[Chun Yin Tsoi](https://csueb.instructure.com/courses/33151/users/65635) (John)

[Hasibullah Ayoubi](https://csueb.instructure.com/courses/33151/users/82946)

[Ian Trawick](https://csueb.instructure.com/courses/33151/users/13626)

Indraneel [Parthasarathy](https://csueb.instructure.com/courses/33151/users/77914)

[Likitha Kommineni](https://csueb.instructure.com/courses/33151/users/77257)

Matthew Carreon

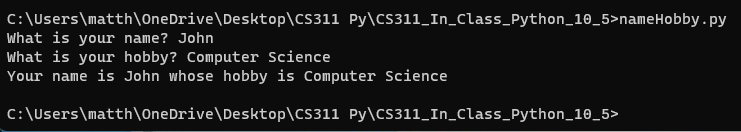
In class Python Practice

1) Write a Python program that asks the user to enter their name and hobby. The program should then display this information.

name = input("What is your name? ")

hobby = input ("What is your hobby? ")

print("Your name is " + name + " whose hobby is " + hobby)



2) One acre of land is equivalent to 43,560 square feet. Write a Python function that asks the user to enter a number of acres. The function should calculate the equivalent number of square feet and the price assuming land costs $35/square feet.

def acre\_to\_sqft(acres):

sqft = acres \* 43560

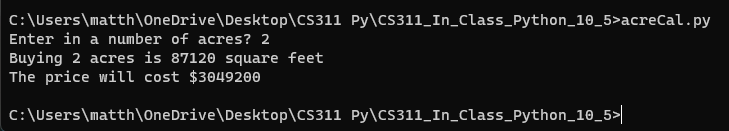
price = sqft \* 35

print("Buying " + str(acres) + " acres is " + str(sqft) + " square feet")

print("The price will cost $" + str(price))

numOfAcre = float(input("Enter in a number of acres? "))

acre\_to\_sqft(numOfAcre)



3) Write a function that uses a for loop to create a list of 5 grocery items. Ask the user to type in a grocery item and indicate whether that item is in the list.

groceryList = []

#loop def

def loop():

for i in range (5):

item = input("Enter a grocery item: ")

groceryList.append(item)

loop()

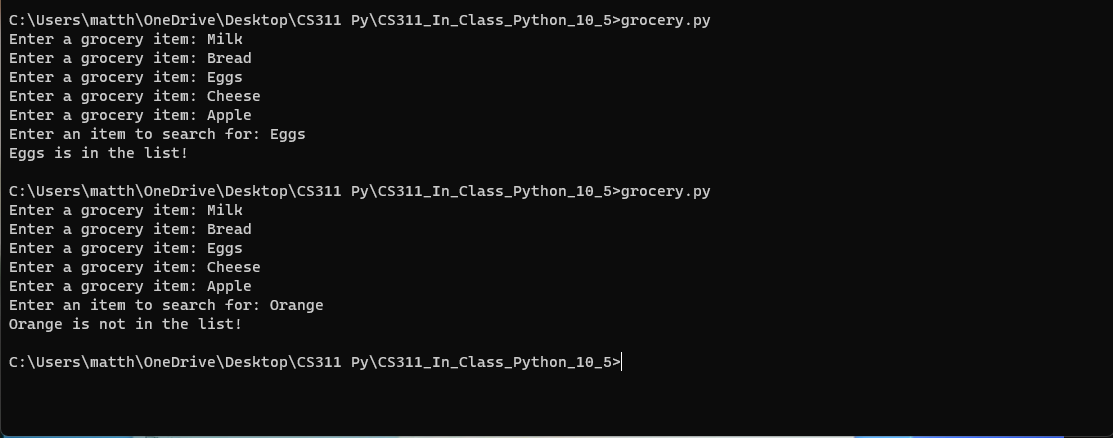
searchItem = input("Enter an item to search for: ")

if searchItem in groceryList:

print(searchItem + " is in the list!")

else:

print(searchItem + " is not in the list!")



4) Write a class called student with Name, GPA, major. Write methods to access these values. Create a student object and test your functions.

#Create a class name student

class Student:

def \_\_init\_\_(self, name, gpa, major):

self.name = name

self.gpa = gpa

self.major = major

def get\_name(self):

return self.name

def get\_gpa(self):

return self.gpa

def get\_major(self):

return self.major

#Create a student object

student1 = Student("John Doe", 2.0, "Computer Science")

#Methods testing

print(student1.get\_name())

print(student1.get\_gpa())

print(student1.get\_major())

