

## Question 1:

What is the output of the following program :

```
public class Employee {
    private String name;
    private int workedHours;
    private int hourlyPay = 125;

    public Employee(String inName, int n) {
        name = inName;
        workedHours = 2*n;
    }

    public String getName() { return name; }

    public double ComputeSalary() {
        try {
            if (workedHours>0){
                System.out.println ( "No Problem Everything is OK for " + name);
                return workedHours * hourlyPay;
            }
            System.out.println ( "yes There is a Problem for " + name);
            throw new ArithmeticException();
        }
        catch ( ArithmeticException exception ) {
            System.out.println ( "Problem with Worked hours" + workedHours);
            throw new ArithmeticException();
        }
    }

    public void CheckSalary() throws Exception {
        try {
            double x =ComputeSalary();
            System.out.println("The result is : " + workedHours);
        }
        catch ( ArithmeticException e ) {
            System.out.println("Exception handled in the catcher" );
            throw new ArithmeticException();
        }
    }
}

//===== Main =====
public class ExceptionTestMain {
    public static void main(String[] args) {
        Employee a, b ;
        double result ;
        a = new Employee("Mohammad", 9);
        b = new Employee("Ali", -9);
        try {
            a.CheckSalary ();
        }
        catch (Exception e) {
            System.out.print("Problem when checking the salary of " + a.getName());
        }
        try {
            b.CheckSalary ();
        } catch (Exception e) {
            System.out.print("Problem when checking the salary of " + b.getName());
        }
    }
}
```

### Question 2:

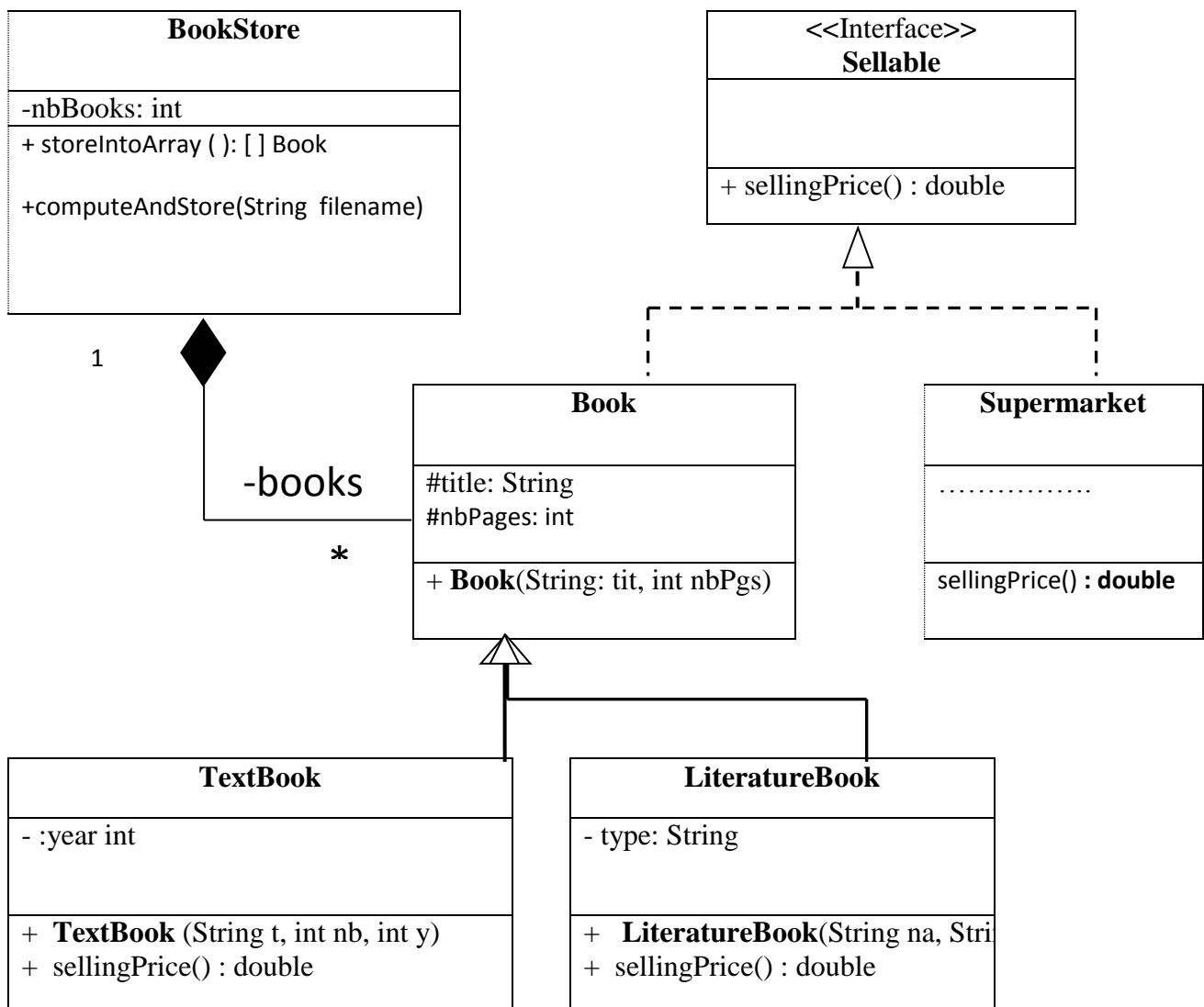
Your company assigned you the task of designing a hospital management program that can be sold later to many client hospitals. The hospital is characterized by its name, address and number of beds besides lists of patients, employees and departments.

Patients and employees have names and phone numbers. Patients have also a patient id and assigned a doctor. Each employee has an employee id, and works in a certain department. Employees can be either doctors or are characterized by specialty, or a staff characterized by job name. A department has a name and building number.

Draw the UML diagram of the classes Hospital, Department, Patient, Employee, Doctor, and Staff described above. In this diagram show clearly the type of relationship between these classes and assure that attributes can be accessed solely through methods.

### Question 3:

Consider the following UML class diagram



The interface **Sellable** contains the following methods:

- *sellingPrice()* : this method returns the selling price

The class **Book** contains the following attributes and methods:

- title : title of the book
- nbPages : number of the pages
- *sellingPrice()* : this method returns sale price of the Book

The class **TextBook** contains the following attributes and methods:

- year: published year of the book.
- *sellingPrice()*  
The sale price of the Book is calculated as follows

$$\text{selling price} = \text{nbPages} * 2 + (\text{year} \% 2000) * 30$$

Write in Java the interface **Sellable**, the class **Book**, and the class **TextBook**.

Note: You can call getters and setters without implementation. Assume that the class **Supermarket** and **TextBook** are implemented.

Question 4:

We want to manage a bookstore containing books, as in the previous UML class diagram.

**Class BookStore**: contains the following attributes and methods:

- nbBooks : represents number of the book in the bookstore
- books : represents the array of book objects

***public Book[] storeIntoArray()***

This method reads Book objects from the object file "*book.data*" and returns an array containing only *TextBook* objects having published year less than 1890.

***public void computeAndStore(String filename)***

This method stores into a binary file called *filename* the following information: (i) number of books in bookstore, (ii) average selling price of Textbooks, and (ii) number of literature books.

Write in Java the two methods ***storeIntoArray*** and ***computeAndStore***. Consider that the other methods of the class List are implemented.