

sandeepsuryaprasad / python_tutorials Private[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Security](#) [Ins](#)

master ▾

python_tutorials / 10_oops /
3_shopping_cart_class.py / <> Jump to ▾

Go to file

...



Sandeep Suryaprasad clean-up

Latest commit 19b6a56 on 29 Mar

[History](#)

0 contributors

35 lines (31 sloc) | 1.33 KB

Raw

Blame



```
1 # Shopping Cart Class
2 class ShoppingCart:
3     # Class Variables
4     products = {'iPhone': 5, 'iMac': 3, 'iWatch': 2, 'iPad': 4}
5     prices = {'iPhone': 800, 'iMac': 2500, 'iWatch': 3000, 'iPad': 3500}
6     def __init__(self):
7         self.cart = [ ]
8
9     def add_item(self, name, quantity):
10         if name not in self.__class__.products.keys():
11             raise Exception('Product Not Available')
12         elif self.__class__.products[name] >= quantity:
13             self.cart.append({'name': name, 'price': self.__class__.prices[name], 'quantity': self.__class__.products[name] - quantity})
14             self.__class__.products[name] -= quantity
15         else:
16             raise ValueError('Product Out of Stock')
17
18     def total_cost(self):
19         return sum([item['price'] * item['quantity'] for item in self.cart])
20
21     def remove_item(self, name):
22         for item in self.cart:
23             if name == item['name']:
24                 if item['quantity'] == 1:
25                     self.cart.remove(item)
26                 else:
27                     item['quantity'] -= 1
```

```
28
29     def summary(self):
30         print(f'{"name":>20} {"price":>20} {"quantity":>20}')
31         print('-' * 65)
32         for item in self.cart:
33             print(f'{item["name"]:>20} {item["price"]:>20} $ {item["quantity"]>20}')
34             print('-' * 65)
35         print(f'{"Total Cost":>20}{self.total_cost():>20} $')
```