Hyperiondev

TASK

Capstone Project IV — Object-Oriented Programming

Visit our website

Introduction

WELCOME TO THE FINAL CAPSTONE!

This is the penultimate Capstone Project for this bootcamp! This Capstone is a milestone in your learning so far. In this project, you will be using object-oriented programming to create a solution for a real-world problem. Remember, it is worth putting some extra time and effort into this project. It will eventually become part of your developer portfolio.

DEVELOPER PORTFOLIO

Developers who have the edge are those who find ways to apply their newfound skills from the get-go. A **developer portfolio** (a collection of online creations that you have made) allows you to demonstrate your skills rather than just telling people about them. It's a way of bringing your CV to life and introducing yourself to the world. As you learn more skills and put these into practice, each project that you complete will become more efficient and eye-catching.

Object-oriented programming is one of the most important programming paradigms today! Prospective employers will want evidence that a Software Engineer is comfortable using object-oriented programming. This application series offers you the means to an object-oriented program to add to your developer portfolio.

THE TASK AT HAND

Let us assume that you work for a Nike warehouse. Your role is that of a store master, meaning you are responsible for managing the warehouse, and more importantly, doing stock taking. To optimise your delivery time and for neater organisation, you have decided to use your Python knowledge to get an overview of what each stock-taking session entailed.

Nike warehouses store the following information for each stock-taking list:

- Country
- Code
- Product
- Cost
- Quantity
- Value

Other store managers (in other regions) would like to be able to use your program to do the following:

- Search products by code.
- Determine the product with the lowest quantity and restock it.
- Determine the product with the highest quantity.
- Calculate the value of each item entry, based on the quantity and cost of the item. The value is calculated by multiplying the cost by the quantity for each item entered.

Before you begin

A key focus of this project will be ensuring that your code is correct, well-formatted and readable. In this regard, make sure that you do the following before submitting your work:

1. Make sure that you have identified and removed all syntax, runtime and logical errors from your code.

2. Make sure that your code is readable. To ensure this, add comments to your code, use descriptive variable names and make good use of whitespace and indentation. See **this style guide** to see how classes and methods should be named and how your program should be formatted.

3. Make sure that your code is as efficient as possible. How you choose to write code to create the solution to the specified problem is up to you. However, make sure that you write your code as efficiently as possible.

4. Make sure that all output that your program provides to the user is easy to read and understand. Labelling all data that you output (whether in text files or to the screen) is essential to make the data your program produces more user-friendly.

Compulsory Task

Follow these steps:

- Code a Python program that will read from the text file **inventory.txt** and perform the following on the data, to prepare for presentation to your managers:

    - Create a file named **inventory.py**, where a Shoe class should be defined.

    - Create a class named Shoes with the following attributes:
        - country,
        - code,
        - product,
        - cost, and
        - Quantity.

    - Inside this class define the following function:

        - **get_cost** - which return the cost of the shoe
        - **get_quanty** - which return the quantity of the shoes
        - **__str__** - This function returns a string representation of a class.

    - Outside this class create a variable with an empty list. This variable will be used to store a list of shoes objects

    - Then you must define the following functions:

        - **read_shoes_data** - this function will open the file inventory.txt and read the data from this file the create shoes object and append this object into the shoes list. one line in this file represents data to create one object of shoes. You must use the try except in this function for error handling.

        - **capture_shoes** - this function will allow a user to capture data about a shoe and use this data to create a shoe object and append this object inside the shoe list.

        - **view_all** - this function will iterate over all the shoes list and print the details of the shoes that you return from the __str__

function. (**Optional:** You can organise you data in a table format by using Python's tabulate module)

- **re_stock** - this function will find the shoe object with the lowest quantity, which is the shoes that need to be restocked. Ask the user if he wants to add the quantity of these shoes and then update it. This quantity should be updated on the file for this shoe.

- **seach_shoe** - This function will search for a shoe from the list using the shoe code and return this object so that it will be printed

- **value_per_item** - this function will calculate the total value for each item . (Please keep the formula for value in mind; **value = cost * quantity**). Print this information on the console for all the shoes.

- **highest_qty** - Write code to determine the product with the highest quantity and print this shoe as being for sale.

o Now in your main create a menu that executes each function above. This menu should be inside the while loop. Be creative.

Rate us

Share your thoughts

Hyperion strives to provide internationally-excellent course content that helps you achieve your learning outcomes.

Think that the content of this task, or this course as a whole, can be improved, or think we've done a good job?

Click here to share your thoughts anonymously.