# Department of Computing

# CS 212: Object Oriented Programming

# Class: BESE-11 AB

# Lab 11: Inheritance and Composition in Java

# Date: May 19, 2021

# Instructor: Ms. Hania Aslam

**Learning Objectives**

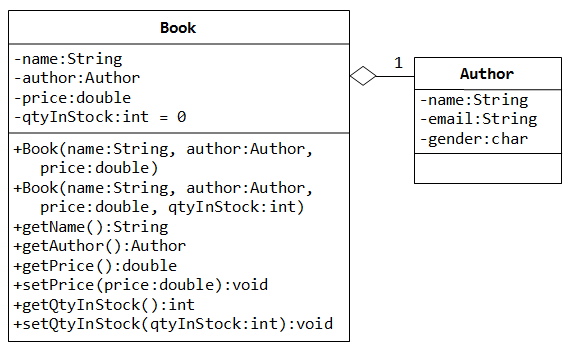
The learning objective of this lab is to understand and practice the concept of inheritance as well as the concept of composition in Java.

There are two ways to reuse the existing classes, namely, composition and inheritance. With composition, you define a new class, which is composed of existing classes. With inheritance, you derive a new class based on an existing class, with modifications or extensions. Composition is also expressed as Has-A relationship in Java. Some scenarios illustrate involvement of both these relationship and a few focus on just one of them.

In the following tasks, apply most appropriate concepts and solve the problems with the best possible approach/solution.

**Task #1:**

Implement the following structure by defining appropriate classes and relationships amongst them. Once you are finished with the class implementation, also provide a test class to demonstrate the capabilities of your class’ i.e: create multiple book objects and display them.



**Code:**

**Book Class:**

package com.company;  
  
*//Creating class Book*public class Book {  
  
 *//Declaring fields* private String name;  
 private Author author;  
 private double price;  
 private int qtyInStock;  
  
 *//Defining the three parameter constructor* Book(String *name*, Author *author*, double *price*){  
 this.name = *name*;  
 this.author = *author*;  
 this.price = *price*;  
 }  
 *//Defining the four parameter constructor* Book(String *name*, Author *author*, double *price*, int *qtyInStock*){  
 this.name = *name*;  
 this.author = *author*;  
 this.price = *price*;  
 this.qtyInStock = *qtyInStock*;  
 }  
  
 *//Name Getter* public String getName(){  
 return name;  
 }  
 *//Author Getter* public Author getAuthor(){  
 return author;  
 }  
 *//Price Getter* public double getPrice(){  
 return price;  
 }  
 *//Price Setter* public void setPrice(double *price*){  
 this.price = *price*;  
 }  
 *//Stock Setter* public void setQtyInStock(int *qtyInStock*){  
 this.qtyInStock = *qtyInStock*;  
 }  
 *//Stock Getter* public int getQtyInStock(){  
 return qtyInStock;  
 }  
 public String toString(){  
 return "Name of Book: " + name + "\n" + "Name of Author: " +  
 author + "\n" + "Price of Book: " +  
 price + "\n" + "Quantity in Stock: " + qtyInStock;  
 }  
}

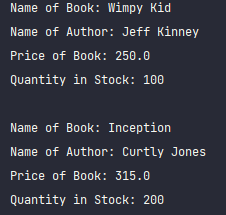
**Author Class:**

package com.company;  
  
*//Creating class Author*public class Author {  
  
 *//Declaring fields* private String name;  
 private String email;  
 private char gender;  
  
 *//Defining the constructor* Author(String *name*){  
 this.name = *name*;  
 }  
 public String toString(){  
 return name;  
 }  
  
}

**Test Class:**

package com.company;  
  
*//Creating the Test class*public class TestTask01 {  
  
 *//Main Method* public static void main(String[] args) {  
  
 *//Creating an instance of Book class and giving required inputs* Book b1 = new Book("Wimpy Kid", new Author("Jeff Kinney"), 250, 100);  
  
 *//Printing teh details of book 1* System.out.println(b1);  
  
 System.out.println();  
  
 *//Creating another object of Book class and giving inputs* Book b2 = new Book("Inception", new Author("Curtly Jones"), 315, 200);  
  
 *//Printing details of book 2* System.out.println(b2);  
 }  
}

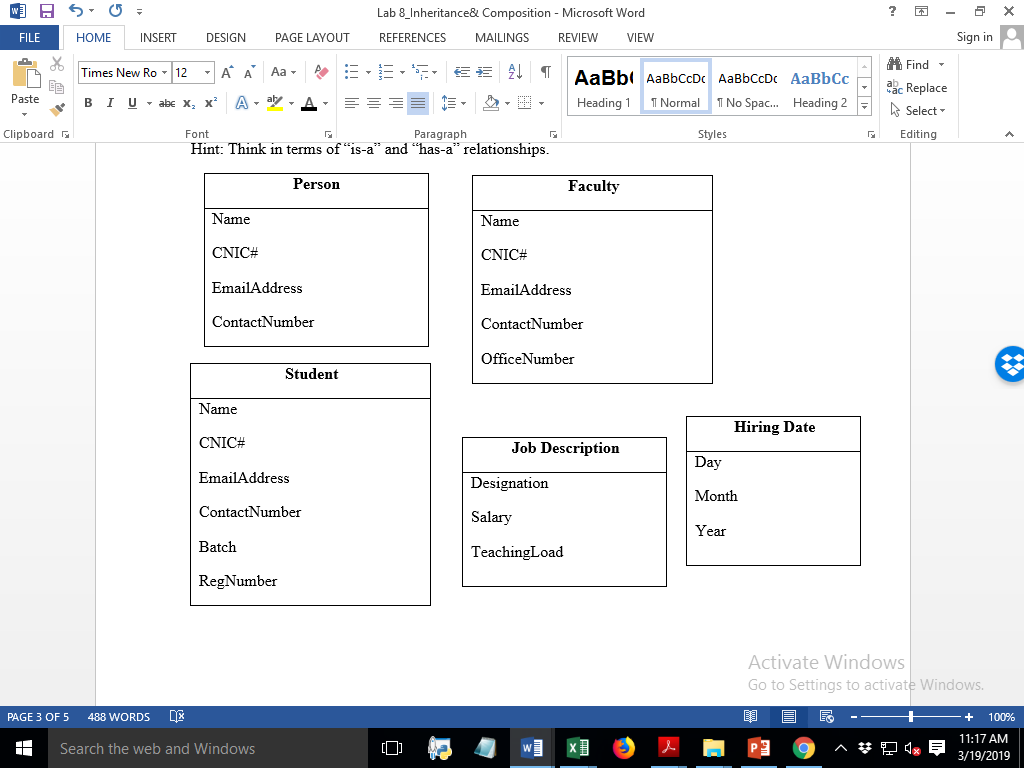
**Output Screenshot:**

****

**Task 2:**

The following partial UML class diagrams represent classes for Person, Student, Faculty Hiring Date and Job Description. Create these classes individually with the given data members and add few methods according to relevance on your own. Also create constructors and getter/setter methods. You are required to ensure maximum reusability while implementing these classes wherever possible. Additionally, write a Test class to create objects of each class for testing purposes.

Hint: Think in terms of **“is-a”** and **“has-a”** relationships.



**Code:**

**Person Class:**

package com.company;  
  
*//Creating parent class Person*public class Person {  
 *//Declaring fields* String name;  
 String cnic;  
 String email;  
 String contact\_no;  
  
 *//Defining the constructor* Person (String *name*, String *cnic*, String *email*, String *contact\_no*){  
 this.name = *name*;  
 this.cnic = *cnic*;  
 this.email = *email*;  
 this.contact\_no = *contact\_no*;  
 }  
  
 public void setName(String *name*){this.name = *name*;} *//Name setter* public void setCnic(String *cnic*){this.cnic = *cnic*;} *//Cnic setter* public void setEmail(String *email*){this.email = *email*;} *//Email setter* public void setContact\_no(String *contact\_no*){this.contact\_no = *contact\_no*;} *//Number setter* public String getName(){return name;} *//Name getter* public String getCnic(){return cnic;} *//Cnic getter* public String getEmail(){return email;} *//Email getter* public String getContact\_no(){return contact\_no;} *//Number getter  
  
 //Modifying the toString method* public String toString(){  
 return "Person's Name = " + name + "\n" + "CNIC = " + cnic + "\n" +  
 "Email Adress = " + email + "\n" + "Ph. No. = " + contact\_no;  
 }  
  
}

**Student Class:**

package com.company;  
  
*//Creating child class Student*public class Student extends Person{  
 *//Declaring fields* int batch;  
 int reg\_num;  
  
 *//Defining the constructor* Student (String *name*, String *cnic*, String *email*, String *contact\_no*, int *batch*, int *reg\_num*){  
 super(*name*, *cnic*, *email*, *contact\_no*);  
 this.batch = *batch*;  
 this.reg\_num = *reg\_num*;  
 }  
  
 *//Batch getter* public int getBatch() {  
 return batch;  
 }  
  
 *//Batch setter* public void setBatch(int *batch*) {  
 this.batch = *batch*;  
 }  
  
 *//Reg\_num getter* public int getReg\_num() {  
 return reg\_num;  
 }  
  
 *//Reg\_num setter* public void setReg\_num(int *reg\_num*) {  
 this.reg\_num = *reg\_num*;  
 }  
  
 *//Modifying the toString method* public String toString(){  
 return "Student Name = " + name + "\n" + "CNIC = " + cnic + "\n" +  
 "Email = " + email + "\n" + "Ph. No. = " + contact\_no + "\n" +  
 "Batch = " + batch + "\n" + "Registration No. = " + reg\_num;  
 }  
  
}

**Faculty Class:**

package com.company;  
  
*//Creating the child class Faculty*public class Faculty extends Person {  
 String office\_num;  
 JobDescription job;  
 HiringDate date;  
  
 *//Defining the constructor* Faculty(String *name*, String *cnic*, String *email*, String *contact\_no*, String *office\_num*, JobDescription *job*, HiringDate *date*){  
 super(*name*, *cnic*, *email*, *contact\_no*);  
 this.office\_num = *office\_num*;  
 this.job = *job*;  
 this.date = *date*;  
 }  
  
 *//office\_num getter* public String getOffice\_num() {  
 return office\_num;  
 }  
 *//Office\_num setter* public void setOffice\_num(String *office\_num*) {  
 this.office\_num = *office\_num*;  
 }  
 *//Job getter* public JobDescription getJob() {  
 return job;  
 }  
 *//Job setter* public void setJob(JobDescription *job*) {  
 this.job = *job*;  
 }  
 *//date getter* public HiringDate getDate() {  
 return date;  
 }  
 *//date setter* public void setDate(HiringDate *date*) {  
 this.date = *date*;  
 }  
  
 *//Modifying the toString method* public String toString(){  
 return "Member Name = " + name + "\n" + "CNIC = " + cnic + "\n" +  
 "Email = " + email + "\n" + "Ph. No. = " + contact\_no + "\n" +  
 "Office No. = " + office\_num + "\n" + job + "\n" + date;  
 }  
  
  
}

**JobDescription Class:**

package com.company;  
  
*//Creating class JobDescription*public class JobDescription {  
 *//Defining fields* String designation;  
 int salary;  
 String teaching\_load;  
  
 *//Defining the constructor* JobDescription(String *designation*, int *salary*, String *teaching\_load*){  
 this.designation = *designation*;  
 this.salary = *salary*;  
 this.teaching\_load = *teaching\_load*;  
 }  
  
 *//Modifying the toString method* public String toString(){  
 return "Designation = " + designation + "\n" + "Salary = " + salary + "\n" +  
 "Teaching Load = " + teaching\_load;  
 }  
}

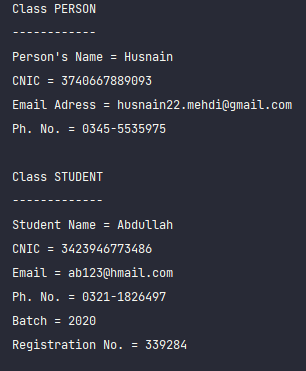
**HiringDate Class:**

package com.company;  
  
*//Creating the class HiringDate*public class HiringDate {  
 *//Declaring the fields* int day;  
 int month;  
 int year;  
  
 *//Defining the constructor* HiringDate(int *day*, int *month*, int *year*){  
 this.day = *day*;  
 this.month = *month*;  
 this.year = *year*;  
 }  
  
 *//Modifying the toString method* public String toString(){  
 return "Hiring Date = " + day + "/" + month + "/" + year;  
 }  
}

**Test Class:**

package com.company;  
  
*//Creating Test class*public class TestTask02 {  
  
 *//Main Method* public static void main(String[] *args*) {  
  
 *//Printing info of Person class* System.*out*.println("Class PERSON");  
 System.*out*.println("------------");  
 Person person = new Person("Husnain","3740667889093","husnain22.mehdi@gmail.com","0345-5535975");  
 System.*out*.println(person);  
  
 System.*out*.println();  
  
 *//Printing info of Student class* System.*out*.println("Class STUDENT");  
 System.*out*.println("-------------");  
 Student student = new Student("Abdullah","3423946773486","ab123@hmail.com","0321-1826497",2020,339284);  
 System.*out*.println(student);  
  
 System.*out*.println();  
  
 *//printing info of Faculty class* System.*out*.println("Class FACULTY");  
 System.*out*.println("-------------");  
 Faculty faculty = new Faculty("Zain","3826856298746","zain23@yahoo.com","0311-7395752","051-9732936",new JobDescription("Professor",300000,"6 courses per year"),new HiringDate(12,3,2011));  
 System.*out*.println(faculty);  
  
 }  
}

**Output Screenshot:**

****



## Grade Criteria

This lab is graded as per following criteria. Min marks: 0. Max marks: 10.

|  |  |  |
| --- | --- | --- |
| **Activity** | **Minimum** | **Maximum** |
| Documentation with clearly defined understanding of the lab task and approach. Code clarity with clean, formatted and commented code. | 0 | 2 |
| Complete Functionality Implementation/ Error Free Execution | 0 | 8 |
| **Total** | **0** | **10** |

**Hand in**

Hand in the source code from this lab at the appropriate location on the LMS system.

**To Receive Credit**

1. Comment your program heavily. Intelligent comments and a clean, readable formatting of your code account for 20% of your grade.
2. The lab time is not intended as free time for working on your programming/other assignments.