Hyperiondev

TASK

Capstone Project - Object-Oriented Programming

Visit our website

Introduction

WELCOME TO THE OBJECT-ORIENTED PROGRAMMING CAPSTONE PROJECT!

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

DEVELOPER PORTFOLIO

Object-oriented programming is one of the most important programming paradigms today! Prospective employers will want evidence that a software engineer is comfortable using OOP.

THE TASK AT HAND

Let us assume that you work as a store manager for a Nike warehouse. As a store manager, you are responsible for managing the warehouse, and more importantly, performing stock-taking. To optimise your delivery time and for improved organisation, you have decided to use your Python knowledge to get an overview of what each stock-taking session entails.

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 total value of each stock item. The total value is calculated by multiplying the cost by the quantity for each item entered into the system.

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. Use the **PEP 8 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:

- For this task, use the file template provided named **inventory.py**.
- Code a program that will read from the text file **inventory.txt** and perform the following on the data, to prepare for presentation to your managers:
- Inside this file, you will find a class named `Shoe` with the following attributes:
    - `country`
    - `code`
    - `product`
    - `cost`
    - `quantity`
- Inside this class define the following methods:
    - `get_cost` - Returns the cost of the shoes.
    - `get_quantity` - Returns the quantity of the shoes.
    - `__str__` - This method returns a string representation of a class.
- Outside this class create a `shoes_list` variable with an empty list. This variable will be used to store a list of shoe objects.
- Then you must define the following functions outside the class:
    - `read_shoes_data` - This function will open the file **inventory.txt** and read the data from this file, then create a shoes object with this data and append this object to the shoe 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. Remember to skip the first line using your code.
    - `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 the shoe list and print the details of the shoes returned from the `__str__` function. (**Optional**: you can organise your 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 are the shoes that need to be re-stocked. Ask the

user if they want to add this quantity of 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.

- Now 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.