



Object Oriented Programming

Lab Manual 4



Introduction

After a week of rigorous coding, Welcome back!

You have learned all about the **Classes, Constructors, and member functions** in the previous lab manuals. Let's move on to the next, **new, and interesting concepts.**

Students, In Object-Oriented Programming, the Class is a combination of data members and member functions. In this Lab, we will learn about including **multiple classes** into our program to achieve the object's oriented philosophy.

Let's do some coding.

Students, Recall this question from the class that you attempted and developed CRC cards.

Problem Statement 01:

Identify Classes, Data and Behaviour

Now, we need to identify the classes, data and behaviour from real time scenarios.
Suppose you have to implement a software for a **College or University Student Record Management System**. University should maintain the information about student's full name, roll number, cgpa, matric marks, fsc marks, ecat marks, current semester, fees, home town, whether day scholar or hostelite and whether availing a scholarship or not. Merit of student can be calculated by adding 60% of Fsc Marks and 40% of ECat marks. Students can check their scholarship status: A student is eligible for scholarship if her merit is greater than 80% and she is hostelite.

Solution



Object Oriented Programming

Lab Manual 4



– CRC Card

student

Name: string
rollNumber: int
cGPA: float
matricMarks: int
fscMarks: int
ecatMarks: int
homeTown: string
isHostelite: bool
isTakingScholarship: bool

student(parametrized)
calculateMerit(): float
isEligibleforScholarship(float meritPercentage): bool

Self Assessment Task: Implement the Class Student by using the CRC card that you developed in the class and use the driver program to test your member functions.

Congratulations !!!!!!!!!!!!!!! you have implemented your first class by using the CRC card.



Object Oriented Programming

Lab Manual 4



Problem Statement 02:

Identify Classes, Data and Behaviour

Suppose you have to implement a software for a Library Management System. The **library system** must keep track of the books whether the book is available or borrowed. Books contains the title, list of chapters, number of pages, price, name of the author.

A person can see is the book **available or borrowed**. He can also see the **bookmark** is on which page number. Also he can see the **name of a specific chapter**.

What will be class for Book, its Data Members, and Behaviours.

Solution

CRC Card

Book

author: string
pages: int
Chapters: List<string>
bookMark: int
price: int

Book(parametrized)
getChapter(chapterNumber: int): string
getBookMark(): int
setBookMark(pageNumber: int): void
getBookPrice(): int
setBookPrice(newPrice: int): void

Self Assessment Task: Implement the Class Student by using the CRC card that you developed in the class and use the driver program to test your member functions.

Great Work Students !!! You guys are doing an excellent job. Well done. Take a two-minute break. Well deserved.



Object Oriented Programming

Lab Manual 4



Problem Statement 03:

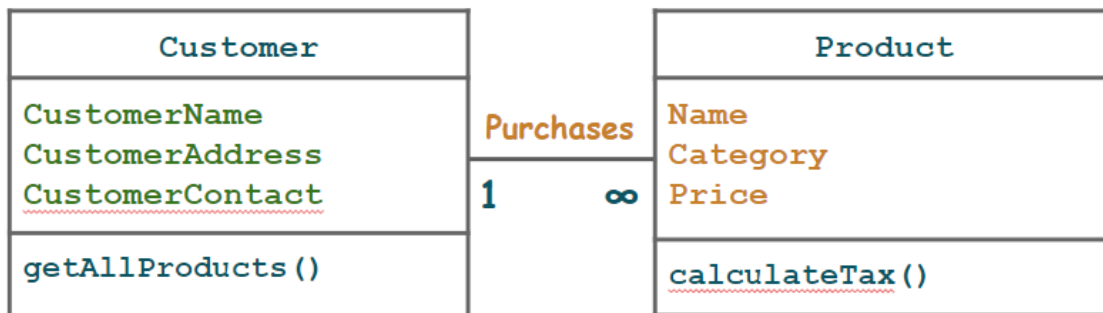
Problem Scenario

Let's suppose a store needs to **save the information** of the **product** and the **customer** who has bought the product.

Store wants to calculate the **total purchases** of a customer. Store wants to calculate the **tax on the purchased products** as well.

Solution

CRC Card



We need to define **two separate classes** in order to achieve the object-oriented philosophy. So same kinds of data shall reside in the same place.



Object Oriented Programming

Lab Manual 4



```
class Customer
{
    public string CustomerName;
    public string CustomerAddress;
    public string CustomerContact;
    public List<Product> products = new List<Product>();
    public List<Product> getAllProducts()
    {
        return products;
    }
    public addProduct(Product p)
    {
        products.Add(p);
    }
}
```

This code creates the **Customer** class **Data Members**, **Constructor**, and **Member Function** for adding products to the list.

```
class Product
{
    public string name;
    public string category;
    public int price;
    public float calculateTax()
    {
        // Implementation
    }
}
```

This code creates the **Product** class **Data Members** and **Member Function** for calculating tax.

Self Assessment Task 01: Implement this program by using the CRC card that we have developed and then use the driver program to test your member functions.

Self Assessment Task 02: Implement the member function in the appropriate class for calculating the tax of all the products that a customer has purchased.



Object Oriented Programming

Lab Manual 4



Challenge # 01:

Read the following question carefully.

Self Assessment

1. Identify the **classes** within the following case study.

Academic branch offers **different programs** within different departments each program has a **degree title** and **duration of degree**.

Student Apply for admission in University and provides his/her **name, age, FSC, and Ecat Marks** and selects **any number of preferences** among the available programs.

Admission department prepares a merit list according to the **highest merit** and **available seats** and registers selected **students** in the program.

Academic Branch also **add subjects** for each program. A subject have **subject code, credit hours, subjectType**. A Program cannot have more than **20 Credit hour** subjects. A Student Registers multiple subjects but he/she can not take more than **9 credit hours**.

Fee department **generate fees** according to registered subjects of the students.

Try out yourself.



Object Oriented Programming

Lab Manual 4



Identification of Classes

By looking at the above-mentioned self-assessment you can extract the following possible class-like structures from the given statement.

- Student
- Subject
- Degree Program

Note: Create a separate class in the same BL(Business Logic) folder of your program.

Now Try to Build the Class Diagram/Domain Model of these classes.

After creating your Class diagram you have to write the Driver Program by using your Classes.

Wireframes of Driver Program:

1. Main Menu

```
*****
                        UAMS
*****
1. Add Student
2. Add Degree Program
3. Generate Merit
4. View Registered Students
5. View Students of a Specific Program
6. Register Subjects for a Specific Student
7. Calculate Fees for all Registered Students
8. Exit
Enter Option:
```

2. Option 2: Degree Program



Object Oriented Programming

Lab Manual 4



```
Enter Degree Name: CE
Enter Degree Duration: 4
Enter Seats for Degree: 1
Enter How many Subjects to Enter: 1
Enter Subject Code: 162
Enter Subject Type: OOP
Enter Subject Credit Hours: 3
Enter Subject Fees: 8000
Press any key to Continue..
```

3. Option 1: Add Student

```
Enter Student Name: AAA
Enter Student Age: 12
Enter Student FSc Marks: 1000
Enter Student Ecat Marks: 390
Available Degree Programs
CS
Enter how many preferences to Enter: 1
CS
Press any key to Continue..
```

4. Option 3: Generate Merit

```
AAA got Admission in CS
BBB did not get Admission
CCC got Admission in CE
DDD did not get Admission
Press any key to Continue..
```

5. Option 4: Registered Students

| Name | FSC | Ecat | Age |
|------|------|------|-----|
| AAA | 1000 | 390 | 12 |
| CCC | 999 | 380 | 15 |

```
Press any key to Continue..
```

6. Option 5: Specific Degree



Object Oriented Programming

Lab Manual 4



```
Enter Degree Name: CS
Name      FSC      Ecat      Age
AAA       1000     390       12
Press any key to Continue..
```

7. Option 6: Register Subjects

Ask the Student name and then ask for the subject code. If the conditions are satisfied then the student's subject should be registered.

8. Option 7: Generate Fee

Fees should be generated for all the registered students



Object Oriented Programming

Lab Manual 4



You have made it through all that. Excellent work students !!!
You guys are successfully en route to be Kamyab Programmers.

“No more Work for Today.”

Good Luck and Best Wishes !!
Happy Coding ahead :)