**SYEDA ALINA KHURRAM**

**MIDTERM**

**Question # 01:**

**Create a class called "BankAccount" with the following attributes and methods:**

**● account\_number (integer)**

**● balance (double)**

**● account\_type (string)**

**● interest\_rate (double)**

**● deposit(amount): adds the amount to the balance.**

**● withdraw(amount): subtracts the amount from the balance. You cannot withdraw**

**more than the current balance.**

**● add\_interest(): adds interest to the balance based on the interest rate.**

**● display(): prints out the account number, balance, account type, and interest rate.**

**Then, create two instances of the BankAccount class, each with its own account number,**

**balance, account type, and interest rate.**

**Finally, call the deposit(), withdraw(), add\_interest(), and display() methods on each instance**

**and confirm that the information is updated and displayed correctly.**

CODE

class BankAccount {

int accountNumber;

double balance;

String accountType;

double interestRate;

BankAccount(this.accountNumber, this.balance, this.accountType, this.interestRate);

void deposit(double amount) {

balance += amount;

}

void withdraw(double amount) {

if (amount <= balance) {

balance -= amount;

} else {

print('Insufficient funds');

}

}

void addInterest() {

double interestAmount = balance \* interestRate / 100;

balance += interestAmount;

}

void display() {

print('Account Number: $accountNumber');

print('Balance: \$${balance.toStringAsFixed(2)}');

print('Account Type: $accountType');

print('Interest Rate: $interestRate%');

}

}

void main() {

// Creating two instances of BankAccount

BankAccount account1 = BankAccount(123456, 1000.0, 'Savings', 5.0);

BankAccount account2 = BankAccount(987654, 5000.0, 'Checking', 3.0);

// Calling methods on the first account

account1.deposit(500.0);

account1.withdraw(200.0);

account1.addInterest();

account1.display();

print(''); // Adding a line break for readability

// Calling methods on the second account

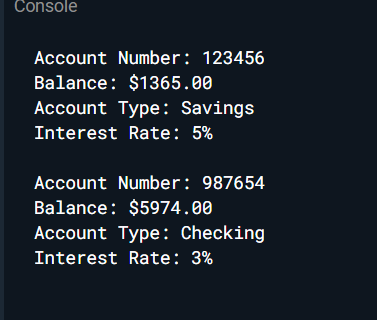
account2.deposit(1000.0);

account2.withdraw(200.0);

account2.addInterest();

account2.display();

}



**Question # 02:**

**Create a class called "Student" with the following attributes and methods:**

**● name (string)**

**● id (string)**

**● courses (list of strings)**

**● add\_course(course): adds a course to the student's list of courses.**

**● drop\_course(course): removes a course from the student's list of courses.**

**● display\_courses(): prints out the student's list of courses.**

**Then, create two instances of the Student class, each with their name, id, and courses.**

**Finally, call the add\_course(), drop\_course(), and display\_courses() methods on each**

**instance and confirm that the information is updated and displayed correctly.**

CODE

class Student {

String name;

String id;

List<String> courses;

Student(this.name, this.id, this.courses);

void addCourse(String course) {

courses.add(course);

}

void dropCourse(String course) {

courses.remove(course);

}

void displayCourses() {

print('Courses for $name (ID: $id):');

for (String course in courses) {

print(course);

}

}

}

void main() {

// Creating two instances of Student

Student student1 = Student('John Doe', '123456', ['Math', 'English', 'History']);

Student student2 = Student('Jane Smith', '987654', ['Science', 'Art', 'PE']);

// Calling methods on the first student

student1.addCourse('Biology');

student1.dropCourse('English');

student1.displayCourses();

print(''); // Adding a line break for readability

// Calling methods on the second student

student2.addCourse('Music');

student2.dropCourse('PE');

student2.displayCourses();

}

