



python

The intermediaries



W3Schools is my main resource for making these slides and I **HEAVILY** recommend using it as your go to resource for Python development.

Another resource I would recommend for fixing errors is Stack Overflow

<https://www.w3schools.com/python/>

<https://stackoverflow.com/>

Lists

There are many ways to interact with lists in Python. For more detailed info go to:

https://www.w3schools.com/python/python_lists.asp

```
thon > 9_22 > Concepts > 14_lists.py > ...
1  #!/usr/bin/env python3
2
3  # Create a list with different data types
4  myList = ["Item 1", 2, 3.5, True]
5
6  # For every item in the list state it's type
7  for item in myList:
8      print(type(item))
9
10 print(myList)
11 # For every item in the list change it's value to index number
12 for index in range(len(myList)):
13     myList[index] = index
14
15 # Insert a string in to the middle of the list
16 myList.insert(2,"I'm not supposed to be here")
17 print(myList)
18
19 # Append another string to the list
20 myList.append("I'm at the end!")
21 print(myList)
22
23 myList.remove(2) # Removes the 2 entry in the list
24 print(myList)
25
26 myList.pop(2) # Removes the 3rd entry in the list
27 myList.pop(-1) # Removes the last entry in the list
28 print(myList)
29
30 myList.sort(reverse=True) # Sorts list numerically backwards
31 print(myList)
```

Dictionaries

Dictionaries are used to store data values in
key:value pairs.

```
thon > 9_22 > Concepts > 15_dictionaries.py > ...
1  #!/usr/bin/env python3
2
3  # Create a dictionary
4  myCar = {
5      "make": "Honda",
6      "model": "Accord",
7      "year": 2010
8  }
9  print(myCar)
10
11 # Print the amount of entries in myCar
12 print(len(myCar))
13
14 # Add the color of myCar
15 myCar["color"] = "white"
16 print(myCar)
17
18 # Change the year of myCar
19 myCar["year"] = 2020
20
21 # Print the year of myCar
22 print(myCar.get("year"))
23
24 # Remove the make data of myCar
25 myCar.pop("make")
26 print(myCar)
27
28 # List every entry in myCar
29 for data in myCar:
30     print(data) # Key
31     print(myCar[data]) # Value
```

Project 3!

Project time! This is going to combine most of the concepts we have learned so far.

Your objective is to make a program that will delete items off your grocery list when the user enters them:

["Apples", "Oranges", "Bananas"]

What item should I remove? Apples

["Oranges", "Bananas"]

REMINDER: You can always reuse code from other projects to speed up dev

Challenge 1: Have line spacing after every result

Challenge 2: Tell the user if what they entered isn't in the list

Challenge 3: make your program accept "Apples" but also 0 as an option

```
1  #!/usr/bin/env python3
2
3  groceryList = ["Apples", "Oranges", "Bananas"]
4
5  while len(groceryList):
6      print(groceryList)
7      userInput = input("What item should I remove? ")
8      try:
9          userInput = int(userInput)
10     except:
11         if userInput in groceryList:
12             groceryList.remove(userInput)
13         else:
14             print("That's not on our list!")
15     else:
16         if userInput < len(groceryList):
17             groceryList.pop(userInput)
18         else:
19             print("That's out of range of the list!")
20     finally:
21         print("\n")
```

String Manipulation

Being able to manipulate strings allows you to give the user a custom tailored experience and improve their experience.

```
> 9_22 > Concepts > 16_strings.py > ...
1  #!/usr/bin/env python3
2
3  a = """This
4  Is a
5  Multiline
6  String      """
7  print(a) # Print whole string
8  print(a[3]) # Print the 4th letter
9  print(len(a)) # Print length of string
10 print("This" in a) # True
11 print("This" not in a) # False
12 print(a[0:4]) # Print 1st-5th letters
13 print(a[-6:]) # Print last 6 letters
14 print(a.upper()) # Print all uppercase
15 print(a.replace("I","")) # Remove all I's
16 # Print all uppercase and remove all I's
17 print(a.upper().replace("I",""))
18 # Split the string into a list on every newline
19 print(a.split("\n"))
20 # Removes that whitespace at the end before split
21 print(a.strip().split("\n"))
22 print(a + "\nNew Data!")
23 print("You're \"Smart\" ") # Escape quotes
24 fillInTheBlank = "I am {} and {}"
25 qualities = ["Old","Sad"]
26 print(fillInTheBlank.format(qualities[0],qualities[1]))
```


File Management

Being able to access and manipulate files is very important when it comes to developing games.

This will most of the time be used to save and persist user settings in a file like a settings.json

```
1  #!/usr/bin/env python3
2
3  def readFile(file):
4      f = open(file,"rt")
5      print(f.read())
6      f.close()
7
8  # Open file in read only mode as a text
9  f = open("read.txt", "rt")
10 # Read only the first 2 lines
11 print(f.readline())
12 print(f.readline())
13
14 print(f.read()) # Read rest of file
15 f.close() # Close files when done
16
17 f = open("write.txt","wt") # Overwrite
18 f.write("This overwrote everything!")
19 f.close()
20
21 readFile("write.txt")
22
23 f = open("write.txt","at") # Append
24 f.write("\nSneaky extra line\nLine 3")
25 f.close()
26
27 readFile("write.txt")
28
29 # The more proper way to open a file for use
30 with open("read.txt", "rt") as f:
31     print(f.read())
32
33 with open("write.txt", "at") as f:
34     f.write("\nLine 4")
35
36 readFile("write.txt")
```


Project 4!

Project time!

Your objective is to program a function that will edit a user's config file when they respond with 0, 1, or 2

Fullscreen: False

Monsters: False

Permadeath: False

What setting should I change? 1

Fullscreen: False

Monsters: True

Permadeath: False

```
1  #!/usr/bin/env python3
2
3  def readConfig():
4      with open("settings.txt", "r") as f:
5          return f.read()
6
7  def writeConfig(string):
8      with open("settings.txt", "w") as f:
9          f.write(string)
10
11 def toggle(index, config):
12     if "True" in config[index]:
13         config[index] = config[index].replace("True","False")
14     else:
15         config[index] = config[index].replace("False", "True")
16     config = config[0] + "\n" + config[1] + "\n" + config[2]
17     writeConfig(config)
18
19 while True:
20     config = readConfig()
21     print(config)
22     config = config.split("\n")
23     try:
24         userInput = int(input("What setting should I change? "))
25     except:
26         print("That's not an integer!")
27     else:
28         if userInput < 3:
29             toggle(userInput, config)
30         else:
31             print("That integer is too large!")
```

My Solution