Usman Institute of Technology

Affiliated with NED University of Engineering and Technology

**Spring 2020**

CS121 Object Oriented Programming

**Midterm Exam -Assignment 1 (Max. Marks: 10)**

**Name: SAIFULLAH Course Code: CS-121 Department: Computer Science**

**Roll #: 18B-092-CS Course Title: Object Oriented Prog Batch: 2018**

**Section: SE-B Instructor: Dr. Lubaid Ahmed Year: Spring-2020**

**Submission Date: Saturday, July 25, 2020**

**Question1**: You have been hired as a developer in ABC software house. They are currently working on the following project.

***HYPERMART CASHFLOW***

1. Analyze the system and break them according to both techniques (top down and bottom up).

**HYPERMART CASHFLOW:**

The system is to take records of cash flow within the store and tells whether the store is in profit or in loss. So, for this record we must take record of every transaction in store like sales purchases expenses so at the end of the month or any time period we could know that store is in profit or not. So, for achieving this goal we need to break it into following parts (classes).

We have 6 classes:

1. Transaction
2. Sale
3. Purchase
4. Expense
5. Stock
6. Cashflow
7. Transaction:

Method:

1. Assign id()
2. Add()
3. View()
4. Delete()
5. Total()
6. Sale:

Methods:

1. Assign id()
2. Add()
3. View()
4. Delete()
5. Total()
6. Purchase

Methods:

1. Assign id()
2. Add()
3. View()
4. Delete()
5. Total()
6. Expense

Methods:

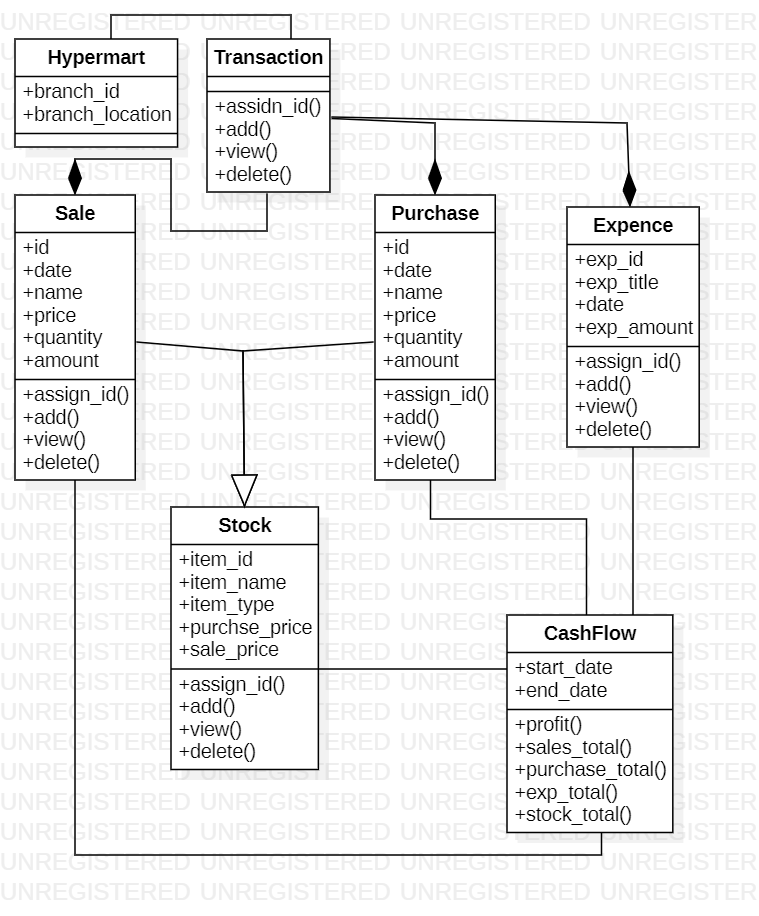
1. Assign id()
2. Add()
3. View()
4. Delete()
5. Total()
6. Stock

Methods

1. Assign id()
2. Add()
3. View()
4. Delete()
5. Total()
6. Cashflow

Methods:

1. Profit()
2. Sale total()
3. Expense total()
4. Purchase total()
5. Stock total()
6. Make UML use case, class diagrams, sequence diagram and activity diagram of your project.

Class diagram:

Use case diagram: A picture containing computer

Description automatically generated

Sequence diagram: A close up of a window

Description automatically generated

Activity diagram:A screenshot of a computer

Description automatically generated

1. Analyze the behaviour of each class and use (Abstract, Polymorphism, Inheritance) where necessary.

**Abstract class:**

Transaction class is used as abstract class.

**Inheritance:**

**Single Inheritance:**

The classes Sale, Purchase and Expense are inherited from abstract class transaction.

**Multiple inheritance:**

The stock class inherits from sale and purchase class.

**Multi-level inheritance:**

Stock class is inherited from sale and purchase class which are also inherited from class transaction abstract class.

**Polymorphism:**

**Method overriding:**

In sub classes of Transaction method overriding is used.

1. Create your GUI template (You are free to use PYQT5 or tkinter).

GUI is created using tkinter.

1. Code according to your Use case and class diagrams.