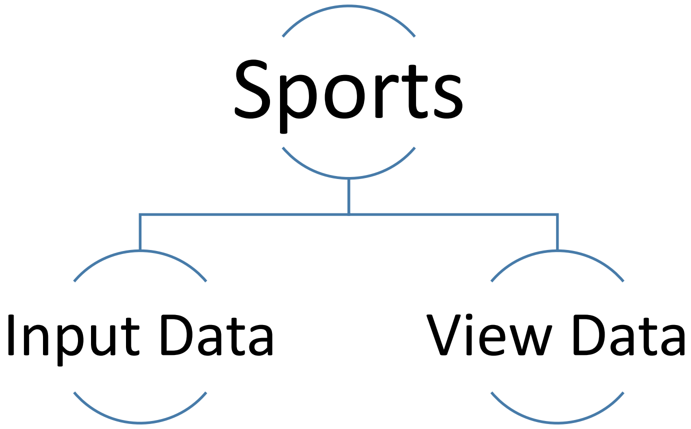
• Use exception handling to avoid from the exceptions occurs when run time.

2) View data

• Here is what happens Display information related to the games currently included in the system.

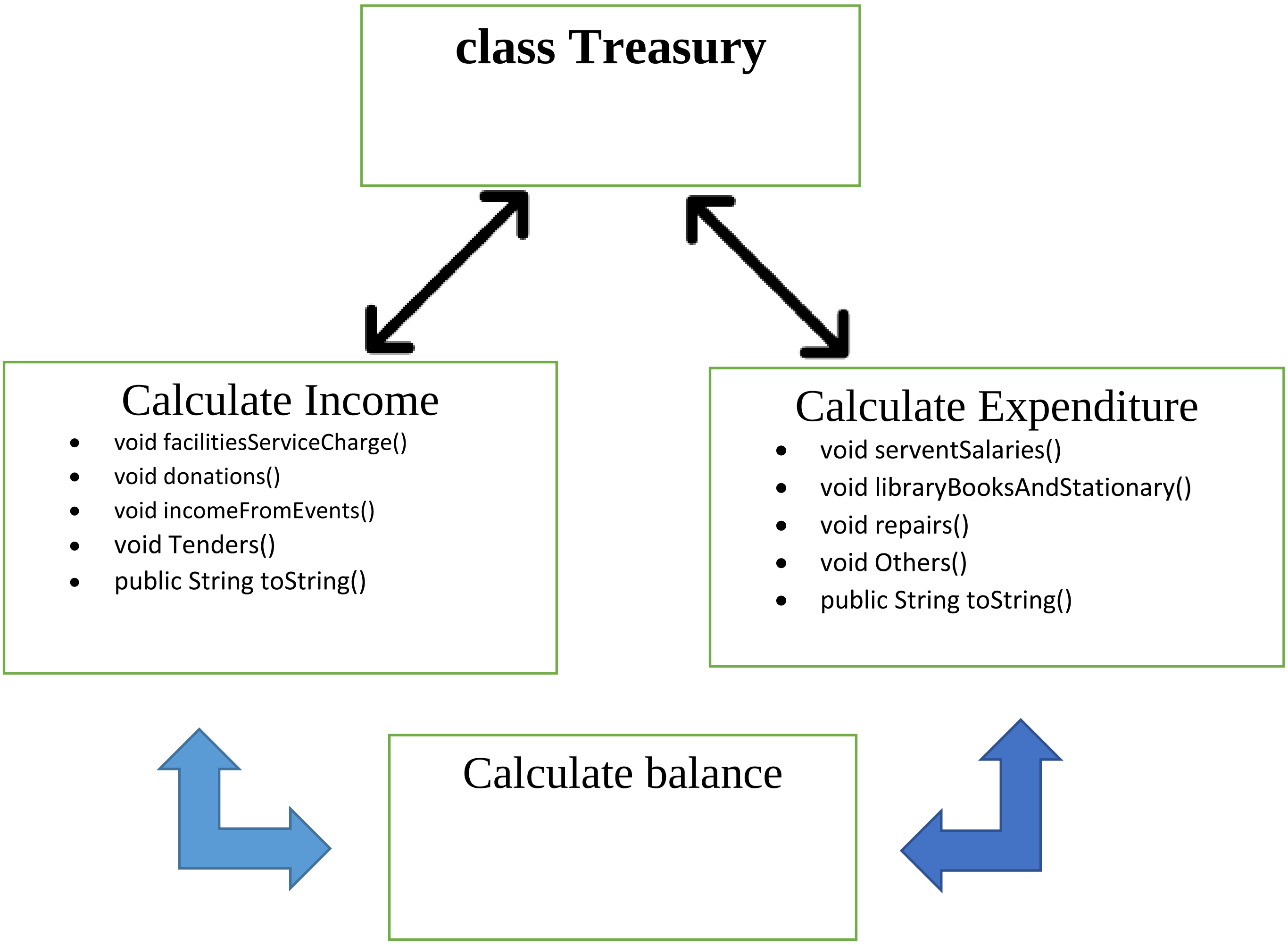
• When the number related to the sport is entered, the name of the sport, the coach, the instructor, the date and time of training are displayed.

• Use exception handling to avoid from the exceptions occurs when run time.



------------------------------------------------------------------------

## Class 07 – Treasury Class



class Treasury

Calculate Income

- void facilitiesServiceCharge()
- void donations()
- void incomeFromEvents()
- void Tenders()
- public String toString()

Calculate Expenditure

- void serventSalaries()
- void libraryBooksAndStationary()
- void repairs()
- void Others()
- public String toString()

Calculate balance

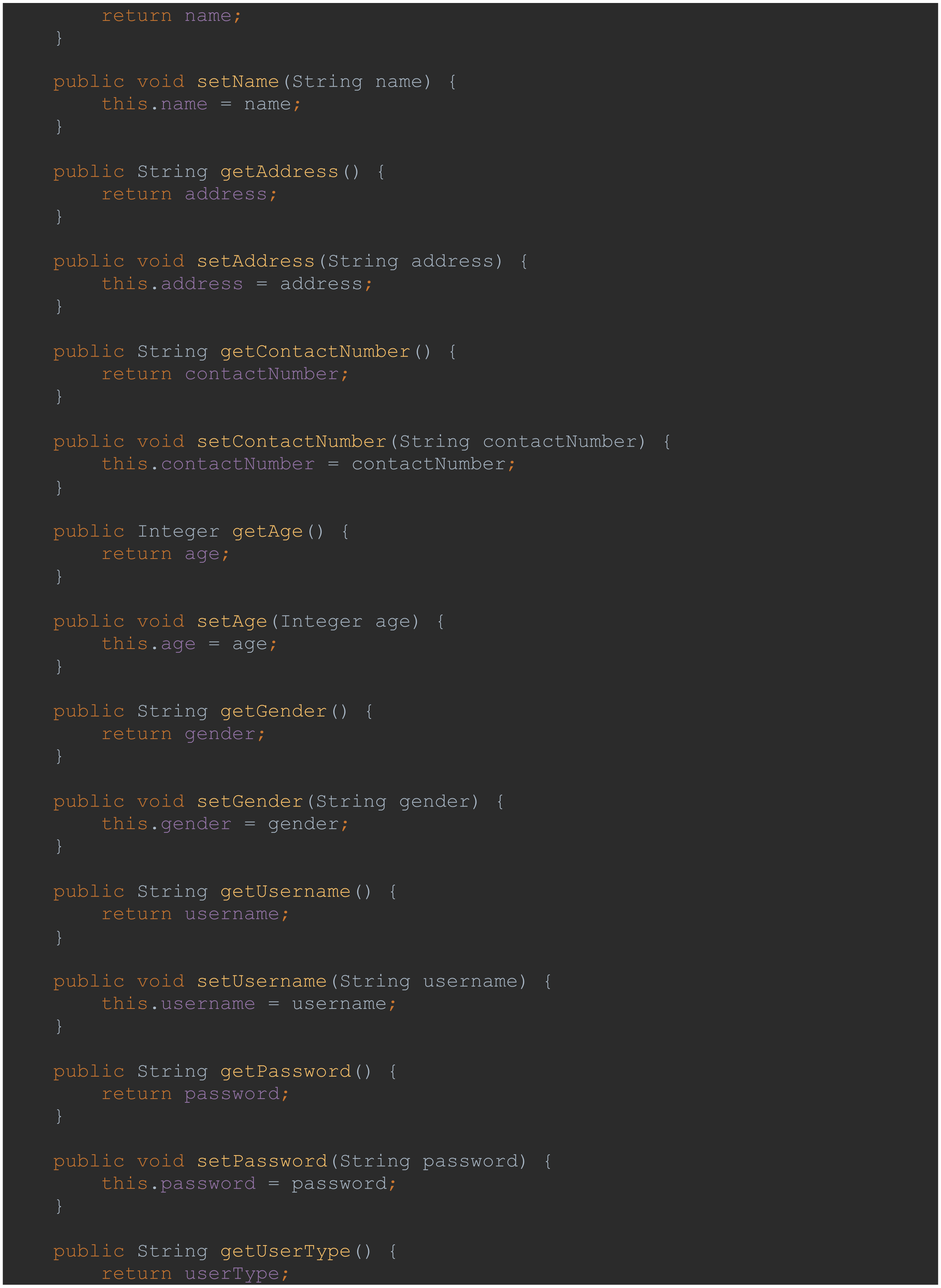
- The purpose of this class is to count the balance of the school treasury.
- Declare some data fields as private to increase the security of these data fields.
- Use inheritance by extending the parent class to two child classes.
- Used scanner in land package to get user inputs to calculations. The scanner variable is defined in the abstract class. Hence both the Income and Expenditure classes can use the scanner variable via inheritance.
- Incomes from canteen and bookshop are calculating according to the month. The user should enter the number of the month. The payments must be made before 15 in the month.
- The toString method has used to give and useful output.

- There are two types of servants. Permanent and temporary. Both are calculated to the servant’s salary.

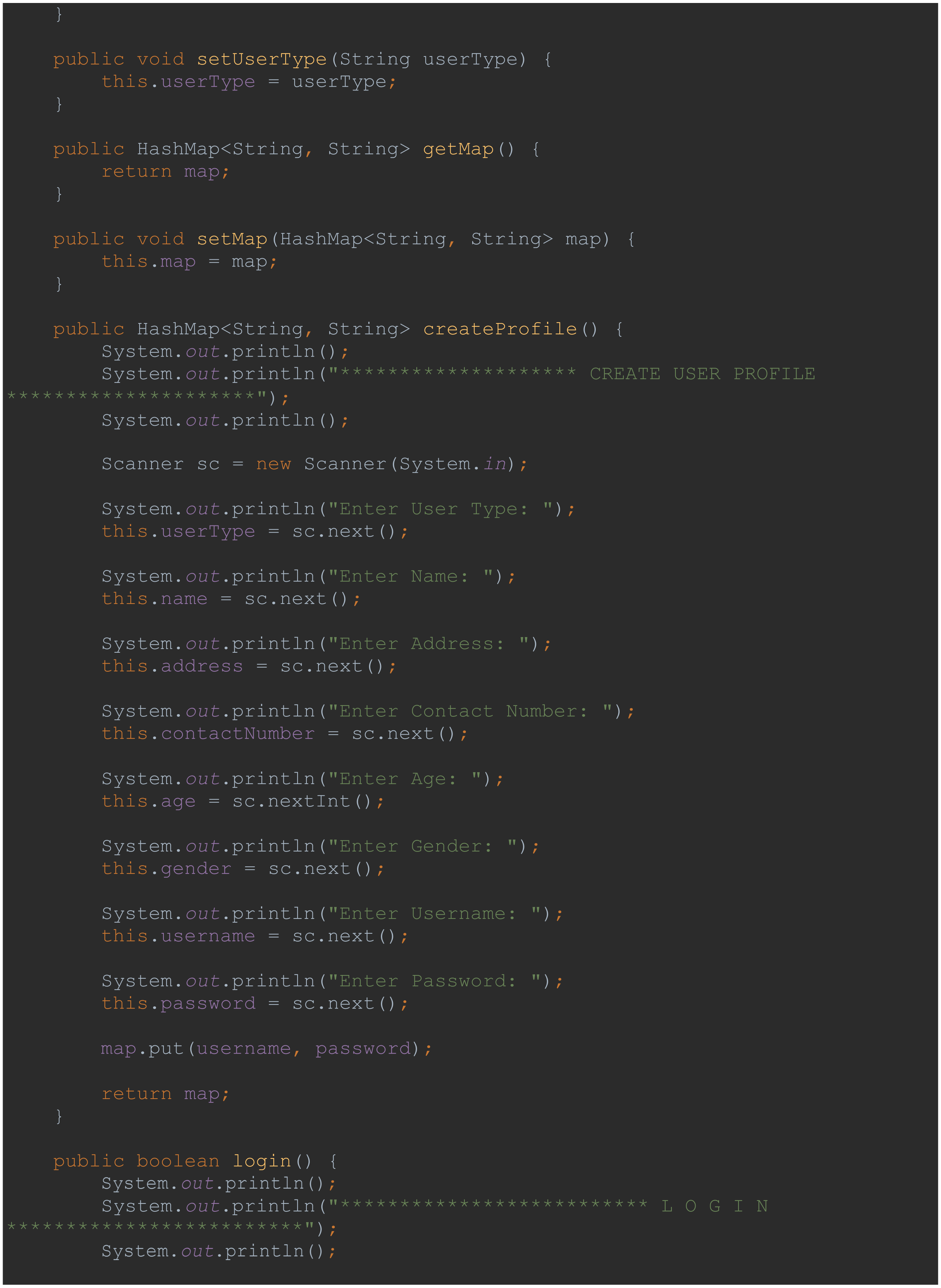
# SchoolManagement System

User

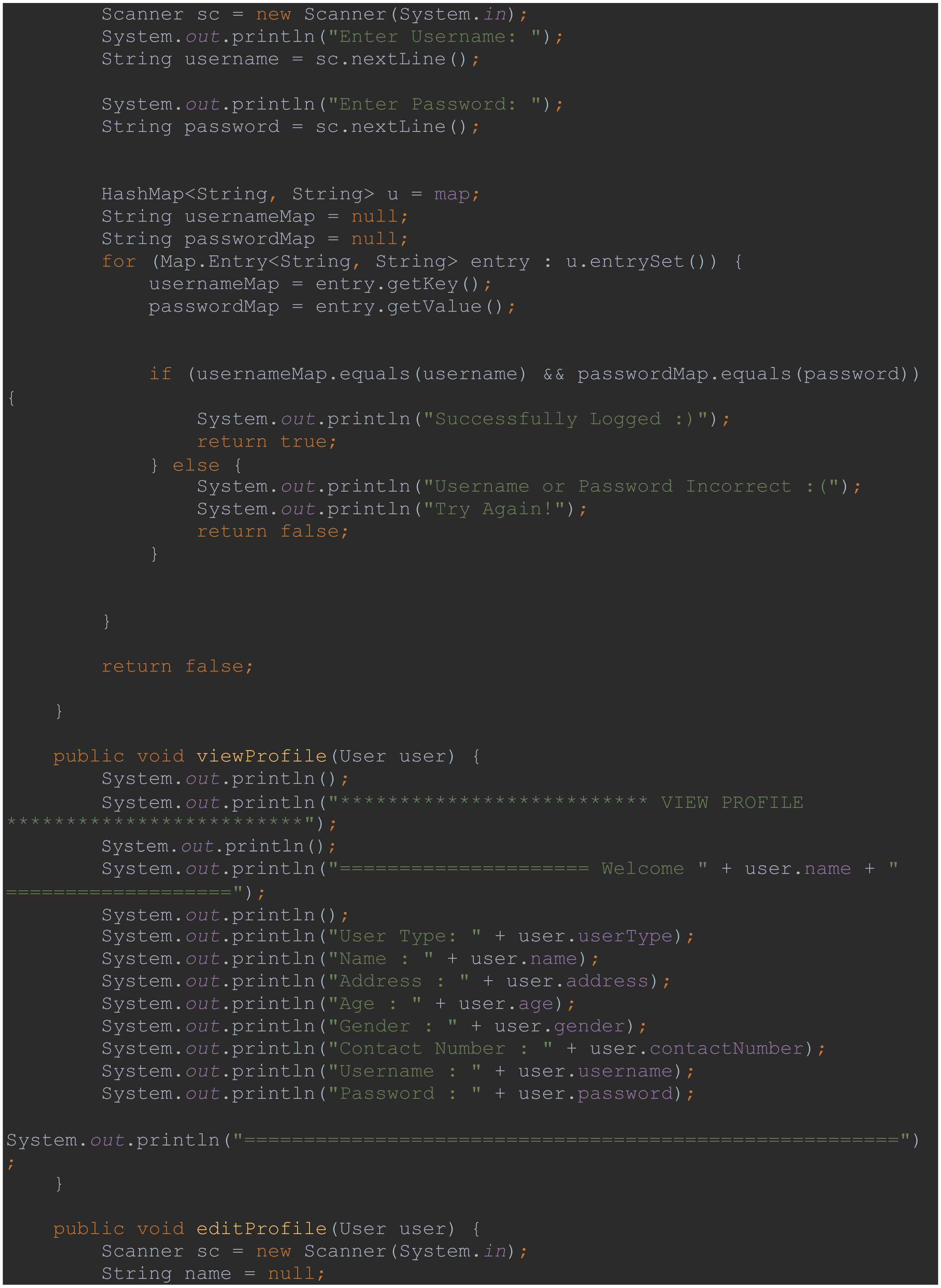
```
import java.util.*;

public class User {


    private Integer userId;
    private String name;
    private String address;
    private String contactNumber;
    private Integer age;
    private String gender;
    private String username;
    private String password;
    private String userType;

    HashMap<String, String> map = new HashMap<>();


    public User() {

    }

    public User(Integer userId, String name, String address, String
contactNumber, Integer age, String gender, String username, String password,
String userType) {
        this.userId = userId;
        this.name = name;
        this.address = address;
        this.contactNumber = contactNumber;
        this.age = age;
        this.gender = gender;
        this.username = username;
        this.password = password;
        this.userType = userType;
    }

    public User(String username, String password) {
        this.username = username;
        this.password = password;
    }

    public Integer getUserId() {
        return userId;
    }

    public void setUserId(Integer userId) {
        this.userId = userId;
    }

    public String getName() {
```



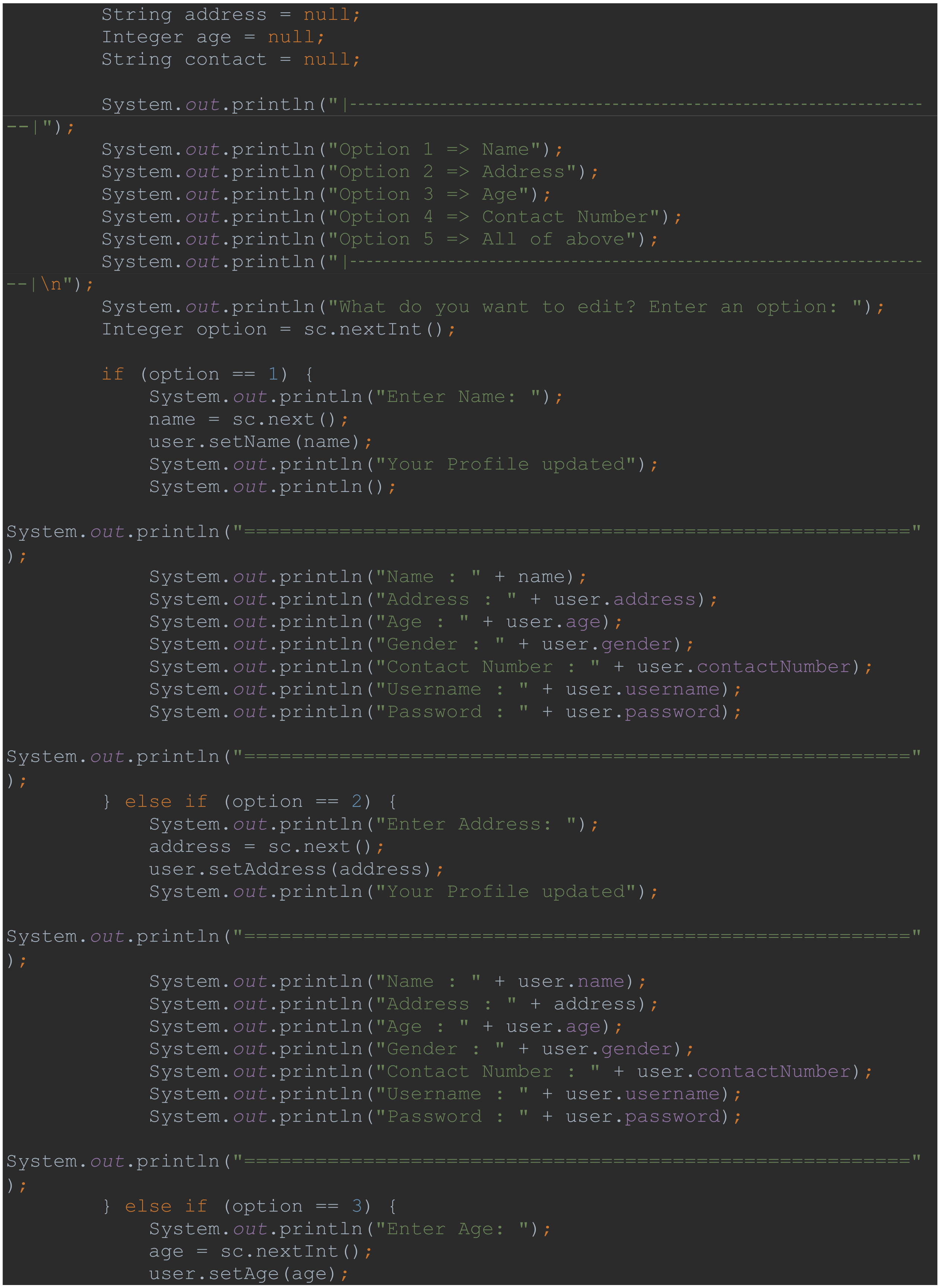
```
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getAddress() {
        return address;
    }

    public void setAddress(String address) {
        this.address = address;
    }

    public String getContactNumber() {
        return contactNumber;
    }

    public void setContactNumber(String contactNumber) {
        this.contactNumber = contactNumber;
    }

    public Integer getAge() {
        return age;
    }

    public void setAge(Integer age) {
        this.age = age;
    }

    public String getGender() {
        return gender;
    }

    public void setGender(String gender) {
        this.gender = gender;
    }

    public String getUsername() {
        return username;
    }

    public void setUsername(String username) {
        this.username = username;
    }

    public String getPassword() {
        return password;
    }

    public void setPassword(String password) {
        this.password = password;
    }

    public String getUserType() {
        return userType;
```



```
    }

    public void setUserType(String userType) {
        this.userType = userType;
    }

    public HashMap<String, String> getMap() {
        return map;
    }

    public void setMap(HashMap<String, String> map) {
        this.map = map;
    }

    public HashMap<String, String> createProfile() {
        System.out.println();
        System.out.println("******************** CREATE USER PROFILE
*********************");
        System.out.println();

        Scanner sc = new Scanner(System.in);

        System.out.println("Enter User Type: ");
        this.userType = sc.next();

        System.out.println("Enter Name: ");
        this.name = sc.next();

        System.out.println("Enter Address: ");
        this.address = sc.next();

        System.out.println("Enter Contact Number: ");
        this.contactNumber = sc.next();

        System.out.println("Enter Age: ");
        this.age = sc.nextInt();

        System.out.println("Enter Gender: ");
        this.gender = sc.next();

        System.out.println("Enter Username: ");
        this.username = sc.next();

        System.out.println("Enter Password: ");
        this.password = sc.next();

        map.put(username, password);

        return map;
    }

    public boolean login() {
        System.out.println();
        System.out.println("************************** L O G I N
*************************");
        System.out.println();
```



```
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter Username: ");
        String username = sc.nextLine();

        System.out.println("Enter Password: ");
        String password = sc.nextLine();


        HashMap<String, String> u = map;
        String usernameMap = null;
        String passwordMap = null;
        for (Map.Entry<String, String> entry : u.entrySet()) {
            usernameMap = entry.getKey();
            passwordMap = entry.getValue();


            if (usernameMap.equals(username) && passwordMap.equals(password))
{
                System.out.println("Successfully Logged :)");
                return true;
            } else {
                System.out.println("Username or Password Incorrect :(");
                System.out.println("Try Again!");
                return false;
            }


        }

        return false;

    }

    public void viewProfile(User user) {
        System.out.println();
        System.out.println("************************* VIEW PROFILE
************************");
        System.out.println();
        System.out.println("==================== Welcome " + user.name + "
==================");
        System.out.println();
        System.out.println("User Type: " + user.userType);
        System.out.println("Name : " + user.name);
        System.out.println("Address : " + user.address);
        System.out.println("Age : " + user.age);
        System.out.println("Gender : " + user.gender);
        System.out.println("Contact Number : " + user.contactNumber);
        System.out.println("Username : " + user.username);
        System.out.println("Password : " + user.password);

System.out.println("=====================================================")
;
    }

    public void editProfile(User user) {
        Scanner sc = new Scanner(System.in);
        String name = null;
```

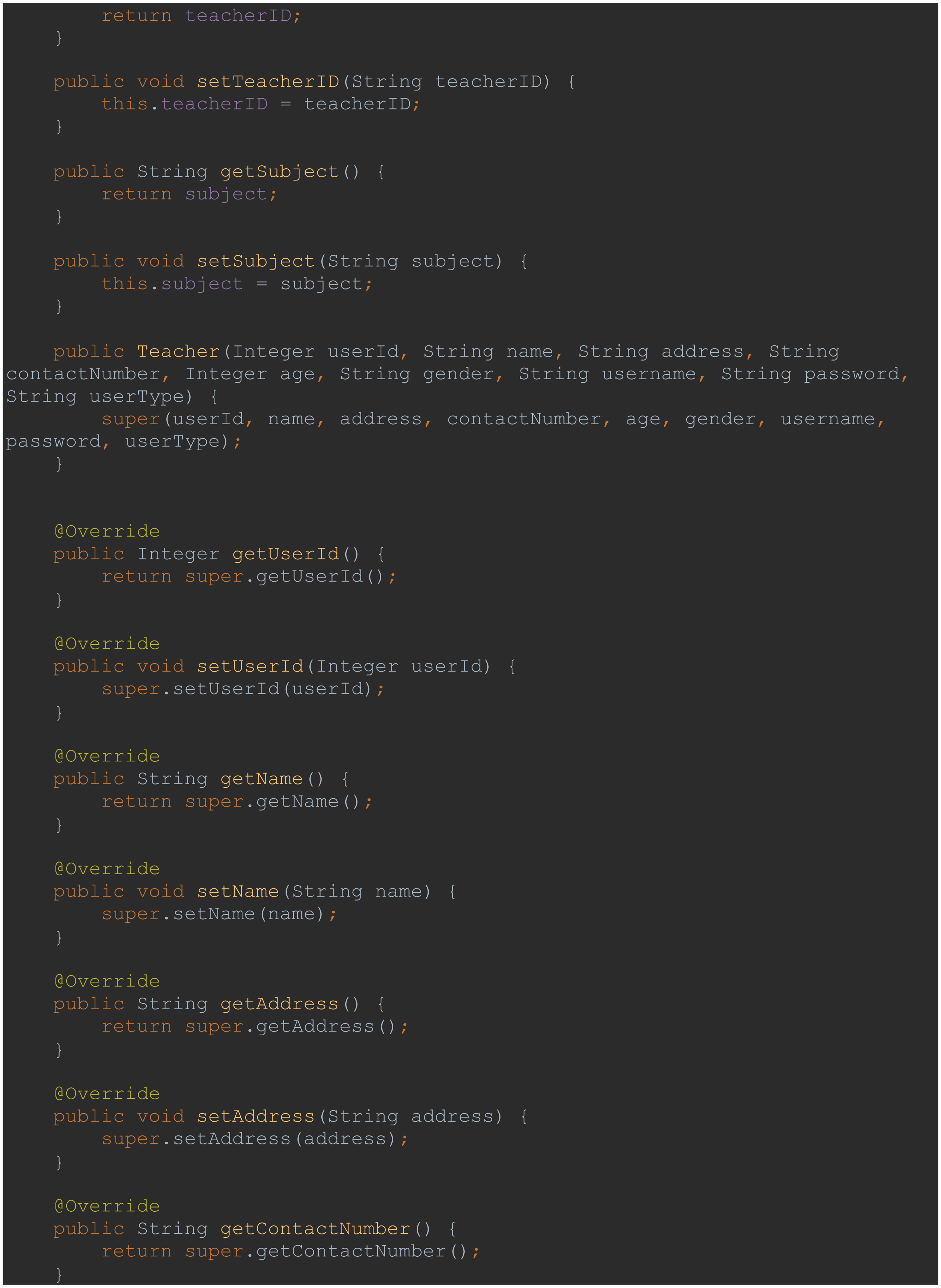


```
        String address = null;
        Integer age = null;
        String contact = null;

        System.out.println("|------------------------------------------------
--|");
        System.out.println("Option 1 => Name");
        System.out.println("Option 2 => Address");
        System.out.println("Option 3 => Age");
        System.out.println("Option 4 => Contact Number");
        System.out.println("Option 5 => All of above");
        System.out.println("|------------------------------------------------
--|\n");
        System.out.println("What do you want to edit? Enter an option: ");
        Integer option = sc.nextInt();

        if (option == 1) {
            System.out.println("Enter Name: ");
            name = sc.next();
            user.setName(name);
            System.out.println("Your Profile updated");
            System.out.println();

System.out.println("========================================================"
);
            System.out.println("Name : " + name);
            System.out.println("Address : " + user.address);
            System.out.println("Age : " + user.age);
            System.out.println("Gender : " + user.gender);
            System.out.println("Contact Number : " + user.contactNumber);
            System.out.println("Username : " + user.username);
            System.out.println("Password : " + user.password);

System.out.println("========================================================"
);
        } else if (option == 2) {
            System.out.println("Enter Address: ");
            address = sc.next();
            user.setAddress(address);
            System.out.println("Your Profile updated");

System.out.println("========================================================"
);
            System.out.println("Name : " + user.name);
            System.out.println("Address : " + address);
            System.out.println("Age : " + user.age);
            System.out.println("Gender : " + user.gender);
            System.out.println("Contact Number : " + user.contactNumber);
            System.out.println("Username : " + user.username);
            System.out.println("Password : " + user.password);

System.out.println("========================================================"
);
        } else if (option == 3) {
            System.out.println("Enter Age: ");
            age = sc.nextInt();
            user.setAge(age);
            System.out.println("Your Profile updated");

System.out.println("========================================================"
);
            System.out.println("Name : " + user.name);
            System.out.println("Address : " + user.address);
            System.out.println("Age : " + age);
            System.out.println("Gender : " + user.gender);
            System.out.println("Contact Number : " + user.contactNumber);
            System.out.println("Username : " + user.username);
            System.out.println("Password : " + user.password);

System.out.println("========================================================"
);
        } else if (option == 4) {
            System.out.println("Enter Contact Number: ");
            contact = sc.next();
            user.setContactNumber(contact);
            System.out.println("Your Profile updated");

System.out.println("========================================================"
);
            System.out.println("Name : " + user.name);
            System.out.println("Address : " + user.address);
            System.out.println("Age : " + user.age);
            System.out.println("Gender : " + user.gender);
            System.out.println("Contact Number : " + contact);
            System.out.println("Username : " + user.username);
            System.out.println("Password : " + user.password);

System.out.println("========================================================"
);
        } else if (option == 5) {
            System.out.println("Enter Name: ");
            name = sc.next();

            System.out.println("Enter Address: ");
            address = sc.next();

            System.out.println("Enter Age: ");
            age = sc.nextInt();

            System.out.println("Enter Contact Number: ");
            contact = sc.next();


            user.setName(name);
            user.setAddress(address);
            user.setAge(age);
            user.setContactNumber(contact);

            System.out.println("Your Profile updated");

System.out.println("********************************************************"
);
            System.out.println("Name : " + name);
            System.out.println("Address : " + address);
```

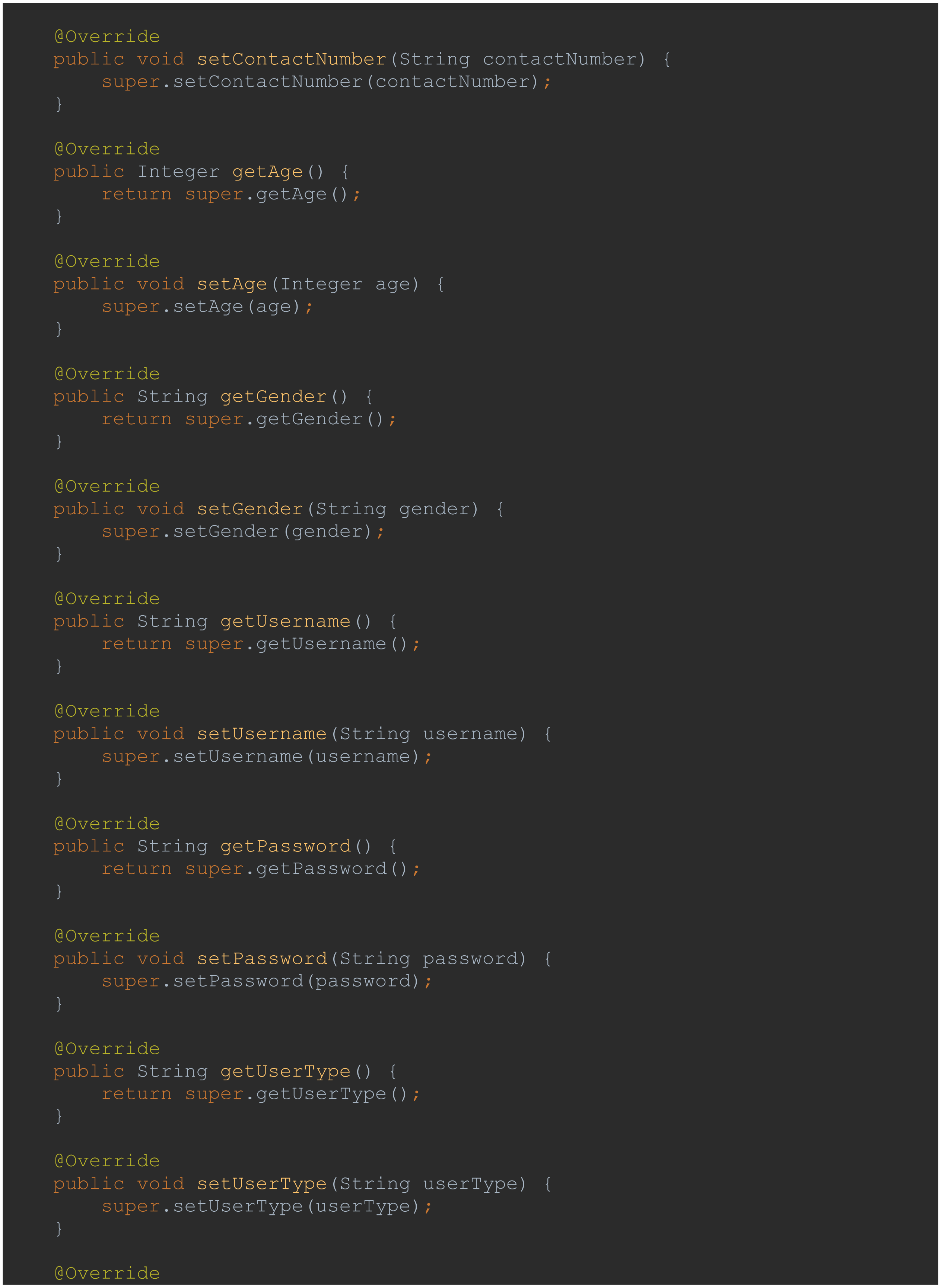
```
			System.out.println("Age : " + age);
			System.out.println("Gender : " + user.gender);
			System.out.println("Contact Number : " + contact);
			System.out.println("Username : " + user.username);
			System.out.println("Password : " + user.password);

System.out.println("******************************************************************* ");
		} else {
			System.out.println("You entered wrong number.");
		}


	}
}
```

Teacher

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

public class Teacher extends	User{

	String teacherID;
	String subject;

	public Teacher(){

	}

	public Teacher(String teacherID, String subject) {
		this.teacherID = teacherID;
		this.subject = subject;
	}

	public Teacher(Integer userId, String name, String address, String
contactNumber, Integer age, String gender, String username, String password, String
userType, String teacherID, String subject) {
		super(userId, name, address, contactNumber, age, gender, username,
password, userType);
		this.teacherID = teacherID;
		this.subject = subject;
	}

	public Teacher(String username, String password, String teacherID, String
subject) {
		super(username, password);
```



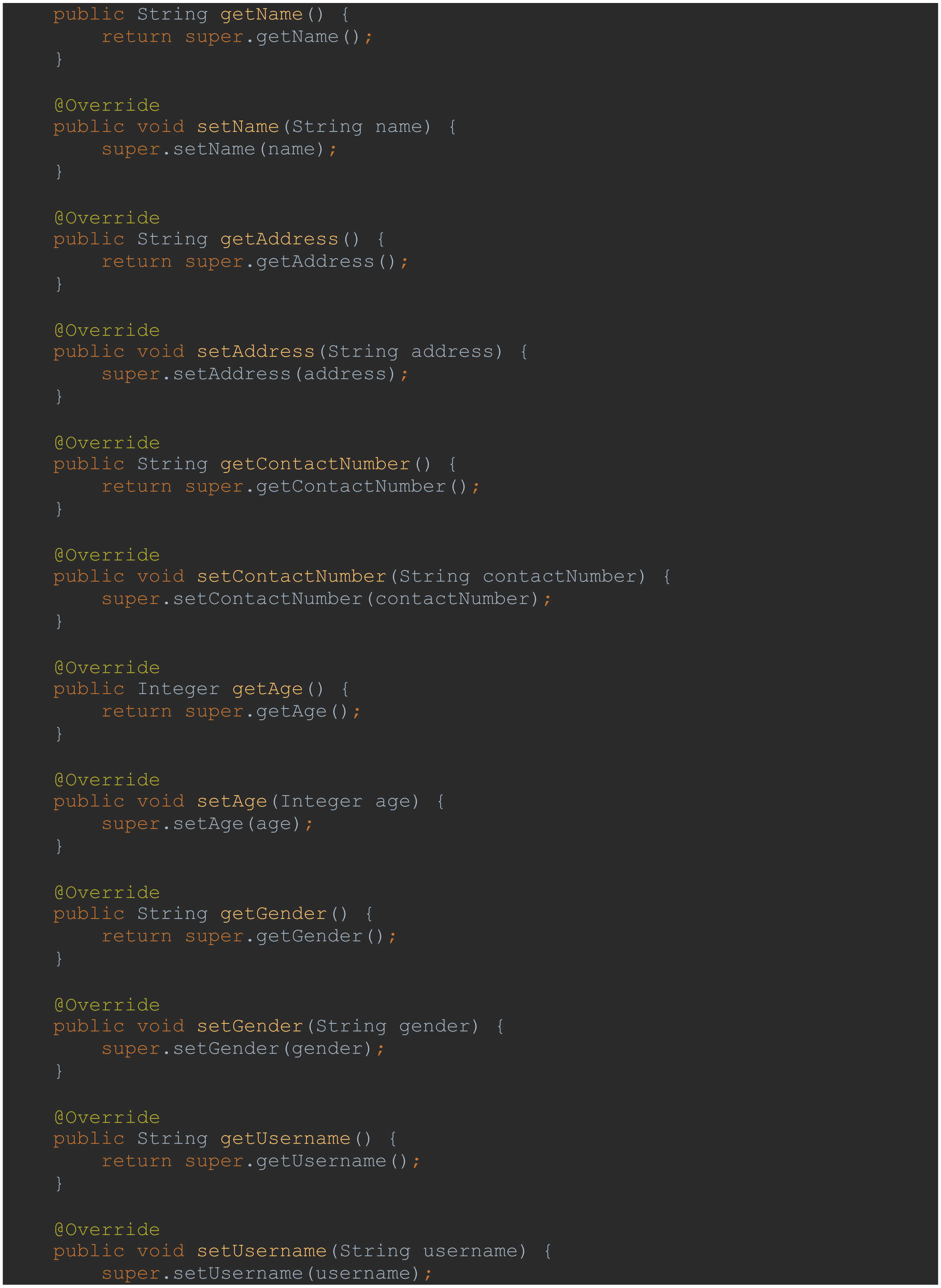
```
        return teacherID;
    }

    public void setTeacherID(String teacherID) {
        this.teacherID = teacherID;
    }

    public String getSubject() {
        return subject;
    }

    public void setSubject(String subject) {
        this.subject = subject;
    }

    public Teacher(Integer userId, String name, String address, String
contactNumber, Integer age, String gender, String username, String password,
String userType) {
        super(userId, name, address, contactNumber, age, gender, username,
password, userType);
    }


    @Override
    public Integer getUserId() {
        return super.getUserId();
    }

    @Override
    public void setUserId(Integer userId) {
        super.setUserId(userId);
    }

    @Override
    public String getName() {
        return super.getName();
    }

    @Override
    public void setName(String name) {
        super.setName(name);
    }

    @Override
    public String getAddress() {
        return super.getAddress();
    }

    @Override
    public void setAddress(String address) {
        super.setAddress(address);
    }

    @Override
    public String getContactNumber() {
        return super.getContactNumber();
    }
```



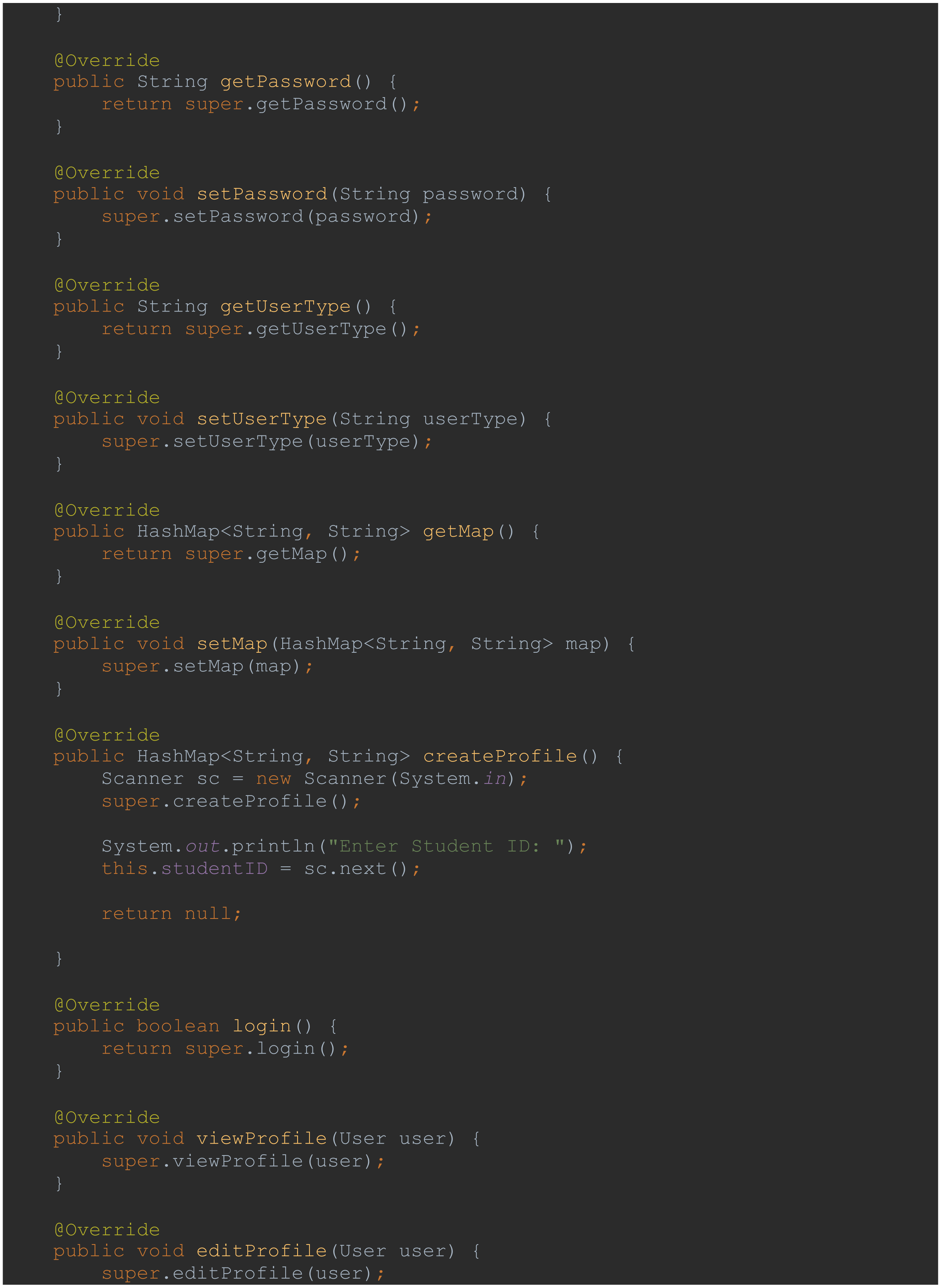
```java
    @Override
    public void setContactNumber(String contactNumber) {
        super.setContactNumber(contactNumber);
    }

    @Override
    public Integer getAge() {
        return super.getAge();
    }

    @Override
    public void setAge(Integer age) {
        super.setAge(age);
    }

    @Override
    public String getGender() {
        return super.getGender();
    }

    @Override
    public void setGender(String gender) {
        super.setGender(gender);
    }

    @Override
    public String getUsername() {
        return super.getUsername();
    }

    @Override
    public void setUsername(String username) {
        super.setUsername(username);
    }

    @Override
    public String getPassword() {
        return super.getPassword();
    }

    @Override
    public void setPassword(String password) {
        super.setPassword(password);
    }

    @Override
    public String getUserType() {
        return super.getUserType();
    }

    @Override
    public void setUserType(String userType) {
        super.setUserType(userType);
    }

    @Override
    public HashMap<String, String> getMap() {
        return super.getMap();
    }

    @Override
    public void setMap(HashMap<String, String> map) {
        super.setMap(map);
    }

    @Override
    public HashMap<String, String> createProfile() {
        Scanner sc = new Scanner(System.in);
        super.createProfile();

        System.out.println("Enter Teacher ID: ");
        this.teacherID = sc.next();

        System.out.println("Enter Subject: ");
        this.subject = sc.next();


        return null;

    }

    @Override
    public boolean login() {
        return super.login();
    }

    @Override
    public void viewProfile(User user) {
        super.viewProfile(user);
    }
```

Student

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

public class Student extends        User{

    String studentID;
    }

    public Student(String studentID/*String subject*/) {
        this.studentID = studentID;

    }

    public Student(Integer userId, String name, String address, String
contactNumber, Integer age, String gender, String username, String password,
String userType, String studentID, String subject) {
        super(userId, name, address, contactNumber, age, gender, username,
password, userType);
        this.studentID = studentID;

    }

    public Student(String username, String password, String studentID, String
subject) {
        super(username, password);
        this.studentID = studentID;
    }

    public String getStudentID() {
        return studentID;
    }

    public void setStudentID(String studentID) {
        this.studentID = studentID;
    }


    public Student(Integer userId, String name, String address, String
contactNumber, Integer age, String gender, String username, String password,
String userType) {
        super(userId, name, address, contactNumber, age, gender, username,
password, userType);
    }

    public List<Student> addStudents(){
        List<Student> students = new ArrayList<>();
        Student student1 = new Student();

        students.add(student1);
        return students;
    }

    @Override
    public Integer getUserId() {
        return super.getUserId();
    }

    @Override
    public void setUserId(Integer userId) {
        super.setUserId(userId);
    }

    @Override
```



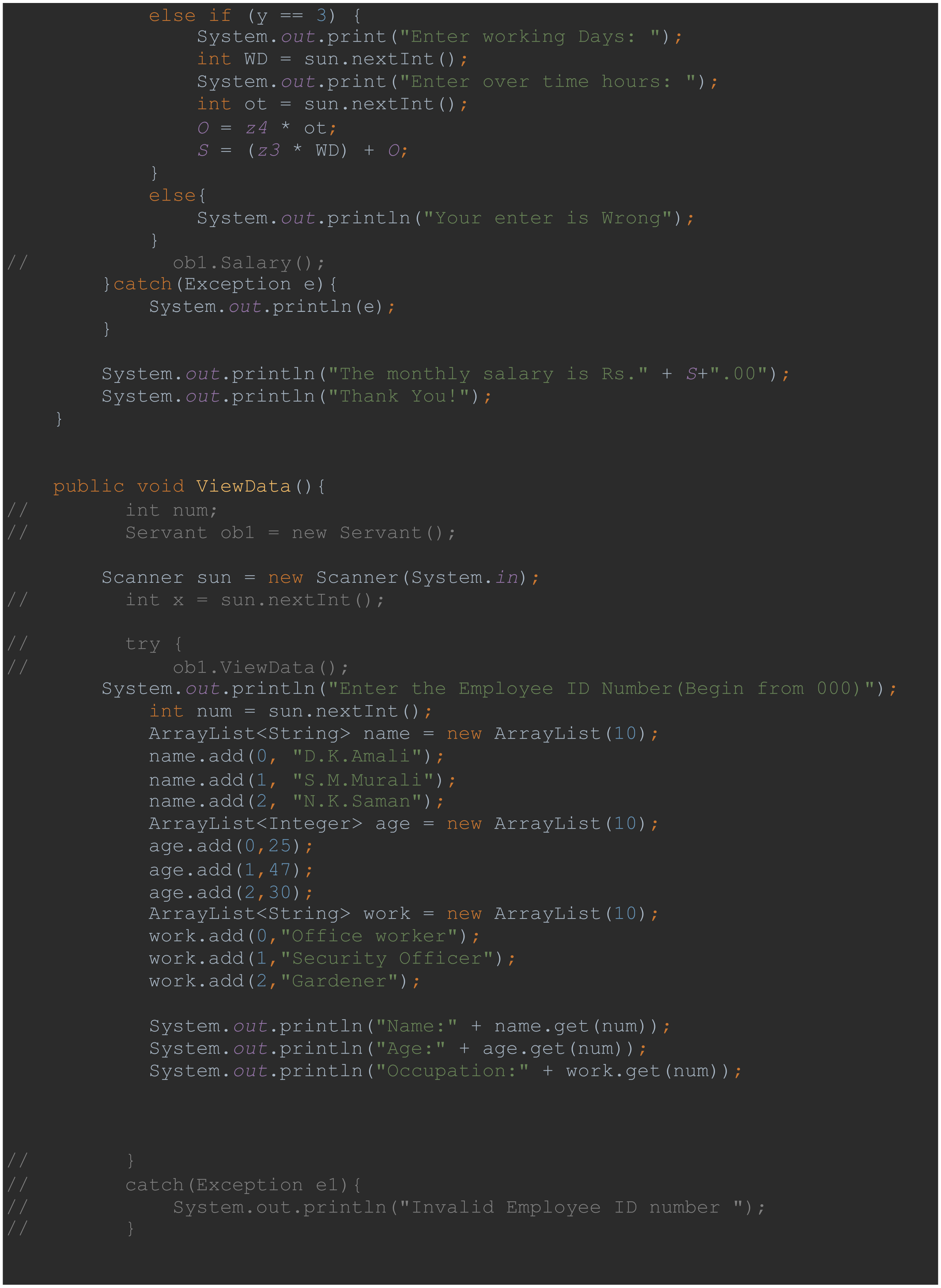
```
    public String getName() {
        return super.getName();
    }

    @Override
    public void setName(String name) {
        super.setName(name);
    }

    @Override
    public String getAddress() {
        return super.getAddress();
    }

    @Override
    public void setAddress(String address) {
        super.setAddress(address);
    }

    @Override
    public String getContactNumber() {
        return super.getContactNumber();
    }

    @Override
    public void setContactNumber(String contactNumber) {
        super.setContactNumber(contactNumber);
    }

    @Override
    public Integer getAge() {
        return super.getAge();
    }

    @Override
    public void setAge(Integer age) {
        super.setAge(age);
    }

    @Override
    public String getGender() {
        return super.getGender();
    }

    @Override
    public void setGender(String gender) {
        super.setGender(gender);
    }

    @Override
    public String getUsername() {
        return super.getUsername();
    }

    @Override
    public void setUsername(String username) {
        super.setUsername(username);
```



```
    }

    @Override
    public String getPassword() {
        return super.getPassword();
    }

    @Override
    public void setPassword(String password) {
        super.setPassword(password);
    }

    @Override
    public String getUserType() {
        return super.getUserType();
    }

    @Override
    public void setUserType(String userType) {
        super.setUserType(userType);
    }

    @Override
    public HashMap<String, String> getMap() {
        return super.getMap();
    }

    @Override
    public void setMap(HashMap<String, String> map) {
        super.setMap(map);
    }

    @Override
    public HashMap<String, String> createProfile() {
        Scanner sc = new Scanner(System.in);
        super.createProfile();

        System.out.println("Enter Student ID: ");
        this.studentID = sc.next();

        return null;

    }

    @Override
    public boolean login() {
        return super.login();
    }

    @Override
    public void viewProfile(User user) {
        super.viewProfile(user);
    }

    @Override
    public void editProfile(User user) {
        super.editProfile(user);
    }
}
```

Servant

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

public class Servant {


    final static int z1 = 1700;
    final static int z2 = 2000;
    final static int z3 = 1500;
    final static int z4 = 100;
    static int S,O;

    Servant(){
        System.out.println("If you want Calculate Salary enter number 1");
        System.out.println("If you want to view data enter number 2");
    }

    public void Salary() {
//        int x;
//        int y,WD,ot;
//        Servant ob1 = new Servant();

        Scanner sun = new Scanner(System.in);
//        int x = sun.nextInt();

        System.out.println("Enter Occupation: ");
        System.out.println("1 for Security Officer: ");
        System.out.println("2 for Office Worker: ");
        System.out.println("3 for Gardener: ");
        int y = sun.nextInt();

        try {

            if (y == 1) {
                System.out.print("Enter working Days: ");
                int WD = sun.nextInt();
                System.out.print("Enter over time hours: ");
                int ot = sun.nextInt();
                O = z4 * ot;
                S = (z1 * WD) + O;
            }
            else if (y == 2) {
                System.out.print("Enter working Days: ");
                int WD = sun.nextInt();
                System.out.print("Enter over time hours: ");
                int ot = sun.nextInt();
                O = z4 * ot;
                S = (z2 * WD) + O;
//                ob1.Salary();
            }
```

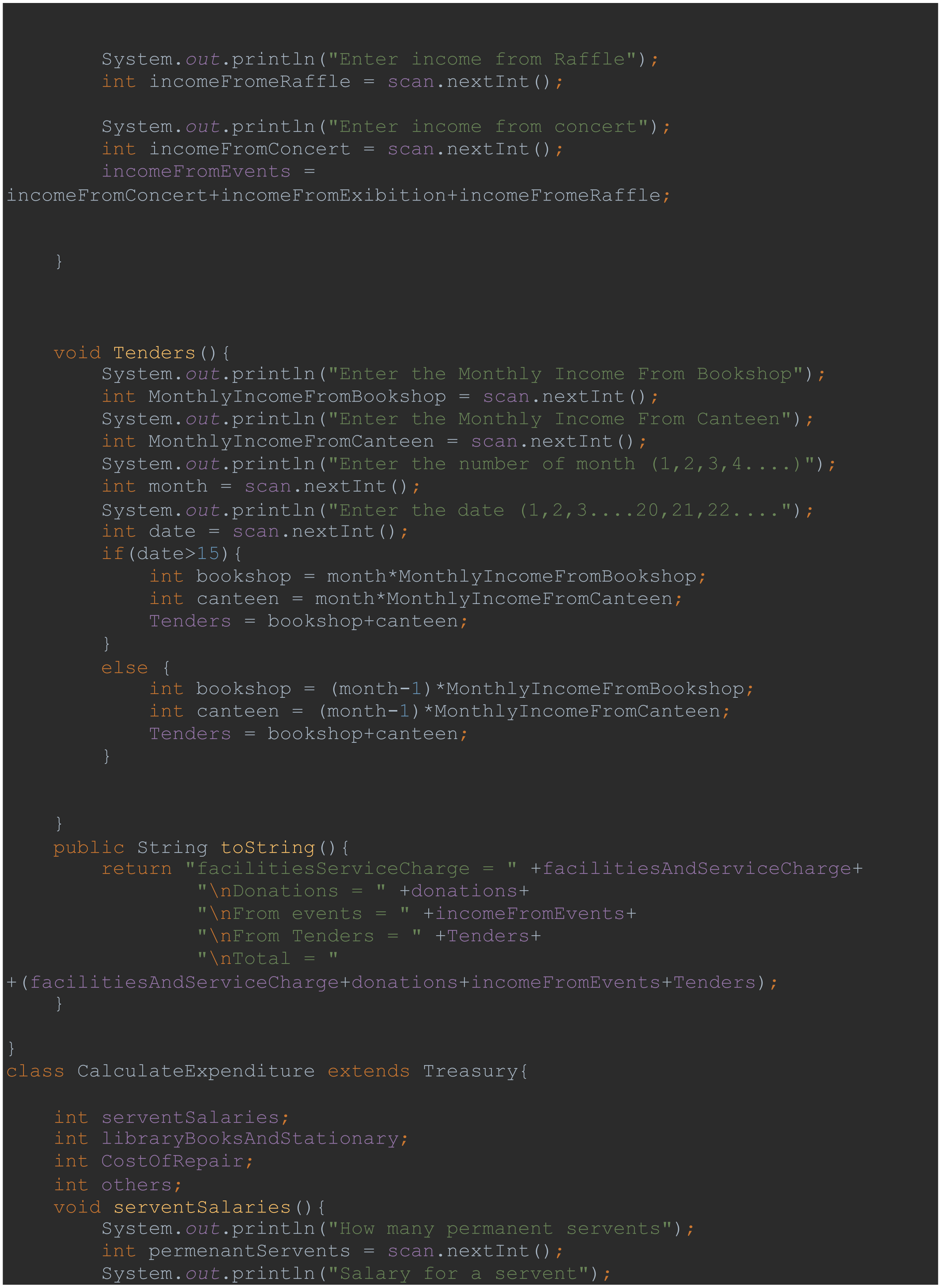


```
            else if (y == 3) {
                System.out.print("Enter working Days: ");
                int WD = sun.nextInt();
                System.out.print("Enter over time hours: ");
                int ot = sun.nextInt();
                O = z4 * ot;
                S = (z3 * WD) + O;
            }
            else{
                System.out.println("Your enter is Wrong");
            }
//            ob1.Salary();
        }catch(Exception e){
            System.out.println(e);
        }

        System.out.println("The monthly salary is Rs." + S+".00");
        System.out.println("Thank You!");
    }


    public void ViewData(){
//        int num;
//        Servant ob1 = new Servant();

        Scanner sun = new Scanner(System.in);
//        int x = sun.nextInt();

//        try {
//            ob1.ViewData();
        System.out.println("Enter the Employee ID Number(Begin from 000)");
            int num = sun.nextInt();
            ArrayList<String> name = new ArrayList(10);
            name.add(0, "D.K.Amali");
            name.add(1, "S.M.Murali");
            name.add(2, "N.K.Saman");
            ArrayList<Integer> age = new ArrayList(10);
            age.add(0,25);
            age.add(1,47);
            age.add(2,30);
            ArrayList<String> work = new ArrayList(10);
            work.add(0,"Office worker");
            work.add(1,"Security Officer");
            work.add(2,"Gardener");

            System.out.println("Name:" + name.get(num));
            System.out.println("Age:" + age.get(num));
            System.out.println("Occupation:" + work.get(num));



//        }
//        catch(Exception e1){
//            System.out.println("Invalid Employee ID number ");
//        }
```

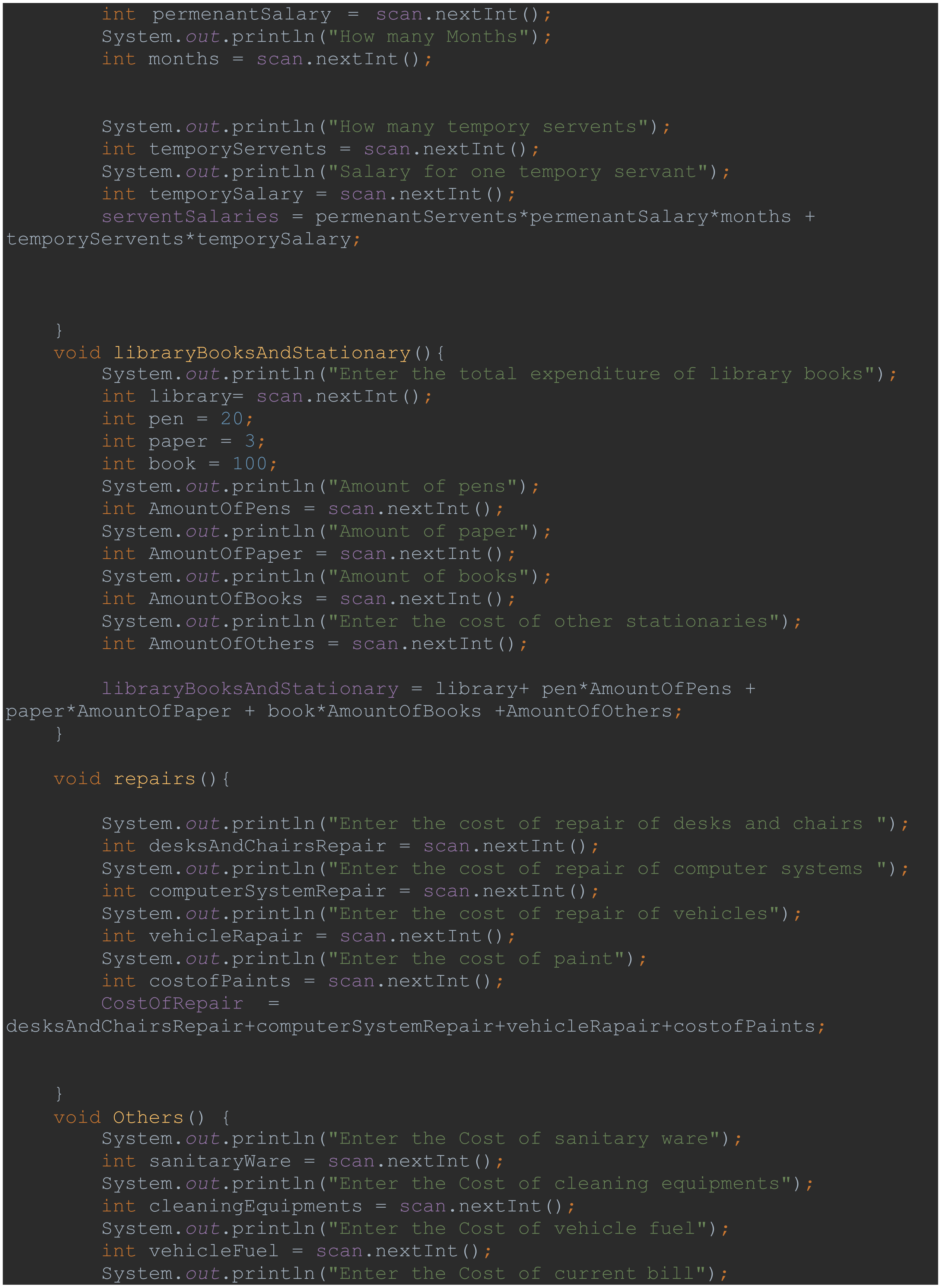
```
    }

}
```

Treasury

```
/*
Purpose : To get information about income and expenditures of a school within
a particular time period


*/
import java.util.InputMismatchException;
import java.util.Scanner;
abstract class Treasury{
    Scanner scan = new Scanner(System.in);
}

class CalculateIncome extends Treasury{
    Scanner scan = new Scanner(System.in);
    int facilitiesAndServiceCharge;
    int donations;
    int incomeFromEvents;
    int Tenders;
    void facilitiesServiceCharge(){
        System.out.println("How many students?");
        int students = scan.nextInt();
        System.out.println("Charge per student");
        int ChargePerStudent = scan.nextInt();
        facilitiesAndServiceCharge = students*ChargePerStudent;
    }
    void donations(){


        System.out.println("Donations from foriegn organizations");
        int foriegnOrganizations = scan.nextInt();
        System.out.println("Donations from local organizations");
        int localOrganizations = scan.nextInt();
        System.out.println("Donations from Old Boys Society");
        int OBDonations = scan.nextInt();
        System.out.println("Donations form Others");
        int OtherDonations = scan.nextInt();
        donations = OBDonations + OtherDonations + foriegnOrganizations +
localOrganizations;


    }
    void incomeFromEvents(){

        System.out.println("Enter income from exibition");
        int incomeFromExibition = scan.nextInt();
```



```
        System.out.println("Enter income from Raffle");
        int incomeFromeRaffle = scan.nextInt();

        System.out.println("Enter income from concert");
        int incomeFromConcert = scan.nextInt();
        incomeFromEvents =
incomeFromConcert+incomeFromExibition+incomeFromeRaffle;


    }



    void Tenders(){
        System.out.println("Enter the Monthly Income From Bookshop");
        int MonthlyIncomeFromBookshop = scan.nextInt();
        System.out.println("Enter the Monthly Income From Canteen");
        int MonthlyIncomeFromCanteen = scan.nextInt();
        System.out.println("Enter the number of month (1,2,3,4....)");
        int month = scan.nextInt();
        System.out.println("Enter the date (1,2,3....20,21,22....");
        int date = scan.nextInt();
        if(date>15){
            int bookshop = month*MonthlyIncomeFromBookshop;
            int canteen = month*MonthlyIncomeFromCanteen;
            Tenders = bookshop+canteen;
        }
        else {
            int bookshop = (month-1)*MonthlyIncomeFromBookshop;
            int canteen = (month-1)*MonthlyIncomeFromCanteen;
            Tenders = bookshop+canteen;
        }


    }
    public String toString(){
        return "facilitiesServiceCharge = " +facilitiesAndServiceCharge+
                "\nDonations = " +donations+
                "\nFrom events = " +incomeFromEvents+
                "\nFrom Tenders = " +Tenders+
                "\nTotal = "
+(facilitiesAndServiceCharge+donations+incomeFromEvents+Tenders);
    }

}
class CalculateExpenditure extends Treasury{

    int serventSalaries;
    int libraryBooksAndStationary;
    int CostOfRepair;
    int others;
    void serventSalaries(){
        System.out.println("How many permanent servents");
        int permenantServents = scan.nextInt();
        System.out.println("Salary for a servent");
```



```
        int permenantSalary = scan.nextInt();
        System.out.println("How many Months");
        int months = scan.nextInt();


        System.out.println("How many tempory servents");
        int temporyServents = scan.nextInt();
        System.out.println("Salary for one tempory servant");
        int temporySalary = scan.nextInt();
        serventSalaries = permenantServents*permenantSalary*months +
temporyServents*temporySalary;



    }
    void libraryBooksAndStationary(){
        System.out.println("Enter the total expenditure of library books");
        int library= scan.nextInt();
        int pen = 20;
        int paper = 3;
        int book = 100;
        System.out.println("Amount of pens");
        int AmountOfPens = scan.nextInt();
        System.out.println("Amount of paper");
        int AmountOfPaper = scan.nextInt();
        System.out.println("Amount of books");
        int AmountOfBooks = scan.nextInt();
        System.out.println("Enter the cost of other stationaries");
        int AmountOfOthers = scan.nextInt();

        libraryBooksAndStationary = library+ pen*AmountOfPens +
paper*AmountOfPaper + book*AmountOfBooks +AmountOfOthers;
    }

    void repairs(){

        System.out.println("Enter the cost of repair of desks and chairs ");
        int desksAndChairsRepair = scan.nextInt();
        System.out.println("Enter the cost of repair of computer systems ");
        int computerSystemRepair = scan.nextInt();
        System.out.println("Enter the cost of repair of vehicles");
        int vehicleRapair = scan.nextInt();
        System.out.println("Enter the cost of paint");
        int costofPaints = scan.nextInt();
        CostOfRepair  =
desksAndChairsRepair+computerSystemRepair+vehicleRapair+costofPaints;


    }
    void Others() {
        System.out.println("Enter the Cost of sanitary ware");
        int sanitaryWare = scan.nextInt();
        System.out.println("Enter the Cost of cleaning equipments");
        int cleaningEquipments = scan.nextInt();
        System.out.println("Enter the Cost of vehicle fuel");
        int vehicleFuel = scan.nextInt();
        System.out.println("Enter the Cost of current bill");
```

```
          int currentBill = scan.nextInt(); System.out.println("Enter
          the Cost of water bill"); int waterBill = scan.nextInt();
          System.out.println("Enter other Expenditures");
          int otherExpenditures = scan.nextInt();
          others =
sanitaryWare+cleaningEquipments+vehicleFuel+currentBill+waterBill+otherExpend
itures;
     }
     public String toString(){
          return "ServentSalaries = " +serventSalaries+
                    "\nLibrary Books And Stationary =
"+libraryBooksAndStationary+
                    "\nCost of repair = " +CostOfRepair+
                    "\nOthers = " +others+
                    "\nTotal Expenditures = "
+(serventSalaries+libraryBooksAndStationary+CostOfRepair+others);
     }
```

Marks

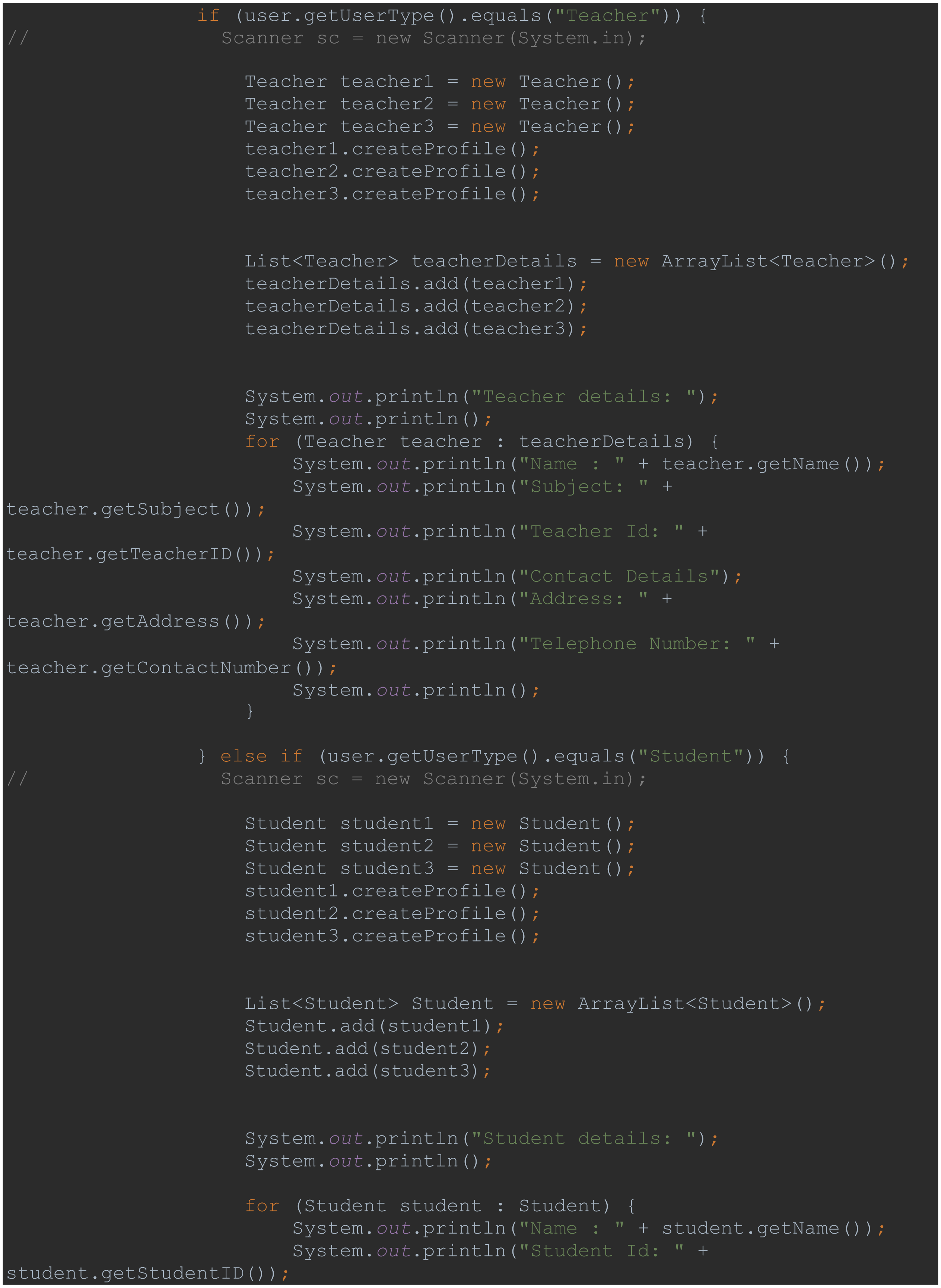
```
import java.util.Scanner;

import java.util.*;

public class Marks {
     // Recursively computes average of a[]
     static double findTotal(int a[], int i, int n) {
          // Last element if
          (i == n - 1)
               return a[i];
          // When index is 0, divide sum computed so
          // far by n. if (i
          == 0)
               return ((a[i] + findTotal(a, i + 1, n)) / n);
          // Compute sum
          return (a[i] + findTotal(a, i + 1, n));
     }

     // Function that returns the average of an array. static
     double findPercentage(int a[], int n) {
          return findTotal(a, 0, n);
     }
     // Main driver method
```

Sport

```
public class Sports {
    public int x, y;
    public String w1, w2, w3, w4, w5;

    Sports() {
        System.out.println("Enter Number 1 for Input New Data:");
        System.out.println("Enter Number 2 for View Data:");
    }

    public void ID() {

    }

    public void VD() {
        System.out.println("Enter the ID number of the sport(THE ID NUMBER IS
BEGIN FROM 000)");
    }
```

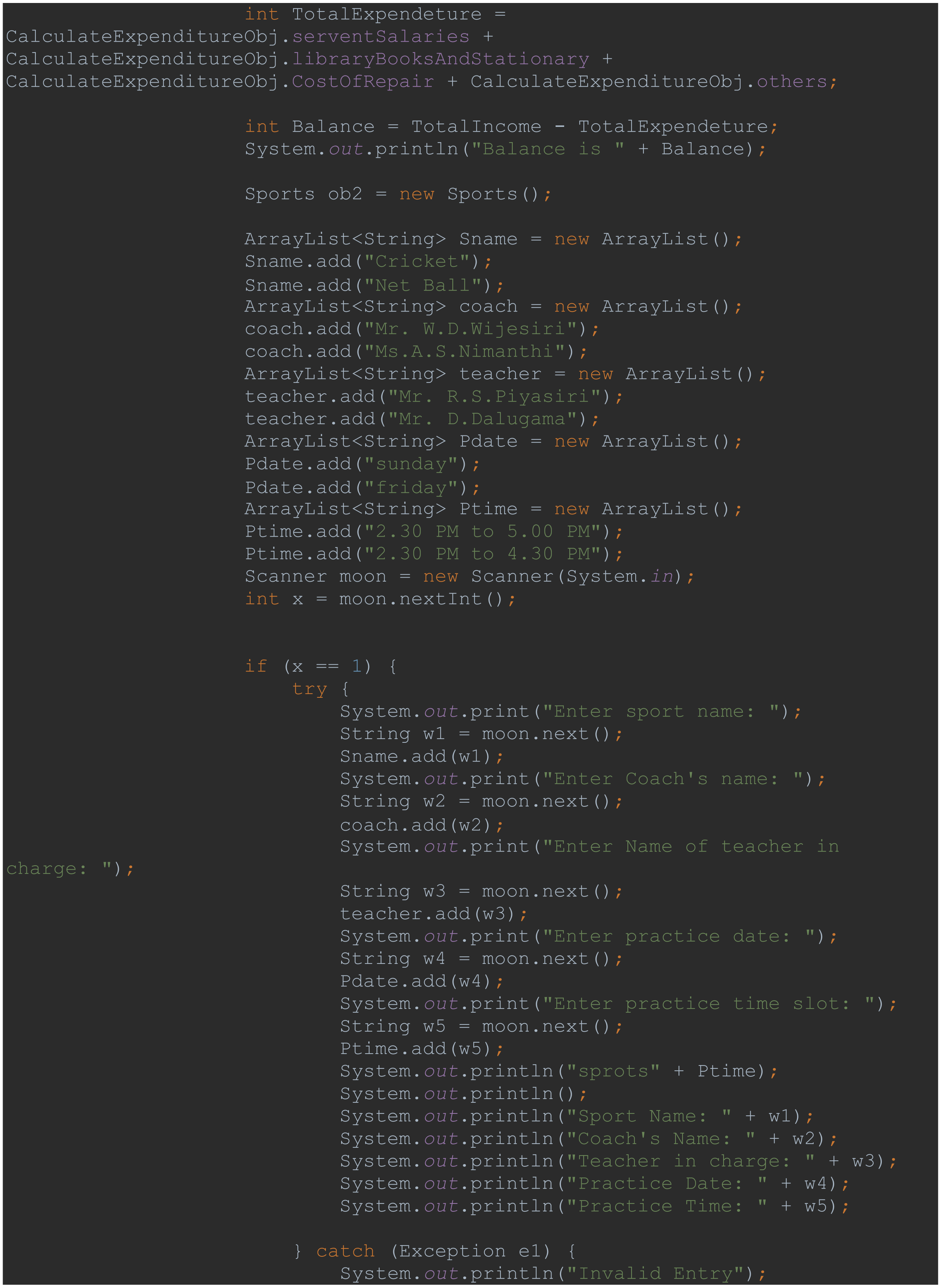
Main

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

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

        User user = new User();
        user.createProfile();



//          while () {
        if (user.login() == true) {

            user.viewProfile(user);

            System.out.println();
            System.out.println("Do you want to update your profile? (Y/N) "); String answer = sc.next();
            if (answer.equals("Y") || answer.equals("y")) { user.editProfile(user);
            } else {
                System.out.println("GOOD BYE :)...................................");
```

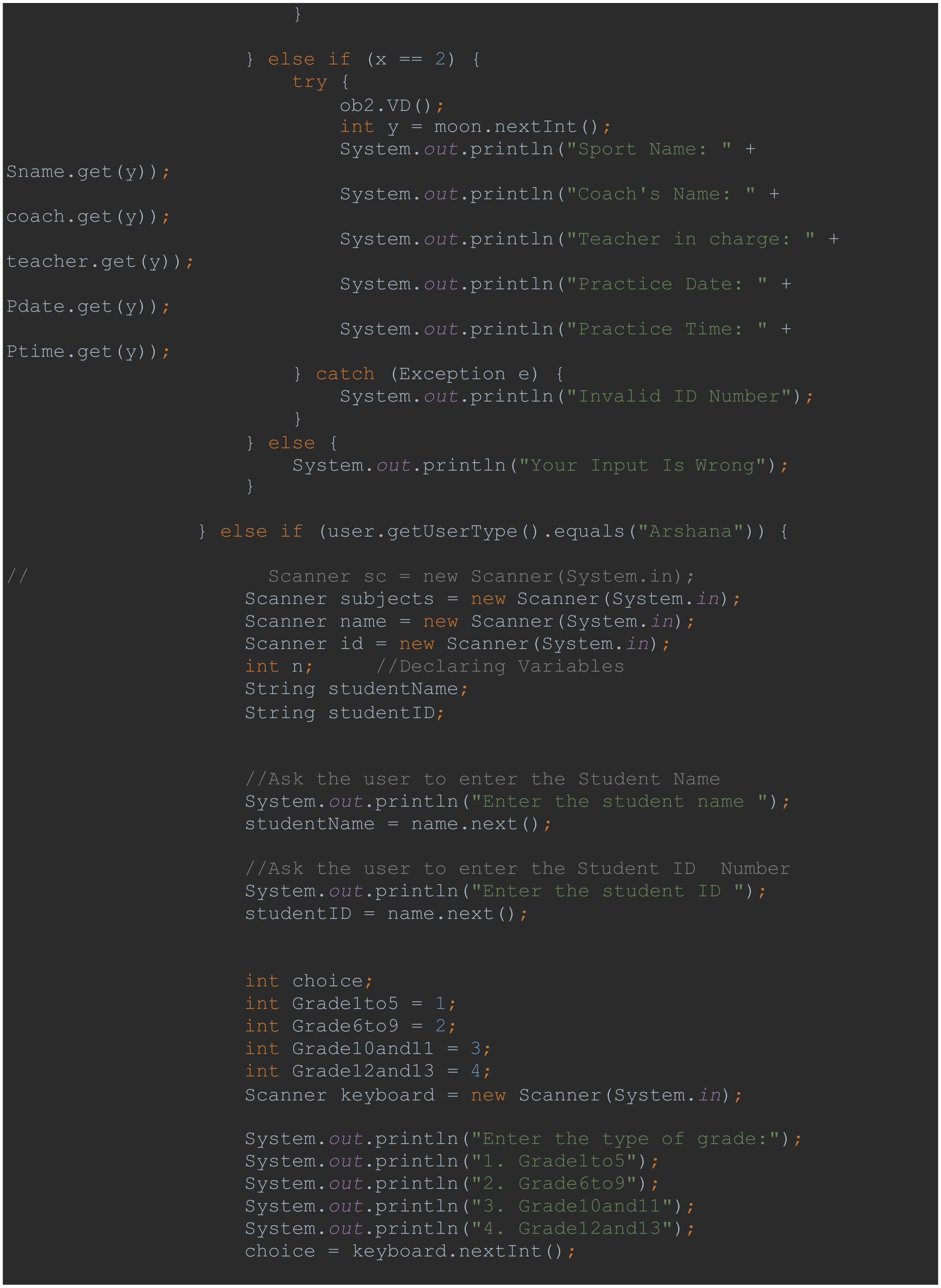


```
                if (user.getUserType().equals("Teacher")) {
//                  Scanner sc = new Scanner(System.in);

                    Teacher teacher1 = new Teacher();
                    Teacher teacher2 = new Teacher();
                    Teacher teacher3 = new Teacher();
                    teacher1.createProfile();
                    teacher2.createProfile();
                    teacher3.createProfile();


                    List<Teacher> teacherDetails = new ArrayList<Teacher>();
                    teacherDetails.add(teacher1);
                    teacherDetails.add(teacher2);
                    teacherDetails.add(teacher3);


                    System.out.println("Teacher details: ");
                    System.out.println();
                    for (Teacher teacher : teacherDetails) {
                        System.out.println("Name : " + teacher.getName());
                        System.out.println("Subject: " +
teacher.getSubject());
                        System.out.println("Teacher Id: " +
teacher.getTeacherID());
                        System.out.println("Contact Details");
                        System.out.println("Address: " +
teacher.getAddress());
                        System.out.println("Telephone Number: " +
teacher.getContactNumber());
                        System.out.println();
                    }

                } else if (user.getUserType().equals("Student")) {
//                  Scanner sc = new Scanner(System.in);

                    Student student1 = new Student();
                    Student student2 = new Student();
                    Student student3 = new Student();
                    student1.createProfile();
                    student2.createProfile();
                    student3.createProfile();


                    List<Student> Student = new ArrayList<Student>();
                    Student.add(student1);
                    Student.add(student2);
                    Student.add(student3);


                    System.out.println("Student details: ");
                    System.out.println();

                    for (Student student : Student) {
                        System.out.println("Name : " + student.getName());
                        System.out.println("Student Id: " +
student.getStudentID());
```

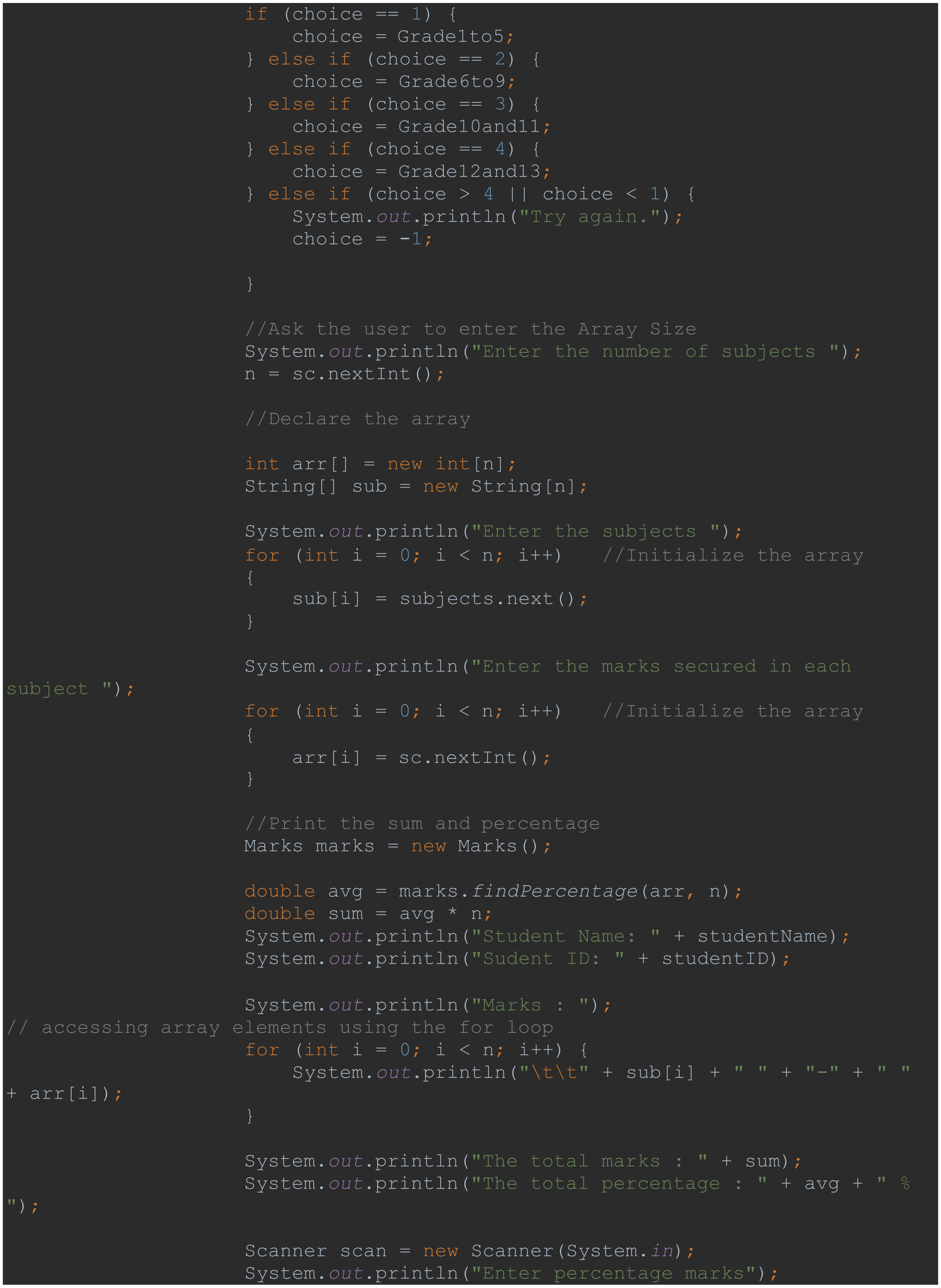
```
                        System.out.println("Contact Details");
                        System.out.println("Address: " +
student.getAddress());
                        System.out.println("Telephone Number: " +
student.getContactNumber());
                        System.out.println();

                    }

                } else if (user.getUserType().equals("Servant")) {

                    Servant ob1 = new Servant();

                    Scanner sun = new Scanner(System.in);
                    int x = sun.nextInt();
                    if (x == 1) {
                        ob1.Salary();
                    } else if (x == 2) {
                        ob1.ViewData();
                    } else {
                        System.out.println("Your Input is Wrong");
                    }
                } else if (user.getUserType().equals("Principle")) {
                    System.out.println("--------------Calculate Income-------
----");
                    System.out.println();
                    CalculateIncome CalculateIncomeObj = new
CalculateIncome();

                    CalculateExpenditure CalculateExpenditureObj = new
CalculateExpenditure();

                    CalculateIncomeObj.facilitiesServiceCharge();
                    CalculateIncomeObj.donations();
                    CalculateIncomeObj.incomeFromEvents();

                    CalculateIncomeObj.Tenders();
                    System.out.println(CalculateIncomeObj.toString());

                    System.out.println("-------------Calculate Expenditure---
----------");
                    System.out.println();

                    CalculateExpenditureObj.serventSalaries();
                    CalculateExpenditureObj.libraryBooksAndStationary();

                    CalculateExpenditureObj.repairs();
                    CalculateExpenditureObj.Others();
                    System.out.println(CalculateExpenditureObj.toString());

                    System.out.println("--------------Calculate Balance------
-------------");
                    System.out.println();
                    int TotalIncome =
CalculateIncomeObj.facilitiesAndServiceCharge + CalculateIncomeObj.donations
+ CalculateIncomeObj.incomeFromEvents;
```



```
                    int TotalExpendeture =
CalculateExpenditureObj.serventSalaries +
CalculateExpenditureObj.libraryBooksAndStationary +
CalculateExpenditureObj.CostOfRepair + CalculateExpenditureObj.others;

                    int Balance = TotalIncome - TotalExpendeture;
                    System.out.println("Balance is " + Balance);

                    Sports ob2 = new Sports();

                    ArrayList<String> Sname = new ArrayList();
                    Sname.add("Cricket");
                    Sname.add("Net Ball");
                    ArrayList<String> coach = new ArrayList();
                    coach.add("Mr. W.D.Wijesiri");
                    coach.add("Ms.A.S.Nimanthi");
                    ArrayList<String> teacher = new ArrayList();
                    teacher.add("Mr. R.S.Piyasiri");
                    teacher.add("Mr. D.Dalugama");
                    ArrayList<String> Pdate = new ArrayList();
                    Pdate.add("sunday");
                    Pdate.add("friday");
                    ArrayList<String> Ptime = new ArrayList();
                    Ptime.add("2.30 PM to 5.00 PM");
                    Ptime.add("2.30 PM to 4.30 PM");
                    Scanner moon = new Scanner(System.in);
                    int x = moon.nextInt();


                    if (x == 1) {
                        try {
                            System.out.print("Enter sport name: ");
                            String w1 = moon.next();
                            Sname.add(w1);
                            System.out.print("Enter Coach's name: ");
                            String w2 = moon.next();
                            coach.add(w2);
                            System.out.print("Enter Name of teacher in
charge: ");

                            String w3 = moon.next();
                            teacher.add(w3);
                            System.out.print("Enter practice date: ");
                            String w4 = moon.next();
                            Pdate.add(w4);
                            System.out.print("Enter practice time slot: ");
                            String w5 = moon.next();
                            Ptime.add(w5);
                            System.out.println("sprots" + Ptime);
                            System.out.println();
                            System.out.println("Sport Name: " + w1);
                            System.out.println("Coach's Name: " + w2);
                            System.out.println("Teacher in charge: " + w3);
                            System.out.println("Practice Date: " + w4);
                            System.out.println("Practice Time: " + w5);

                        } catch (Exception e1) {
                            System.out.println("Invalid Entry");
```



```
                        }

                    } else if (x == 2) {
                        try {
                            ob2.VD();
                            int y = moon.nextInt();
                            System.out.println("Sport Name: " +
Sname.get(y));
                            System.out.println("Coach's Name: " +
coach.get(y));
                            System.out.println("Teacher in charge: " +
teacher.get(y));
                            System.out.println("Practice Date: " +
Pdate.get(y));
                            System.out.println("Practice Time: " +
Ptime.get(y));
                        } catch (Exception e) {
                            System.out.println("Invalid ID Number");
                        }
                    } else {
                        System.out.println("Your Input Is Wrong");
                    }

                } else if (user.getUserType().equals("Arshana")) {

//                      Scanner sc = new Scanner(System.in);
                    Scanner subjects = new Scanner(System.in);
                    Scanner name = new Scanner(System.in);
                    Scanner id = new Scanner(System.in);
                    int n;     //Declaring Variables
                    String studentName;
                    String studentID;


                    //Ask the user to enter the Student Name
                    System.out.println("Enter the student name ");
                    studentName = name.next();

                    //Ask the user to enter the Student ID  Number
                    System.out.println("Enter the student ID ");
                    studentID = name.next();


                    int choice;
                    int Grade1to5 = 1;
                    int Grade6to9 = 2;
                    int Grade10and11 = 3;
                    int Grade12and13 = 4;
                    Scanner keyboard = new Scanner(System.in);

                    System.out.println("Enter the type of grade:");
                    System.out.println("1. Grade1to5");
                    System.out.println("2. Grade6to9");
                    System.out.println("3. Grade10and11");
                    System.out.println("4. Grade12and13");
                    choice = keyboard.nextInt();
```



```
                    if (choice == 1) {
                        choice = Grade1to5;
                    } else if (choice == 2) {
                        choice = Grade6to9;
                    } else if (choice == 3) {
                        choice = Grade10and11;
                    } else if (choice == 4) {
                        choice = Grade12and13;
                    } else if (choice > 4 || choice < 1) {
                        System.out.println("Try again.");
                        choice = -1;

                    }

                    //Ask the user to enter the Array Size
                    System.out.println("Enter the number of subjects ");
                    n = sc.nextInt();

                    //Declare the array

                    int arr[] = new int[n];
                    String[] sub = new String[n];

                    System.out.println("Enter the subjects ");
                    for (int i = 0; i < n; i++)    //Initialize the array
                    {
                        sub[i] = subjects.next();
                    }

                    System.out.println("Enter the marks secured in each
subject ");
                    for (int i = 0; i < n; i++)    //Initialize the array
                    {
                        arr[i] = sc.nextInt();
                    }

                    //Print the sum and percentage
                    Marks marks = new Marks();

                    double avg = marks.findPercentage(arr, n);
                    double sum = avg * n;
                    System.out.println("Student Name: " + studentName);
                    System.out.println("Sudent ID: " + studentID);

                    System.out.println("Marks : ");
// accessing array elements using the for loop
                    for (int i = 0; i < n; i++) {
                        System.out.println("\t\t" + sub[i] + " " + "-" + " "
+ arr[i]);
                    }

                    System.out.println("The total marks : " + sum);
                    System.out.println("The total percentage : " + avg + " %
");


                    Scanner scan = new Scanner(System.in);
                    System.out.println("Enter percentage marks");
                    double percentage = scan.nextDouble();

                    if (percentage >= 90) { System.out.println("Excellent: Grade A");
                    } else if (percentage < 90 && percentage >= 80) {
                        System.out.println("Very Good: Grade B");
                    } else if (percentage < 80 && percentage >= 70) {
                        System.out.println("Good: Grade C");
                    } else if (percentage < 70 && percentage >= 60) {
                        System.out.println("Satisfactory: Grade D");
                    } else if (percentage < 60 && percentage >= 50) {
                        System.out.println("Work Hard: Grade E");
                    } else if (percentage < 50 && percentage >= 40) {
                        System.out.println("Just Passed: Grade F");
                    } else {
                        System.out.println("Failed!");

                    }
                } else {
                    System.out.println("GOOD BYE :). ................................ ");
                }

            }
        }
    }
}
```

## Challenges & Solutions

There were various errors came up while developing the program. To correct them, we refer some additional books as well as additional programs and lecture notes.

We were correct some errors by discussing with all the team members and changing some points in our system.

## Teamwork

- PS/2019/279 - Arshana Sagadevan – Developing Marks class
- PS/2019/051 - Pitipanage Malaka Dilshan - Developing Treasury class
- PS/2019/181 - Sashini Udyani Jayawardhana - Developing User class
- PS/2019/233 - Daluwakgodage Dulakshi Anuradha - Developing Servant class and Sports class
- PS/2019/195 - Wathsala wimansani Ranasingha - Developing Student class and Teacher class