

1) Goals and Overview

- 1) To create a class with attributes, constructors and behaviors
- 2) To create an array of objects
- 3) Instance methods, static variables and static methods
- 4) **Create a new project for the quiz and call it Ecommerce**

Overview: In this quiz, you will be implementing a Product Class that's a blueprint of e-commerce product's attributes and behaviors. A Shop class that will contain the main function and 5 product objects. You will create a shopping menu for the user using an array of these 5 products.

2) The Product Class

- 1) Product will have the following attributes:

- String name
- double price
- String category
- int quantity
- char size
- String author
- String color

- 2) Constructors:

- Some products are apparel so they will have **name, price, category, quantity, size, color**
- Some products are books and they'll have **name, price, category, quantity, author**
- Some products are soap and they'll have **name, price, category and quantity**
- We will not create the default constructor

- 3) **All** attributes will be private and they will have public getters and setters. You will be using getters and setters when other classes access Product's attributes. Inside the Product class, you should be using **this.attributeName**
- 4) Create a toString method to print all the attributes. You can use the eclipse shortcut to create constructors, getters setters and toString method.
- 5) Instance Method : **buy()** - This method will not take any parameters and will return a boolean. Decide the method signature accordingly. Inside this method:
 - Check if the quantity of the product is greater than 0
 - If it is, reduce the quantity by 1 (use appropriate operator that would change the product's quantity in memory) and return true,
 - Otherwise, return false

3) Shop Class

- 1) Attributes:
 - A **static** double variable called **totalAmount** initialized to 0
 - A **static** array called **inventory** of 5 Product objects

We'll learn to declare and initialize an array of objects in one line. We can use this technique in our future projects when it's appropriate. Check point 8 below for syntax before writing the array in your code.
- 2) 0th index of inventory will reference to a new Product object and the attributes would be **name:** Business Ritual Women's Long Sleeve Top , **price:** 65. 39, **category:** Apparel, **quantity:** 4, **size:** L, **color:** Blue
- 3) 1st index of inventory will reference to a new Product object and the attributes **name:** The Art of Computer Programming , **price:** 190. 54, **category:** Books, **quantity:** 9, **author:** Donald E. Knuth
- 4) 2nd index of inventory will reference to a new Product object and the attributes **name:** Men's Business Casual Shirt , **price:** 57. 10, **category:** Apparel, **quantity:** 2, **size:** M, **color:** Teal

- 5) 3rd index of inventory will reference to a new Product object and the attributes **name:** Organic Turmeric Soap , **price:** 11. 25, **category:** Bath & Beauty, **quantity:** 1
- 6) 4th index of inventory will reference to a new Product object the attributes **name:** Head First Design Pattern, **price:** 37. 35, **category:** Books, **quantity:** 3, **author:** Eric Freeman
- 7) To declare and initialize an array of objects in one line, you can use the following syntax:

```
ClassName[] arrayName = {new ClassName(parameters), new  
ClassName(parameters)...}
```

Example for an array called employees with 3 FullTime objects:

```
static FullTime[] employees = {new FullTime("XYZ", 12.5,  
"22"), new FullTime("ABC", 8.2, "55-55"), new  
FullTime("TTT", 5.17, "83-55")};
```

- 8) A **static** method called displayMenu() which would return nothing:
- It prints the statement `"\nEnter the product index (0 to 4) you want to buy. To exit the shopping cart, enter any number other than 0 to 4"`
 - It prints the product index : product Name. To display the *index number* : *product name* in the menu, use a **for** loop to loop through the inventory array. Inside the loop print your index : inventory[index]'s getter for name attribute. Output in section 5 shows how your menu will look like.
- 9) Inside the main method:
- Declare an int variable called **userInput**
 - You would need a Scanner object to take user input. Make sure to close the input stream before your main method ends
 - Run a menu option inside a **do while** loop that runs as long as userInput is between 0 and 4. Inside the do while loop:

- Call the `displayMenu()` Method
- Take an integer as **`userInput`**
- **check if `userInput` is between 0 and 4**
- Inside the if statement, you'll have another if statement to call **`inventory[userInput]`**'s buy method – the method call should be within an if condition (Since the buy method returns boolean you can put the method call as `if (method call)`). If buy returns true, print **"Your purchase was successful"** and add that product's price to **`totalAmount`**. Otherwise, print **"Sorry! This product is out of stock."**
- ***NOTE:** You will be checking the condition between 0 to 4 twice, once for the while loop and once inside the loop. This method can be more efficient using the extra credit section which is optional.*
- Once the do while loop has ended, print "Thank you for shopping with us! The total amount is " + `totalAmount`

4) Notes, Implementation and Submission

- 1) **Time limit is 3 hours with a 15 minute break in between**
- 2) Hint: To call an object's method that returns boolean, inside an if statement, you would write `if (employee[0].isManager())`
- 3) Design the flow of your program on a white sheet first
- 4) Write comments and indent your code
- 5) Your instructor or your colleague can only help you with logical errors or understanding of the program. No one is supposed to help/seek help for syntax errors, semicolon, typo, curly braces or datatype errors.
- 6) Do not google anything except syntax if required, and keep your instructor informed. You can use your textbook, class & homework materials/programs for reference.
- 7) **Extra Credit** : Change the do-while loop to a forever **`while(true)`** loop. Break (keyword is **`break`**) the loop if `userInput` is not between 0 and 4.

8) Submit both .java files

5) Output 1 :

Enter the product index (0 to 4) you want to buy. To exit the shopping cart, enter any number other than 0 to 4

0 : Business Ritual Women's Long Sleeve Top
1 : The Art of Computer Programming
2 : Men's Business Casual Shirt
3 : Organic Turmeric Soap
4 : Head First Design Pattern

0

Your purchase was successful

Enter the product index (0 to 4) you want to buy. To exit the shopping cart, enter any number other than 0 to 4

0 : Business Ritual Women's Long Sleeve Top
1 : The Art of Computer Programming
2 : Men's Business Casual Shirt
3 : Organic Turmeric Soap
4 : Head First Design Pattern

3

Your purchase was successful

Enter the product index (0 to 4) you want to buy. To exit the shopping cart, enter any number other than 0 to 4

0 : Business Ritual Women's Long Sleeve Top
1 : The Art of Computer Programming
2 : Men's Business Casual Shirt
3 : Organic Turmeric Soap
4 : Head First Design Pattern

3

Sorry! This product is out of stock

Enter the product index (0 to 4) you want to buy. To exit the shopping cart, enter any number other than 0 to 4

0 : Business Ritual Women's Long Sleeve Top
1 : The Art of Computer Programming
2 : Men's Business Casual Shirt
3 : Organic Turmeric Soap

4 : Head First Design Pattern

1

Your purchase was successful

Enter the product index (0 to 4) you want to buy. To exit the shopping cart, enter any number other than 0 to 4

0 : Business Ritual Women's Long Sleeve Top

1 : The Art of Computer Programming

2 : Men's Business Casual Shirt

3 : Organic Turmeric Soap

4 : Head First Design Pattern

77

Thank you for shopping with us! The total amount is 267.18