1.Bank account

import java.util.Scanner;

class BankAccount {

private double balance;

public BankAccount(double initialBalance) {

balance = initialBalance;

}

public double getBalance() {

return balance;

}

public void deposit(double amount) {

if (amount > 0) {

balance += amount;

System.out.println("Deposited: $" + amount);

} else {

System.out.println("Invalid amount for deposit.");

}

}

public void withdraw(double amount) {

if (amount > 0 && amount <= balance) {

balance -= amount;

System.out.println("Withdrawn: $" + amount);

} else {

System.out.println("Insufficient funds or invalid amount for withdrawal.");

}

}

}

class ATM {

private BankAccount userAccount;

public ATM(BankAccount account) {

userAccount = account;

}

public void checkBalance() {

System.out.println("Your account balance: $" + userAccount.getBalance());

}

public static void main(String[] args) {

BankAccount userAccount = new BankAccount(1000.0); // Initial balance of $1000

ATM atm = new ATM(userAccount);

Scanner scanner = new Scanner(System.in);

while (true) {

System.out.println("\nATM Menu:");

System.out.println("1. Check Balance");

System.out.println("2. Deposit Money");

System.out.println("3. Withdraw Money");

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

System.out.print("Select an option: ");

int choice = scanner.nextInt();

switch (choice) {

case 1:

atm.checkBalance();

break;

case 2:

System.out.print("Enter the amount to deposit: $");

double depositAmount = scanner.nextDouble();

atm.userAccount.deposit(depositAmount);

break;

case 3:

System.out.print("Enter the amount to withdraw: $");

double withdrawAmount = scanner.nextDouble();

atm.userAccount.withdraw(withdrawAmount);

break;

case 4:

System.out.println("Exiting. Thank you!");

scanner.close();

System.exit(0);

default:

System.out.println("Invalid choice. Please select a valid option.");

}

}

}

}

2.student management

import java.util.ArrayList;

import java.util.Scanner;

class Student {

private String name;

private int rollNumber;

private String grade;

public Student(String name, int rollNumber, String grade) {

this.name = name;

this.rollNumber = rollNumber;

this.grade = grade;

}

public String getName() {

return name;

}

public int getRollNumber() {

return rollNumber;

}

public String getGrade() {

return grade;

}

@Override

public String toString() {

return "Name: " + name + ", Roll Number: " + rollNumber + ", Grade: " + grade;

}

}

class StudentManagementSystem {

private ArrayList<Student> students;

public StudentManagementSystem() {

students = new ArrayList<>();

}

public void addStudent(Student student) {

students.add(student);

}

public void removeStudent(int rollNumber) {

students.removeIf(student -> student.getRollNumber() == rollNumber);

}

public Student findStudent(int rollNumber) {

for (Student student : students) {

if (student.getRollNumber() == rollNumber) {

return student;

}

}

return null;

}

public void displayAllStudents() {

for (Student student : students) {

System.out.println(student);

}

}

public static void main(String[] args) {

StudentManagementSystem sms = new StudentManagementSystem();

Scanner scanner = new Scanner(System.in);

while (true) {

System.out.println("\nStudent Management System Menu:");

System.out.println("1. Add Student");

System.out.println("2. Remove Student");

System.out.println("3. Search for Student");

System.out.println("4. Display All Students");

System.out.println("5. Exit");

System.out.print("Select an option: ");

int choice = scanner.nextInt();

scanner.nextLine(); // Consume the newline character

switch (choice) {

case 1:

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

String name = scanner.nextLine();

System.out.print("Enter Roll Number: ");

int rollNumber = scanner.nextInt();

scanner.nextLine(); // Consume the newline character

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

String grade = scanner.nextLine();

sms.addStudent(new Student(name, rollNumber, grade));

break;

case 2:

System.out.print("Enter Roll Number to Remove: ");

int rollToRemove = scanner.nextInt();

sms.removeStudent(rollToRemove);

break;

case 3:

System.out.print("Enter Roll Number to Search: ");

int rollToSearch = scanner.nextInt();

Student foundStudent = sms.findStudent(rollToSearch);

if (foundStudent != null) {

System.out.println("Student Found: " + foundStudent);

} else {

System.out.println("Student not found.");

}

break;

case 4:

sms.displayAllStudents();

break;

case 5:

System.out.println("Exiting. Thank you!");

scanner.close();

System.exit(0);

default:

System.out.println("Invalid choice. Please select a valid option.");

}

}

}

}

3. student mark

import java.util.Scanner;

public class GradeCalculator {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

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

int numberOfSubjects = scanner.nextInt();

double totalMarks = 0;

for (int i = 1; i <= numberOfSubjects; i++) {

System.out.print("Enter marks obtained in subject " + i + ": ");

double marks = scanner.nextDouble();

totalMarks += marks;

}

double averagePercentage = (totalMarks / (numberOfSubjects \* 100)) \* 100;

System.out.println("Total Marks: " + totalMarks);

System.out.println("Average Percentage: " + averagePercentage + "%");

String grade = calculateGrade(averagePercentage);

System.out.println("Grade: " + grade);

scanner.close();

}

public static String calculateGrade(double averagePercentage) {

if (averagePercentage >= 90) {

return "A+";

} else if (averagePercentage >= 80) {

return "A";

} else if (averagePercentage >= 70) {

return "B";

} else if (averagePercentage >= 60) {

return "C";

} else if (averagePercentage >= 50) {

return "D";

} else {

return "F";

}

}

}