Class documentation:

My class is called Shop. It will be focusing on shirts, their colors, size, and brand. The purpose of this class is to find the cost of each aspect of a shirt and how much it would cost if you buy multiple shirts of the same kind. Additionally, it will tell you whether a certain size, brand or color is available or not. Above the class are dictionaries of the prices of each element of a shirt. Additionally, the elements are separated by color, size and brand.

The class attribute is tshirt=”clothes” which indicates that the items in the class are clothes. The data attribute self._color=color is the variable that shows the color of the shirt. The data attribute self._size=size is the variable that shows the size of the shirt. The data attribute self._brand=brand is the variable that shows the brand of the shirt. The data attribute self.total() tells you the cost of one shirt.

For def __str__(self), it returns a string that tells you the cost of one clothing item. In the function it returns self.total which is the function that adds up the chosen size, color and brand together. For def set._total(self), it is a private attribute and it sets the value and equation to create the sum of the whole shirt. It sums up the prices of the variables self._color, self._size and self._brand. Since the prices are already listed above, when you input the size, color and brand on the main function, it will know how much the elements are, which are added up. The function def get_total(self) is a set method and it returns self.total, which is the total of the item imputed in the main. It takes the self.total made by def total() and returns it. The function def total(self) adds up the prices of each element of the shirt. It adds up the price of the color, size, and brand of the item you input. After that, you can return it. The function is_it_available(self, color, size, brand) has the arguments “color”, “size”, and “brand”. In this function, you are inputting sizes, colors and brands and making sure if the shop has it in their inventory. If the inputs are available in the dictionary above the class, then it will return True, if not then it will return false and print out a message. The function def total_price(self, color, size, brand, amount_purchased) calculates the overall total price of the items chosen and bought. The amount_purchased indicates how many of the same items you want to buy, so if you want to buy 3 of the same items, you multiply the prices of each element of the item by 3 or ammount_purchased. The total will be returned as a string, telling you the overall total amount. The main function is where you input what you want to buy or what you want to make sure is available. Additionally, you can also find out the total overall cost if you want to buy multiple items. The inputs you need is just self._color, self._brand and self._size. Another input for def total_price would be the number of items you want.

Demo Program Documentation:

When you call the class and choose a color, size and brand from the dictionary above the class, it will return and print you the total cost of one item. When you call item.is_it_available and write something that is not from the dictionary, then it will tell you that it is not available. The options you have is “pink’, “black”, “white”, “small”, “medium”, “large”, “zara”, “uniqlo” and “gucci”. If you input anything else for is_it_available, it will print that the item not available. Additionally, if you use total_price, you write down the color, size,brand and the number or items you want. For example, if you write “pink”, “large”, “zara”, 3, you will get $90.

If you are running in the terminal, you just type in python3 and the file name.