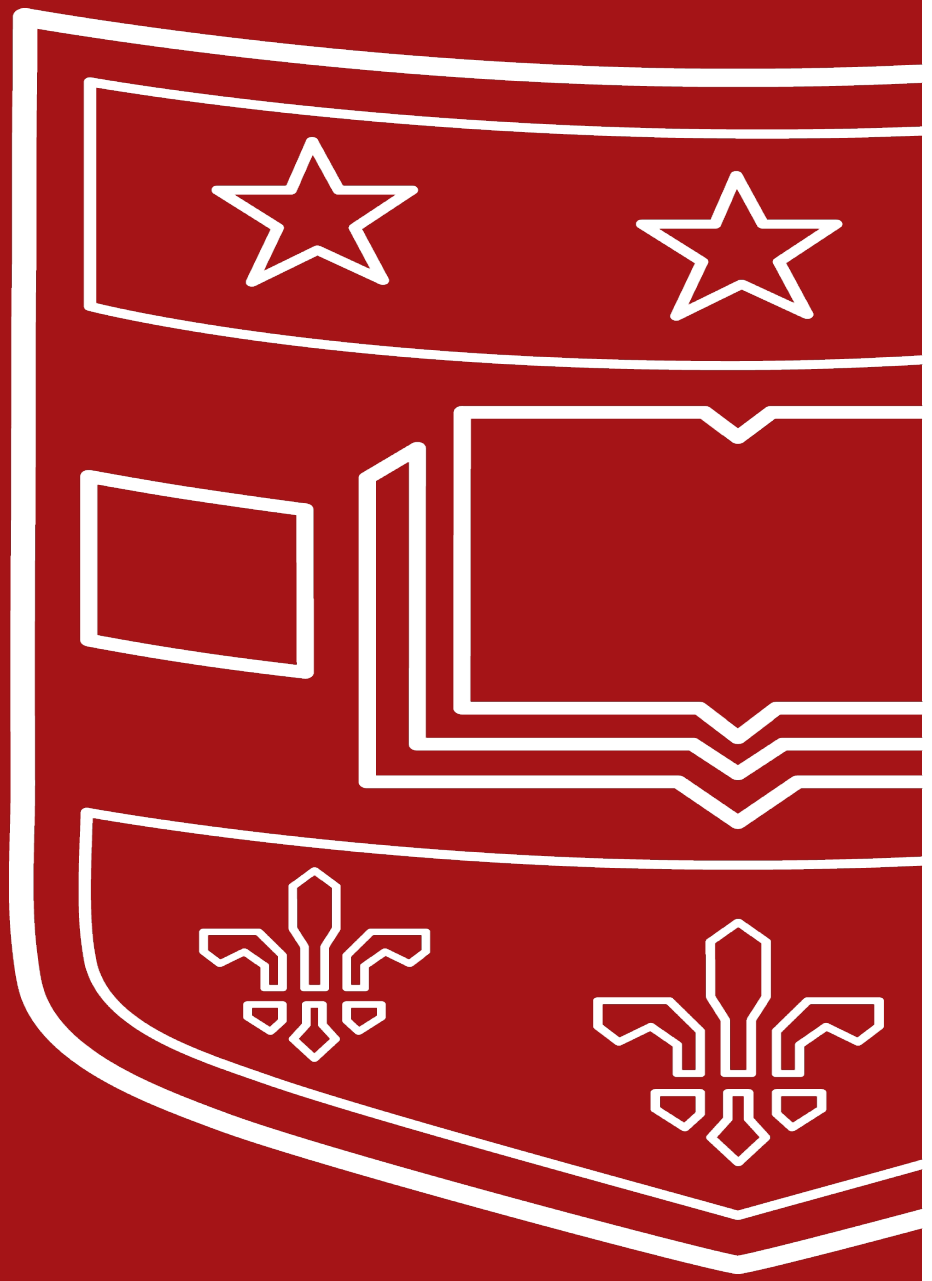
Introduction to Python
Session 3

TRIADS Training Series, Spring 2023

Instructor: Claudia Carroll



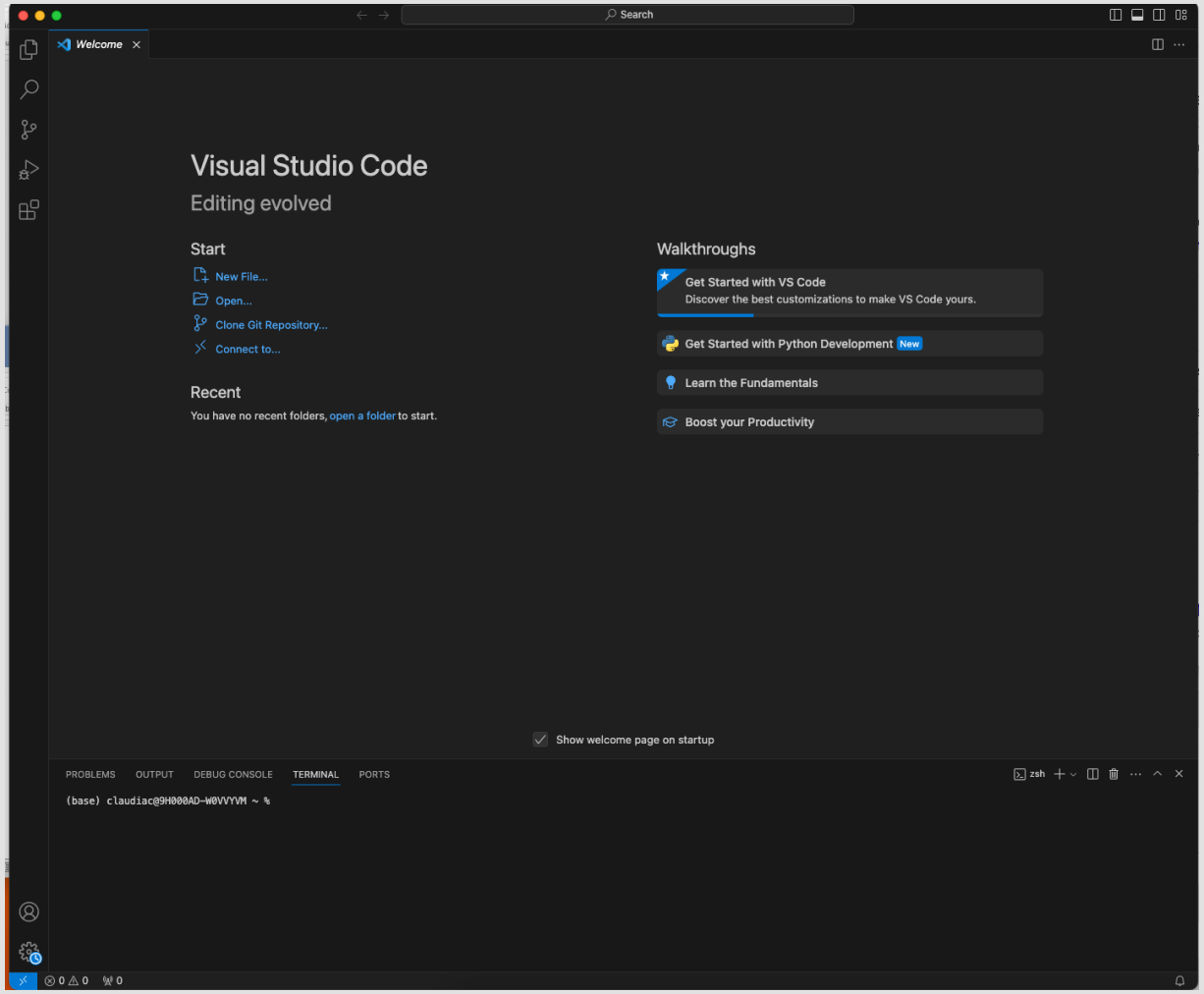
Washington University in St. Louis

Today's Lesson

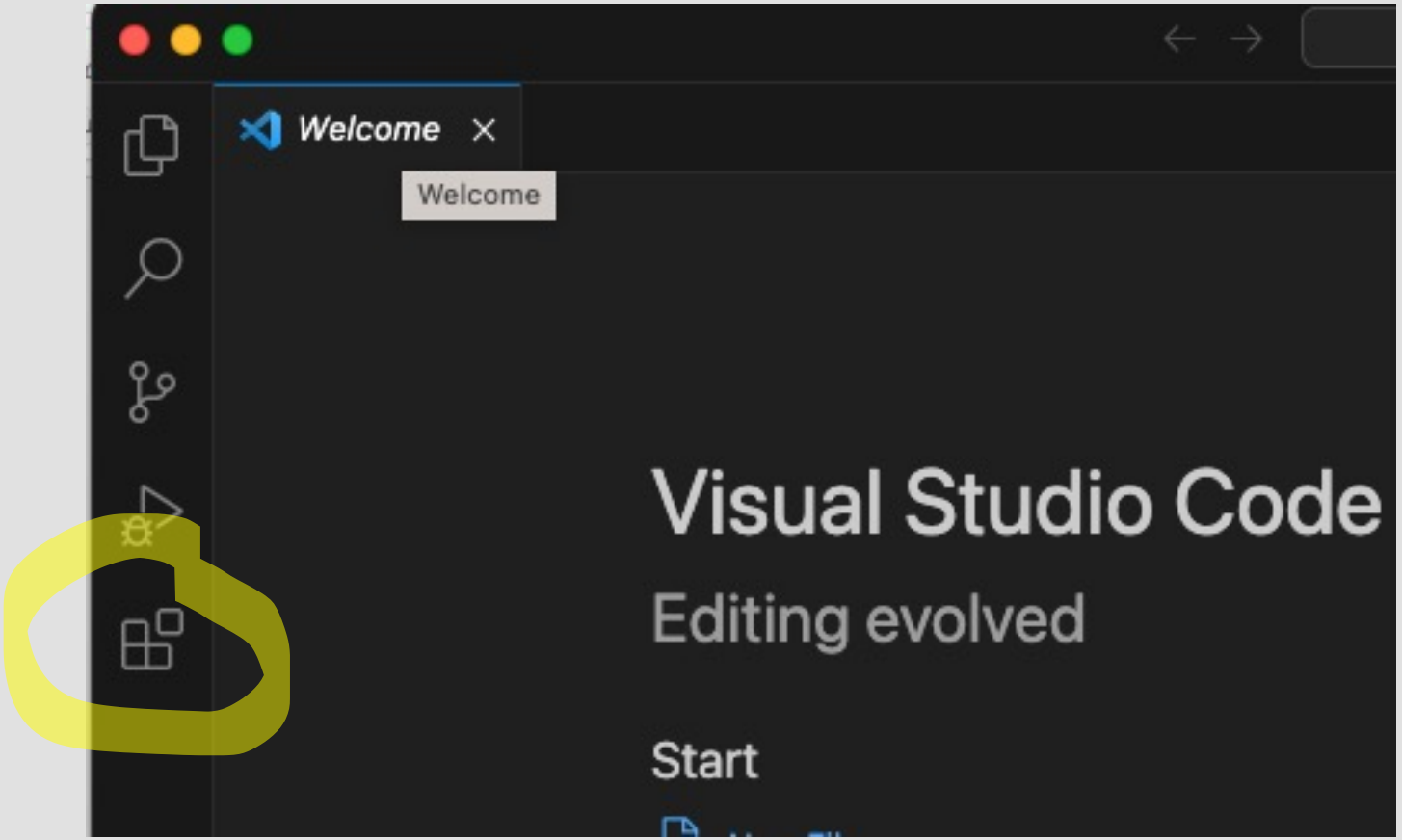
1. VS Code set up
2. Exercise Solutions
3. Comparisons and Conditionals
4. Loops

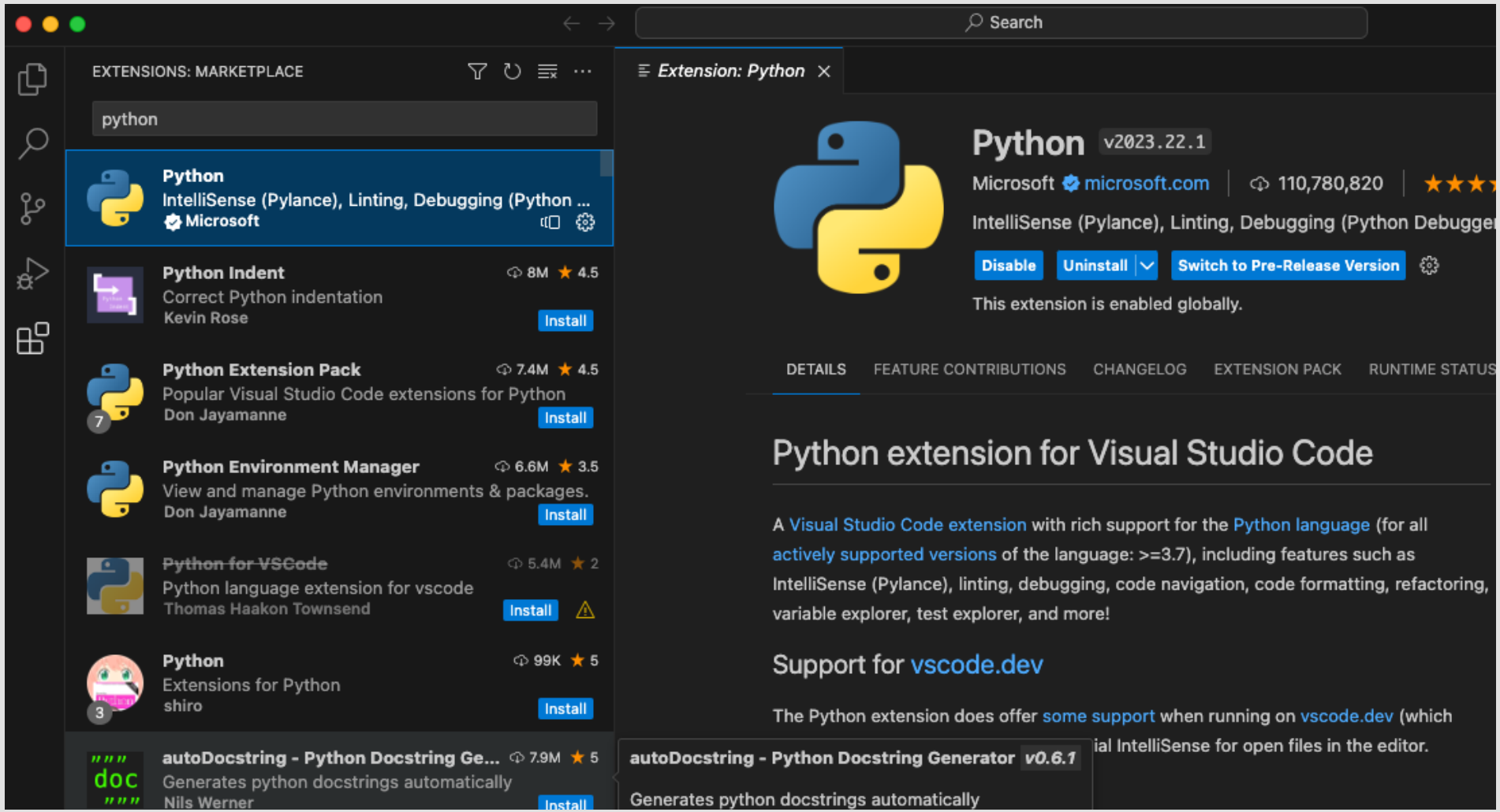
https://github.com/ClaudiaECarroll/Intro_to_Python

VS Code Set-Up



Extension Installer




Search
EXTENSIONS: MARKETPLACE
python
Python
IntelliSense (Pylance), Linting, Debugging (Python ...
Microsoft
Python Indent
8M ★ 4.5
Correct Python indentation
Kevin Rose
Install
Python Extension Pack
7.4M ★ 4.5
Popular Visual Studio Code extensions for Python
Don Jayamanne
Install
Python Environment Manager
6.6M ★ 3.5
View and manage Python environments & packages.
Don Jayamanne
Install
Python for VSCode
5.4M ★ 2
Python language extension for vscode
Thomas Haakon Townsend
Install
Python
99K ★ 5
Extensions for Python
shiro
Install
autoDocstring - Python Docstring Ge...
7.9M ★ 5
Generates python docstrings automatically
Nils Werner
Install
Extension: Python
Python v2023.22.1
Microsoft microsoft.com
110,780,820
IntelliSense (Pylance), Linting, Debugging (Python Debugger
Disable
Uninstall
Switch to Pre-Release Version
This extension is enabled globally.
DETAILS
FEATURE CONTRIBUTIONS
CHANGELOG
EXTENSION PACK
RUNTIME STATUS
Python extension for Visual Studio Code
A Visual Studio Code extension with rich support for the Python language (for all actively supported versions of the language: >=3.7), including features such as IntelliSense (Pylance), linting, debugging, code navigation, code formatting, refactoring, variable explorer, test explorer, and more!
Support for vscode.dev
The Python extension does offer some support when running on vscode.dev (which
ial IntelliSense for open files in the editor.
autoDocstring - Python Docstring Generator v0.6.1
Generates python docstrings automatically

Exercise 2

Create the following list to track your groceries and prices:

1. Groceries = [“apples”, “4”, “milk”, “5.9”, “bread”, “3”, “wine”, “15.5”]
2. Write the code to list out only the food items, followed by the number of food items (do not just manually count them!)
3. Write the code to extract the prices from the list, calculate the total, then output the following statement: *The total cost of the groceries is $X*

Hint: Watch your parentheses!

Solution

Groceries = [“apples”, “4”, “milk”, “5.9”, “bread”, “3”, “wine”, “15.5”]

Question:
Write the code to list out only the food items, followed by the number of food items (do not just manually count them!)

Solution:

```
>>>print(groceries[0], groceries[2], groceries[4], groceries[6], (len(groceries)/2))
```

apples milk bread wine 4.0

groceries = [“apples”, “4”, “milk”, “5.9”, “bread”, “3”, “wine”, “15.5”]

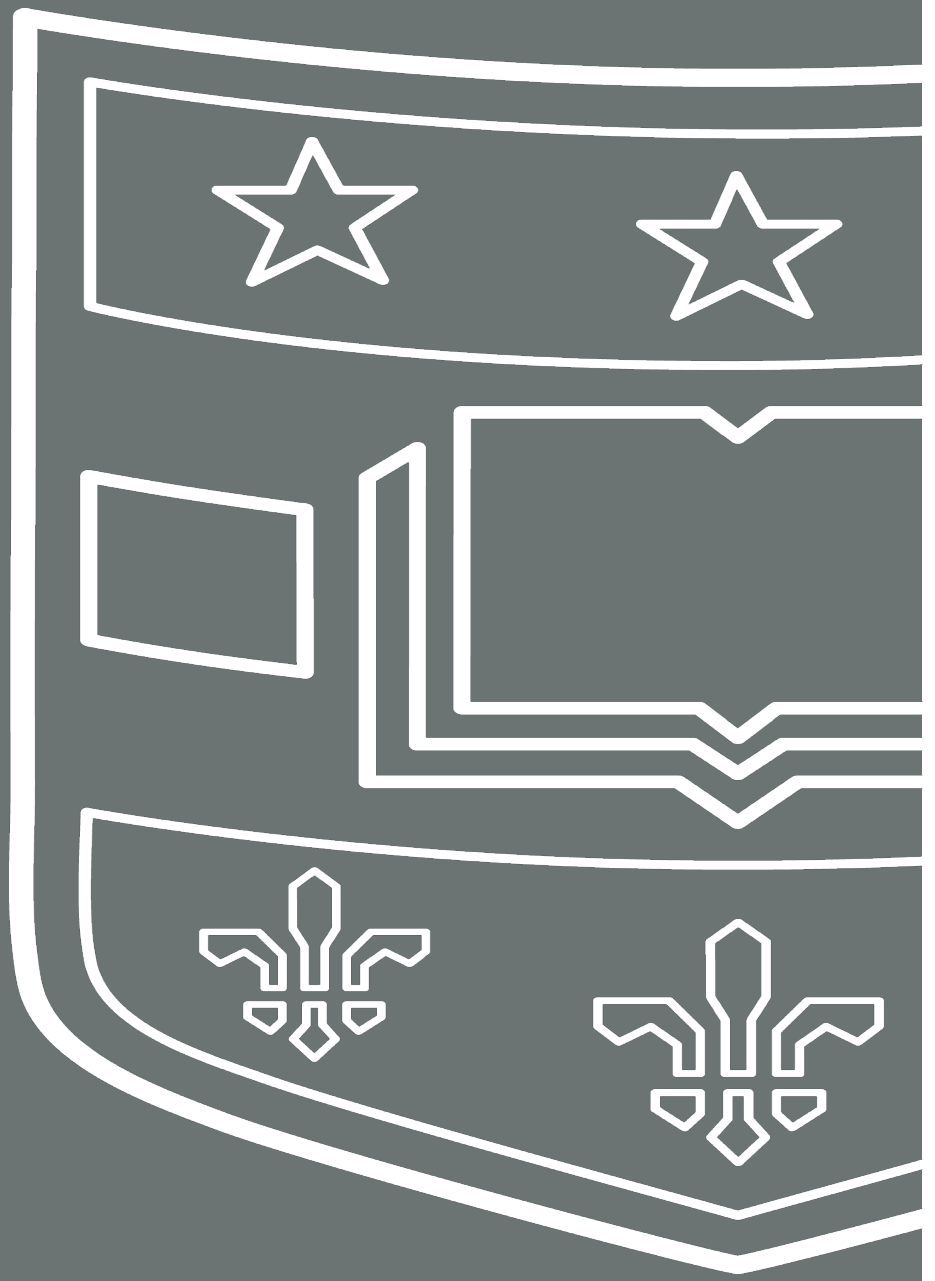
Question:
Write the code to extract the prices from the list, calculate the total, then output the following statement: The total cost of the groceries is $X

Solution:

```
print("The total cost of the groceries is $", int(groceries[1])
+ float(groceries[3]) + int( groceries[5]) + float(groceries[7]))
```

The total cost of the groceries is $ 28.4

Comparisons and Conditionals



Washington University in St. Louis

Boolean Statements

- A **Boolean statement** is a statement that is either True or False
- Boolean statements are primarily used to filter data or methods based on certain conditions
- Boolean statements are usually produced using **Boolean operators**

```
>>> age = 15
>>> print(age < 12)
False
>>> print (age > 12)
True
```

Python Comparison Operators

==	Equal to
!=	Not equal to
>	Greater than
<	Less than
>=	Great than or equal to
<=	Less than or equal to

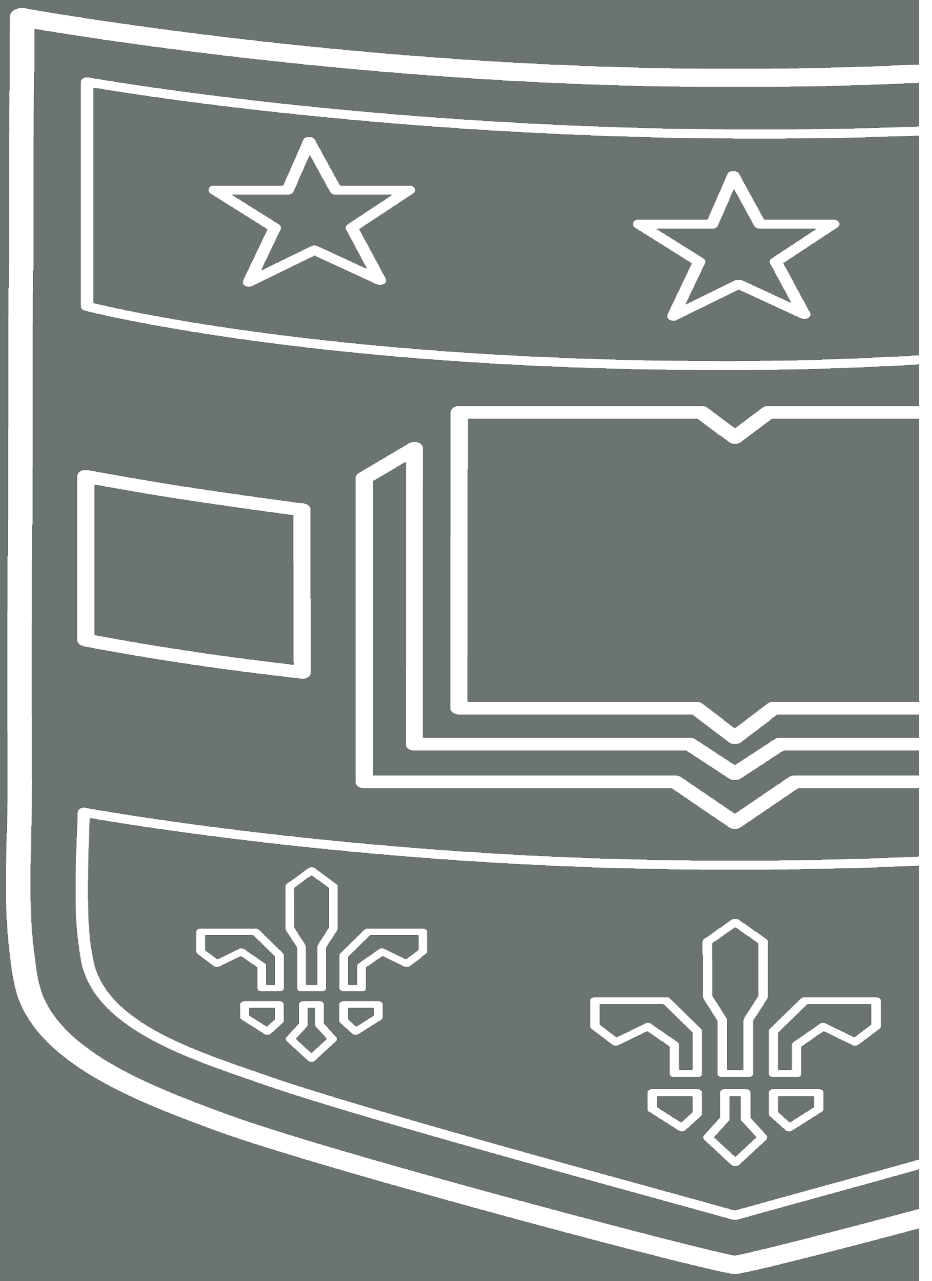
Conditionals

```
number = 0

if number > 0:
   print('Positive number’)
elif number <0:
   print('Negative number’)
else:
   print('Zero') print('This statement is always
executed')
```

Demo 1

Conditionals



Washington University in St. Louis

Exercise 1

Create three lists:

humanities = ["English", "History", "French", "Art History", "Philosophy"]
sciences = ["Biology", "Chemistry", "Neuroscience", "Physics", "Ecology"]
social_sciences = ["Political Science", "Sociology", "Anthropology", "Psychology"]

1. Using conditionals, write the code that prints “You are a philosopher” if the fifth element of humanities is philosophy (hint: remember indices from Monday!)
2. Now write a program that asks what department you are in, and based on the response, outputs one of the following messages (hint: remember the input() function from Monday!):

You are a humanist!
You are a scientist!
You are a social scientist!
You are probably doing something interesting!

Solution Part 1

```
if humanities[4] == "Philosophy":
   print("You are a philosopher")
```

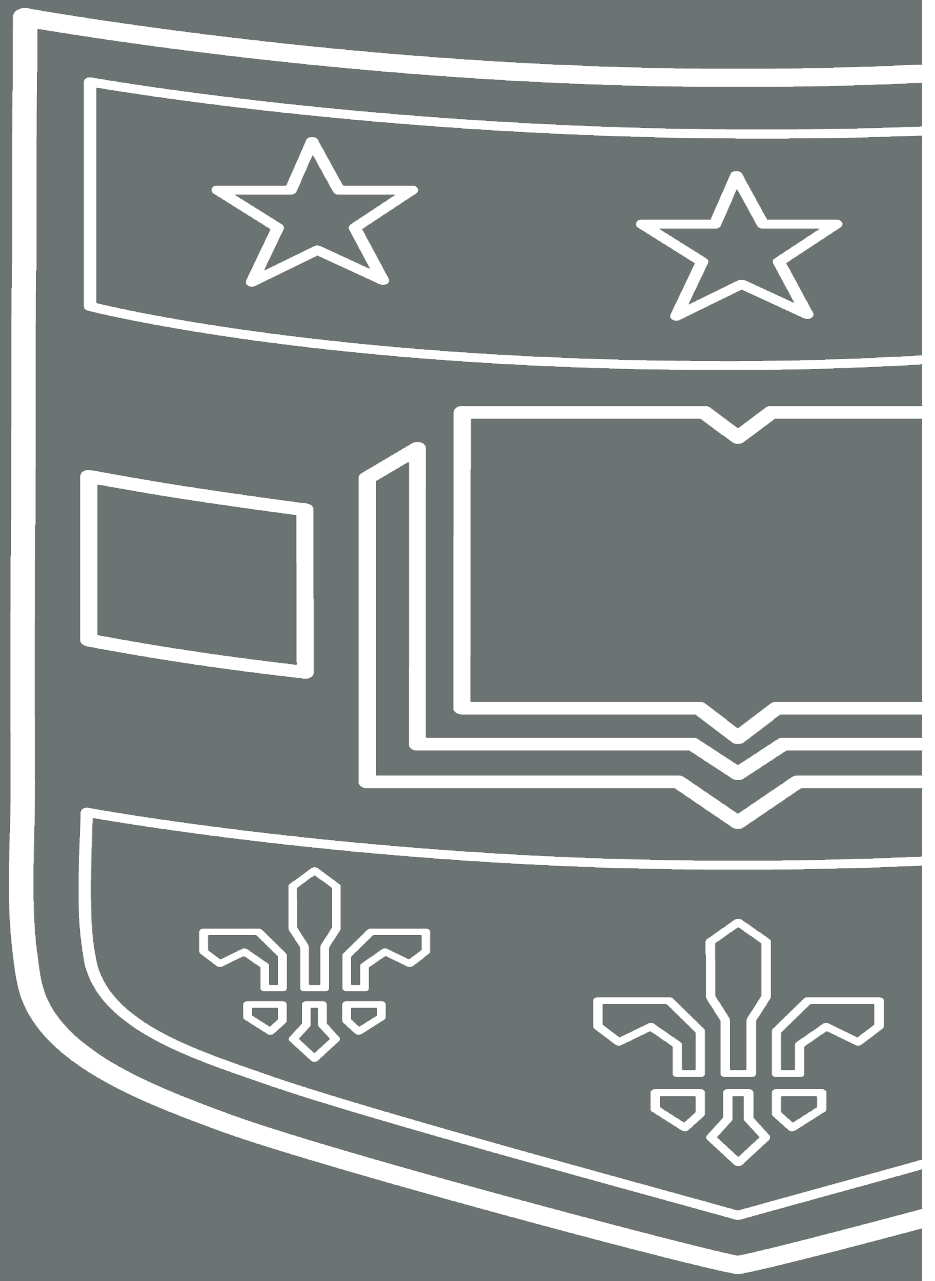
Solution Part 2

```
your_department = input("What department do you work in? ")

if your_department in humanities:
   print("You are a humanist")
elif your_department in sciences:
   print("You are a scientist")
elif your_department in social_sciences:
   print("You are a social scientist")
else:
   print("You are probably doing something interesting!")
```

Demo 2

For loops!



Washington University in St. Louis

Solution Part 1

Groceries = [“apples”, “4”, “milk”, “5.9”, “bread”, “3”, “wine”, “15.5”]

Question:
Write the code to list out only the food items, followed by the number of food items (do not just manually count them!)

Solution:

```
>>>print(groceries[0], groceries[2], groceries[4], groceries[6], (len(groceries)/2))
```

apples milk bread wine 4.0

Solution using for loop

```
groceries = ["apples", "4", "milk", "5.9", "bread", "3", "wine", "15.5"]

for i in range(len(groceries)):
   if i%2 == 0:
      print(groceries[i])
print(len(groceries)/2)
```

Basic For Loop

Form	for variable in collection: #do something with variable
Example code	odds = [1, 3, 5, 7] for num in odds: print(num)
Example output	1 3 5 7

Exercise 2

Using your store_shelf list:

1. Write a for loop to print all the items in the list starting with the letter “a”

2. Write a for loop to print all the items in your list that are longer than 5 letters

Solution Part 1

```
store_shelf = ["apples", "bread", "cookies", "avocados",
"eggs", "milk", "carrots"]


for x in store_shelf:
    if x[0] == "a":
        print(x)
```

Solution Part 2

```
store_shelf = ["apples", "bread", "cookies", "avocados",
"eggs", "milk", "carrots"]


for x in store_shelf:
    if len(x) > 5:
        print(x)
```

Solution Part 3

```
humanities = ["English", "History", "French", "Art History",
"Philosophy"]
sciences = ["Biology", "Chemistry", "Neuroscience", "Physics",
"Ecology"]


for x in humanities:
   for y in sciences:
      if len(x) == len(y):
         print(x, y, "These words have the same number of letters!")
```

Homework

1. Finish any in-class exercises

2. Complete the Class 3 Homework Exercises

https://github.com/ClaudiaECarroll/Intro_to_Python