Here's the first question and three more with different scenarios. All questions will require students to write C++ code with member functions, conditions, and attributes:

Question 1: Employee Class

Scenario:
You are required to create a C++ program that models an **Employee** class. The **Employee** class should have the following attributes:

- string name: The name of the employee.
- int age: The age of the employee.
- double salary: The salary of the employee.

Additionally, the class should have the following member functions:

1. **setDetails()**: This function takes the name, age, and salary of the employee and sets the respective attributes.
2. **displayDetails()**: This function displays the name, age, and salary of the employee.
3. **checkIncrementEligibility()**: This function checks if the employee's age is greater than 25. If it is, the employee is eligible for a salary increment of 10%. The function should display whether the employee is eligible or not and adjust the salary accordingly.

Write the C++ code to implement this scenario.

```
#include <iostream>
using namespace std;

class Employee {
public:
    string name;
    int age;
    double salary;

    // Constructor to initialize the employee details
    Employee(string n, int a, double s) {
        name = n;
        age = a;
```

```
        salary = s;
    }

    // Function to display employee details
    void displayDetails() {
        cout << "Employee Name: " << name << endl;
        cout << "Age: " << age << endl;
        cout << "Salary: " << salary << endl;
    }

    // Function to check salary increment eligibility
    void checkIncrementEligibility() {
        if (age > 25) {
            salary += salary * 0.10;  // 10% increment
            cout << name << " is eligible for a salary increment!" << endl;
            cout << "New Salary: " << salary << endl;
        } else {
            cout << name << " is not eligible for a salary increment." << endl;
        }
    }
};

int main() {
    // Create an Employee object
    Employee emp("John Doe", 30, 50000);

    // Display employee details and check increment eligibility
    emp.displayDetails();
    emp.checkIncrementEligibility();

    return 0;
}
```

Question 2: Student Class

Scenario:
You are required to create a C++ program that models a **Student** class. The **Student** class should have the following attributes:

- string studentName: The name of the student.
- int studentAge: The age of the student.

- double marks: The marks scored by the student.

The class should have the following member functions:

1. **setStudentDetails()**: This function takes the name, age, and marks of the student and sets the respective attributes.
2. **displayStudentDetails()**: This function displays the name, age, and marks of the student.
3. **checkScholarshipEligibility()**: This function checks if the student's marks are greater than or equal to 85%. If the student qualifies, they are eligible for a scholarship. The function should display whether the student is eligible or not.

Write the C++ code to implement this scenario.

```
#include <iostream>
using namespace std;

class Student {
public:
    string studentName;
    int studentAge;
    double marks;

    // Constructor to initialize student details
    Student(string name, int age, double m) {
        studentName = name;
        studentAge = age;
        marks = m;
    }

    // Function to display student details
    void displayStudentDetails() {
        cout << "Student Name: " << studentName << endl;
        cout << "Age: " << studentAge << endl;
        cout << "Marks: " << marks << "%" << endl;
    }

    // Function to check scholarship eligibility
    void checkScholarshipEligibility() {
        if (marks >= 85) {
            cout << studentName << " is eligible for a scholarship!" << endl;
        } else {
```

```
            cout << studentName << " is not eligible for a scholarship." << endl;
        }
    }
};

int main() {
    // Create a Student object
    Student student1("Alice Johnson", 20, 90);

    // Display student details and check scholarship eligibility
    student1.displayStudentDetails();
    student1.checkScholarshipEligibility();

    return 0;
}
```

Question 3: BankAccount Class

Scenario:
You are required to create a C++ program that models a **BankAccount** class. The **BankAccount** class should have the following attributes:

- string accountHolderName: The name of the account holder.
- int accountNumber: The account number of the account holder.
- double balance: The balance in the account.

The class should have the following member functions:

1. **setAccountDetails()**: This function takes the account holder's name, account number, and balance and sets the respective attributes.
2. **displayAccountDetails()**: This function displays the account holder’s name, account number, and balance.
3. **depositAmount()**: This function allows the account holder to deposit a specified amount into their account.
4. **checkBalance()**: This function checks if the balance is less than $1000. If it is, it displays a warning message saying the account balance is low.

Write the C++ code to implement this scenario.

```
#include <iostream>
using namespace std;

class BankAccount {
public:
    string accountHolderName;
    int accountNumber;
    double balance;

    // Constructor to initialize account details
    BankAccount(string name, int accNum, double bal) {
        accountHolderName = name;
        accountNumber = accNum;
        balance = bal;
    }

    // Function to display account details
    void displayAccountDetails() {
        cout << "Account Holder: " << accountHolderName << endl;
        cout << "Account Number: " << accountNumber << endl;
        cout << "Balance: " << balance << endl;
    }

    // Function to deposit amount
    void depositAmount(double amount) {
        balance += amount;
        cout << "Deposited: " << amount << endl;
        cout << "New Balance: " << balance << endl;
    }

    // Function to check if balance is low
    void checkBalance() {
        if (balance < 1000) {
            cout << "Warning: Low balance!" << endl;
        }
    }
};

int main() {
    // Create a BankAccount object
    BankAccount account1("John Smith", 12345, 800);

    // Display account details, deposit amount, and check balance
    account1.displayAccountDetails();
```

```
    account1.depositAmount(500);
    account1.checkBalance();

    return 0;
}
```

Question 4: Vehicle Class

Scenario:
You are required to create a C++ program that models a **Vehicle** class. The **Vehicle** class should have the following attributes:

- string vehicleType: The type of the vehicle (e.g., Car, Truck, Bike).
- double fuelCapacity: The fuel capacity of the vehicle in liters.
- double currentFuel: The current amount of fuel in the vehicle.

The class should have the following member functions:

1. **setVehicleDetails()**: This function takes the vehicle type, fuel capacity, and current fuel amount and sets the respective attributes.
2. **displayVehicleDetails()**: This function displays the vehicle type, fuel capacity, and current fuel.
3. **checkFuelEfficiency()**: This function checks if the current fuel is less than 20% of the fuel capacity. If so, it displays a warning message indicating that the vehicle is running low on fuel.
4. **refuelVehicle()**: This function allows the user to refuel the vehicle. The function should accept an amount of fuel to add to the vehicle and update the current fuel level.

Write the C++ code to implement this scenario.

```
#include <iostream>
using namespace std;

class Vehicle {
public:
    string vehicleType;
    double fuelCapacity;
    double currentFuel;
```

```
    // Constructor to initialize vehicle details
    Vehicle(string type, double capacity, double fuel) {
        vehicleType = type;
        fuelCapacity = capacity;
        currentFuel = fuel;
    }

    // Function to display vehicle details
    void displayVehicleDetails() {
        cout << "Vehicle Type: " << vehicleType << endl;
        cout << "Fuel Capacity: " << fuelCapacity << " liters" << endl;
        cout << "Current Fuel: " << currentFuel << " liters" << endl;
    }

    // Function to check if fuel is low
    void checkFuelEfficiency() {
        if (currentFuel < 0.2 * fuelCapacity) {
            cout << "Warning: Fuel level is below 20%! Please refuel." << endl;
        }
    }

    // Function to refuel the vehicle
    void refuelVehicle(double fuelAmount) {
        if (currentFuel + fuelAmount <= fuelCapacity) {
            currentFuel += fuelAmount;
            cout << "Refueled " << fuelAmount << " liters." << endl;
            cout << "New Fuel Level: " << currentFuel << " liters." << endl;
        } else {
            cout << "Error: Cannot refuel beyond the fuel capacity." << endl;
        }
    }
};

int main() {
    // Create a Vehicle object
    Vehicle car("Car", 50, 8);

    // Display vehicle details, check fuel efficiency, and refuel
    car.displayVehicleDetails();
    car.checkFuelEfficiency();
    car.refuelVehicle(10);

    return 0;
```

```
}
```

These questions allow students to practice creating classes with attributes and member functions that contain conditional logic and calculations.