CSA0961 – JAVA

JF PROJECT 4

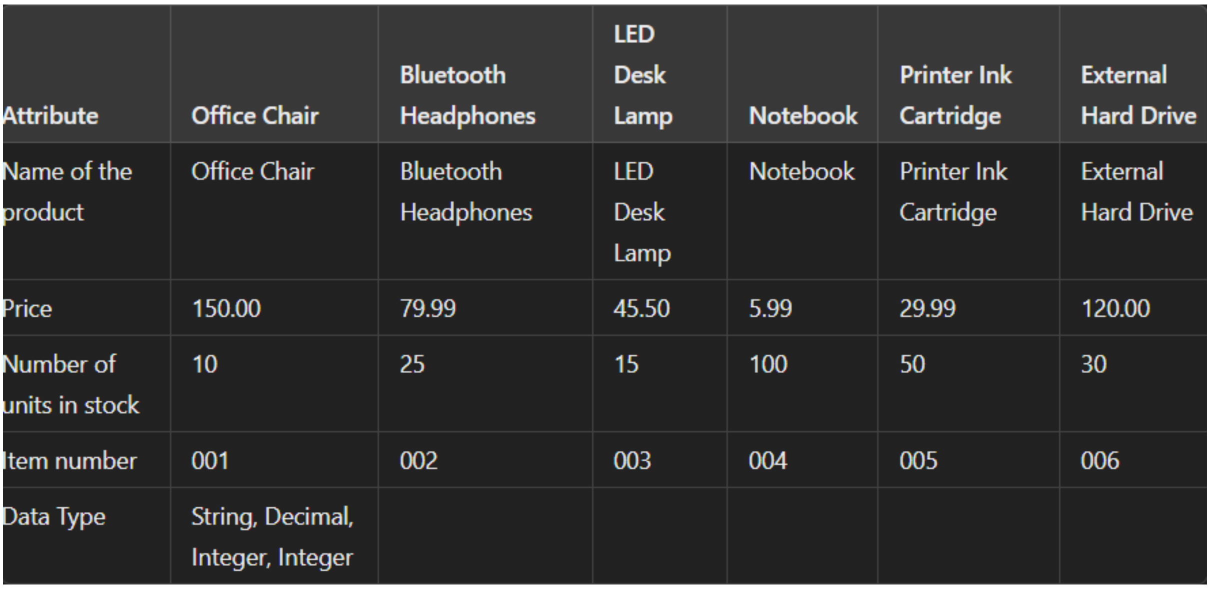
1. For the first part of the project you are required to think about what your inventory system will store. a. Think of specific products that lend themselves to be stored in an inventory (for example, products in your home, school, or workplace: they could be from the following categories; office supplies, music CDs, DVD movies, or software). Write a list of at least 6 products that you want to store in your system, this project could be used to store a wide range of products.
   b. For each of the products that you identified, complete the following table:
   Attribute Sample Data Name of the product (the value that will identify the product in your system).
   Price (this value holds the price that each item will be sold for).
   Number of units in stock (this value will store how many of each product item is currently in stock).
   Item number (used to uniquely identify the product in your system).
   This table gives you an understanding of the type of data that you will want to store for the attributes of each product. It's useful to do this so you have a clear understanding of the data that you will be working with!

1. Product List and Data Attributes

a. List of Products:

1. **Office Chair**
2. **Bluetooth Headphones**
3. **LED Desk Lamp**
4. **Notebook**
5. **Printer Ink Cartridge**
6. **External Hard Drive**

B. Data Table :

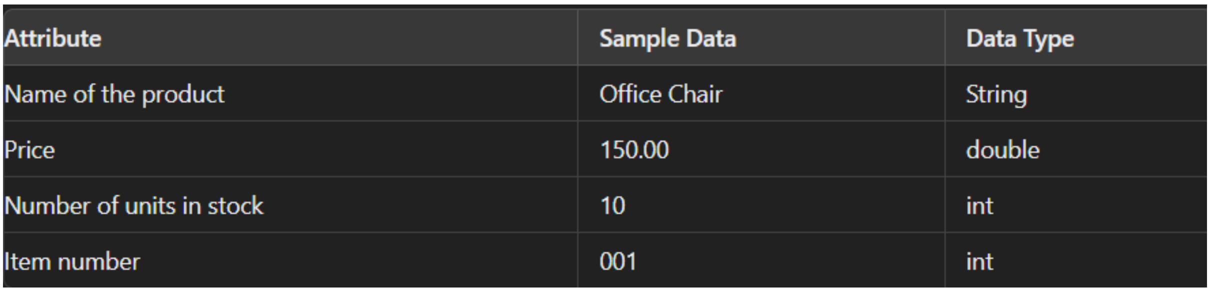


Attribute	Office Chair	Bluetooth Headphones	LED Desk Lamp	Notebook	Printer Ink Cartridge	External Hard Drive
Name of the product	Office Chair	Bluetooth Headphones	LED Desk Lamp	Notebook	Printer Ink Cartridge	External Hard Drive
Price	150.00	79.99	45.50	5.99	29.99	120.00
Number of units in stock	10	25	15	100	50	30
Item number	001	002	003	004	005	006
Data Type	String, Decimal, Integer, Integer					

2. The next step is to think about the correct data types that you will use to store the values in your system. To do this add another column to your table that will identify the correct data type for each value that you have identified.

Attribute	Sample Data	Data Type
Name of the product		
Price		
Number of units in stock		
Item number		

Data Type for Arttibutes



Attribute	Sample Data	Data Type
Name of the product	Office Chair	String
Price	150.00	double
Number of units in stock	10	int
Item number	001	int

3.Create a project named inventory.

Create a project named inventory.

4.Create an object class called Product.

5. Add the following private instance fields (variables) by using the data types you identified in task 2:
a. item number

b. the name of the product

c. the number of units in stock

d. the price of each unit

6. Add a comment above the instance field declarations that states:

//Instance field declarations

7. Create two constructors: a. A default constructor without parameters that will allow the compiler to initialize the fields to their default values. Add a comment above your constructor that explains the purpose of the code.

b. Overload the default constructor by creating a constructor with parameters for all four of the class' instance fields so that they can be initialized with values from the driver class. The parameters should be named; number, name, qty, price. You should use the this.instance_field_name notation to quantify the objects instance field: this.name = name;

8. Write getter/accessor and setter/mutator methods for each of the four instance variables. Write getter/accessor and setter/mutator methods for each of the four instance variables. Add comments above them to explain their purpose

. 9. Override the toString() method from the object class to show a description of each Product object that includes the instance field values in the following format: Item Number : 1 Name : Greatest Hits Quantity in stock: 25 Price : 9.99

```
// File: Product.java
public class Product {
    // Instance field declarations
    private int itemNumber;
    private String name;
    private int numberOfUnitsInStock;
    private double price;

    // Default constructor
    public Product() {
        // Initializes instance variables to default values
        this.itemNumber = 0;
        this.name = "";
        this.numberOfUnitsInStock = 0;
        this.price = 0.0;
    }

    // Parameterized constructor
    public Product(int number, String name, int qty, double price) {
        this.itemNumber = number;
        this.name = name;
        this.numberOfUnitsInStock = qty;
        this.price = price;
    }

    // Getter for itemNumber
    public int getItemNumber() {
        return itemNumber;
    }
```

```
// Setter for itemNumber
public void setItemNumber(int itemNumber) {
    this.itemNumber = itemNumber;
}

// Getter for name
public String getName() {
    return name;
}

// Setter for name
public void setName(String name) {
    this.name = name;
}

// Getter for numberOfUnitsInStock
public int getNumberOfUnitsInStock() {
    return numberOfUnitsInStock;
}

// Setter for numberOfUnitsInStock
public void setNumberOfUnitsInStock(int numberOfUnitsInStock) {
    this.numberOfUnitsInStock = numberOfUnitsInStock;
}

// Getter for price
public double getPrice() {
    return price;
}

// Setter for price
public void setPrice(double price) {
        this.price = price;
    }

    // Override toString() method
    @Override
    public String toString() {
        return "Item Number       : " + itemNumber + "\n" +
               "Name              : " + name + "\n" +
               "Quantity in stock: " + numberOfUnitsInStock + "\n" +
               "Price             : " + price;
    }
}
```

10. Create a Java main class called ProductTester.

11. Create and initialize six Product objects based on the list that you created in task

1. a. Two of the Products should be created using the default constructor.

b. The other four should be created by providing values for the arguments that match the parameters of the constructor.

12. Using the ProductTester class, display the details of each product to the console.

13. Save your project.

```
// File: ProductTester.java
public class ProductTester {
    public static void main(String[] args) {
        // Creating products using default constructor
        Product chair = new Product();
        Product lamp = new Product();

        // Creating products using parameterized constructor
        Product headphones = new Product(2, "Bluetooth Headphones", 25, 79.99);
        Product notebook = new Product(4, "Notebook", 100, 5.99);
        Product inkCartridge = new Product(5, "Printer Ink Cartridge", 50, 29.99);
        Product hardDrive = new Product(6, "External Hard Drive", 30, 120.00);

        // Displaying details of each product
        System.out.println(chair);
        System.out.println();
        System.out.println(lamp);
        System.out.println();
        System.out.println(headphones);
        System.out.println();
        System.out.println(notebook);
        System.out.println();
        System.out.println(inkCartridge);
        System.out.println();
        System.out.println(hardDrive);
    }
}
```