

PYTHON CLASS II

REVIEW

VENDING MACHINE PROJECT



REVIEW - DEBUGGING

- Go to Mr. Hunter's Github page → <https://github.com/davidcbhunter/POP2022>
- Go to the Debugging Practice folder.
- Copy Debugging_1_name_errors.py and Debugging_2_variable_errors.py
- Fix the files so that they run and work correctly.

Waga hai wa yurusan!
Bad programming!



VENDING MACHINE PROJECT

- Make 5 variables for different products (product1 = "Black Coffee", etc)
- Make 5 variables for different prices (price1 = 130, etc)
- Make 5 variables for amounts (amount1 = 50, etc). This is the number of cans or bottles in the vending machine.
- Make one variable for the money in the vending machine.
- Make a function that prints products and prices. Call the function ShowProducts.

Black Coffee	Milk Coffee	Juice	Energy Drink
130	130	160	220

- Ask the customer to pick a product. (Use input)
- If the product is OK and the amount is 1 or more, continue. Otherwise, print a message and stop.
- Ask the customer to enter money. (Use input. Remember to cast the input to an int)
- Check that the money is enough. If it is not, print a message.
- If the money is higher than the price, give change to the customer.
- If the money is enough, reduce the amount by 1.

Challenge: Make these functions: `IsEnoughMoney(cp,cm)` → returns if cm is enough money to buy cp
`IsProduct(cp)` → returns if cp is product1 / 2/3/4/5
`Buy(cp,cm)` → contains the code here

VENDING MACHINE PROJECT

Otsukare!

- Make 5 variables for different products (product1 = "Black Coffee", etc)
 - Make 5 variables for different prices (price1 = 130, etc)
 - Make 5 variables for amounts (amount1 = 50, etc) This is the number of cans or bottles in the vending machine.
 - Make one variable for the money in the vending machine.
 - Make a function that prints products and prices. Call the function ShowProducts.
- | | | | |
|--------------|-------------|-------|--------------|
| Black Coffee | Milk Coffee | Juice | Energy Drink |
| 130 | 130 | 160 | 220 |
- Ask the customer to pick a product. (Use input)
 - If the product is OK and the amount is 1 or more, continue. Otherwise, print a message and stop.
 - Ask the customer to enter money. (Use input. Remember to cast the input to an int)
 - Check that the money is enough. If it is not, print a message. If the money is higher than the price, give change to the customer.
 - If the money is enough, reduce the amount by 1.

