

Data Science, lesson 3

Python Foundations

What did we cover last time?

1. Command line
2. Git and GitHub



Learning objectives

Python Data Types

1. Discuss Python as a programming language.
2. Define integers, strings, tuples, lists, and dictionaries.
3. Demonstrate arithmetic operations and string operations.
4. Demonstrate variable assignment.

Python Iteration, Control Flows, and Functions

1. Define Python control flow and conditional programming.
2. Implement `for` and `while` loops to iterate through data structures.
3. Apply `if`, `else` conditional statements.
4. Create functions to perform repetitive actions.
5. Demonstrate error handling using `try`, `except` statements.
6. Use Python control flow and functions to help us parse, clean, edit and analyze data sets.



Data Science



Python Data Types

About Python

Python was created by Guido van Rossum and released back in 1991.

Since then, Python has greatly grown as a high-level, general-purpose programming language with a huge open-source community supporting it.

The Zen of Python

Beautiful is better than ugly.

Explicit is better than implicit.

Simple is better than complex.

Complex is better than complicated.

Flat is better than nested.

Sparse is better than dense.

Readability counts.

Special cases aren't special enough to break the rules.

Although practicality beats purity.

Errors should never pass silently.

Unless explicitly silenced.

In the face of ambiguity, refuse the temptation to guess.

There should be one — and preferably only one — obvious way to do it. Although that way may not be obvious at first unless you're Dutch. Now is better than never.

Although never is often better than right now.

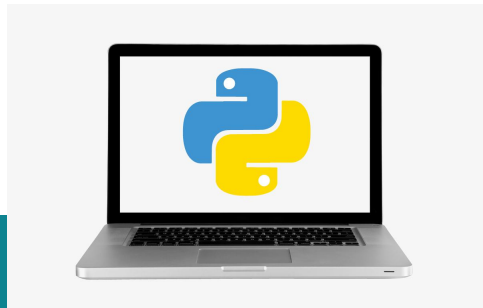
If the implementation is hard to explain, it's a bad idea.

If the implementation is easy to explain, it may be a good idea.

Namespaces are one honking great idea — let's do more of those.

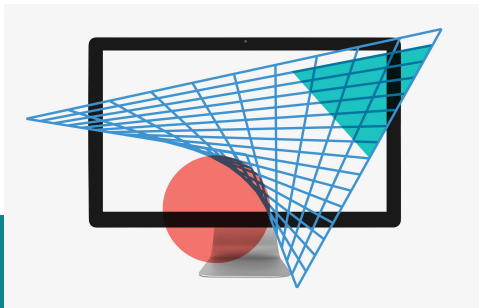
Why Use Python for Data Science?

General Purpose



Python was not intended to just be used for software or website development. Instead, it comes with the basic building blocks you need to develop anything you want with it.

Open Source



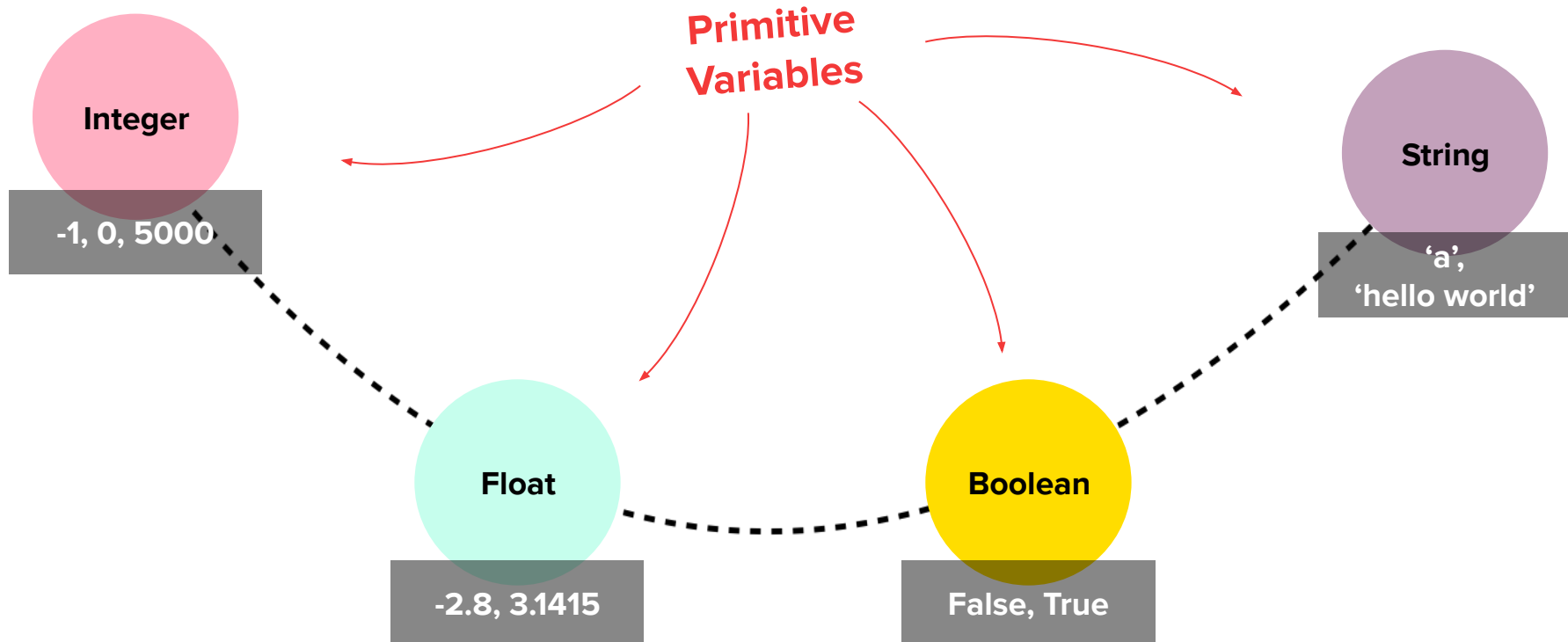
A large open-source community has already created hundreds of libraries containing combinations of the foundation blocks to create more specific tool sets.

Readability



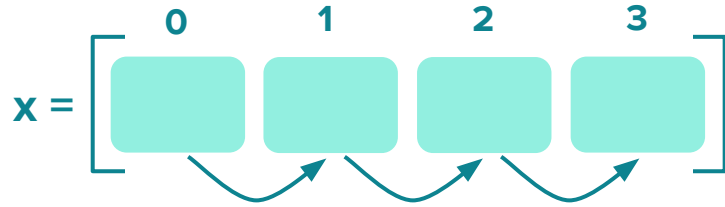
When a foreign language is similar to your native language, it's much easier to pick up. The same can be said for Python, whose general flow makes it a lot easier for humans to read and interpret code.

Python Data Types

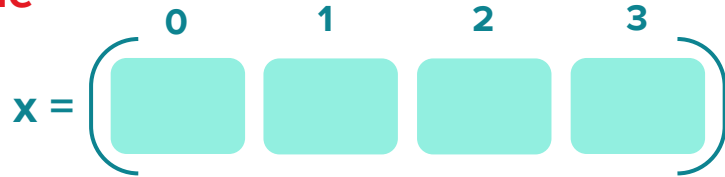


Complex Variables

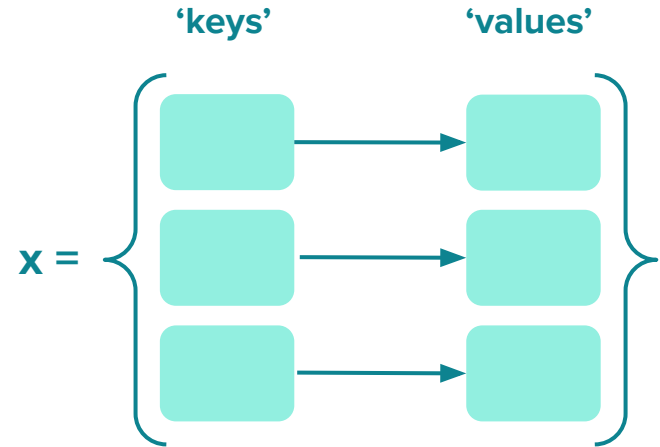
List



Tuple



Dictionary



What is the type of `x = '5.8'`?

- A . Boolean
- B. Integer
- C. String
- D. Float

**What will be the output of
`list(range(6, 12, 2))`?**

A. [6, 8, 10, 12]

B. [7, 9, 11]

C. [6, 8, 10]

Data Science



Control Flow

In Python, indentation matters!

- A condition in the line preceding the indented block determines whether the indented code is run or skipped.
- The line before every indented block must end with a colon ':'

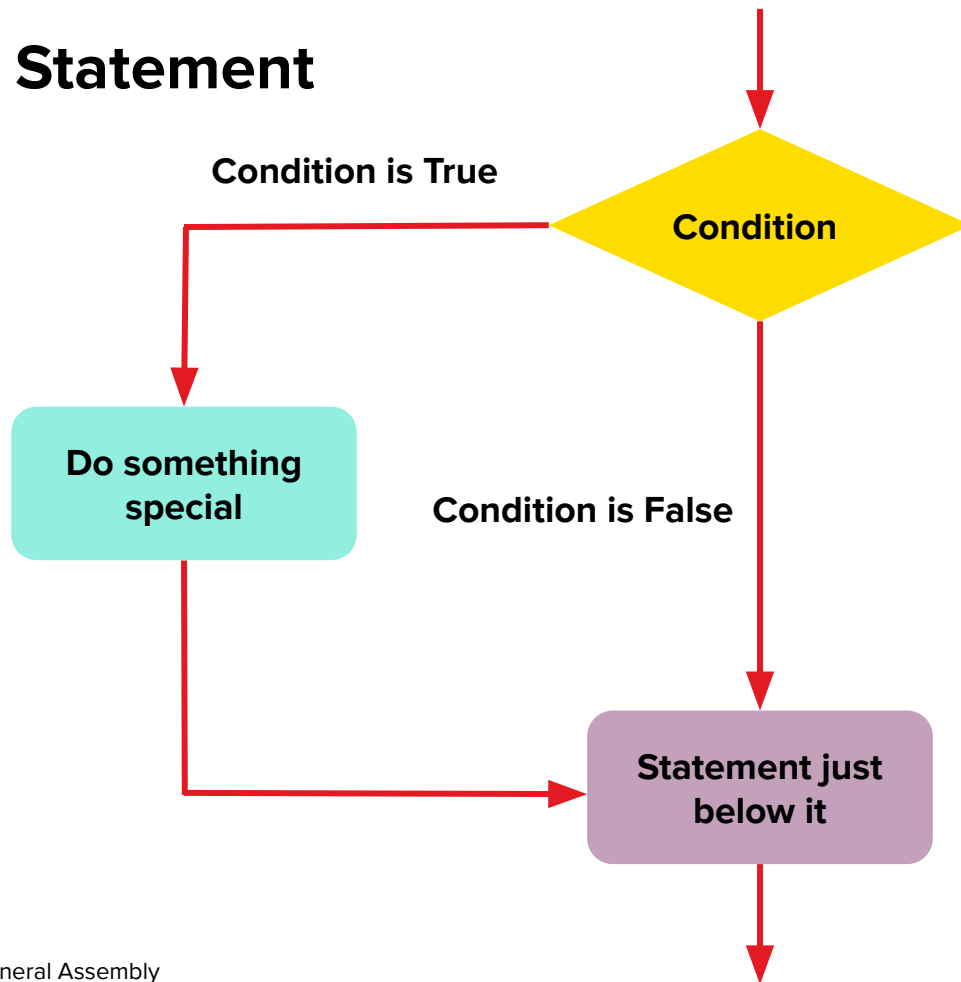
**To control the
flow**



**Follow the
rules**



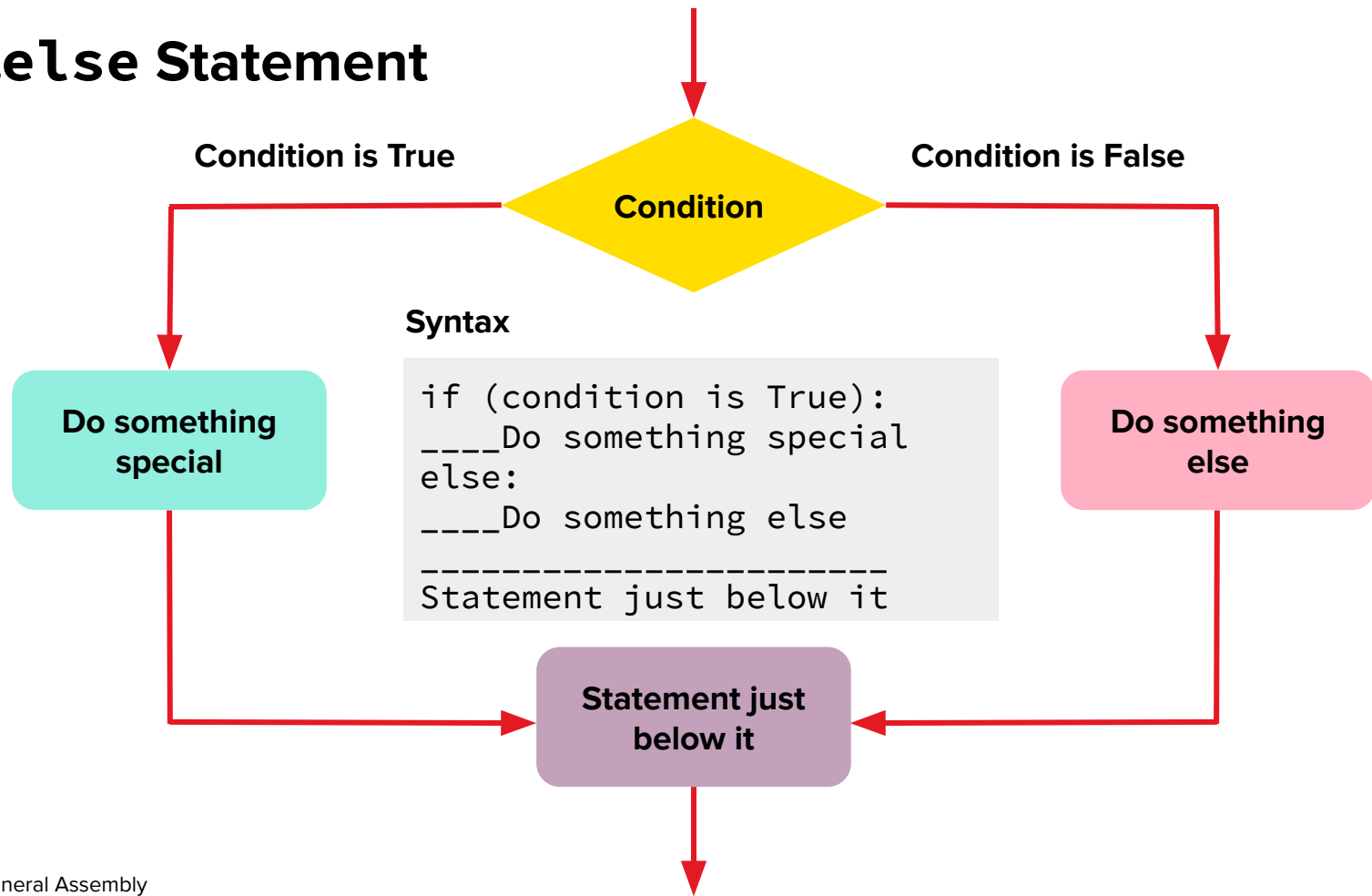
if Statement



Syntax

```
if (condition is True):  
    ____Do something special  
    -----  
    Statement just below it
```

if...else Statement



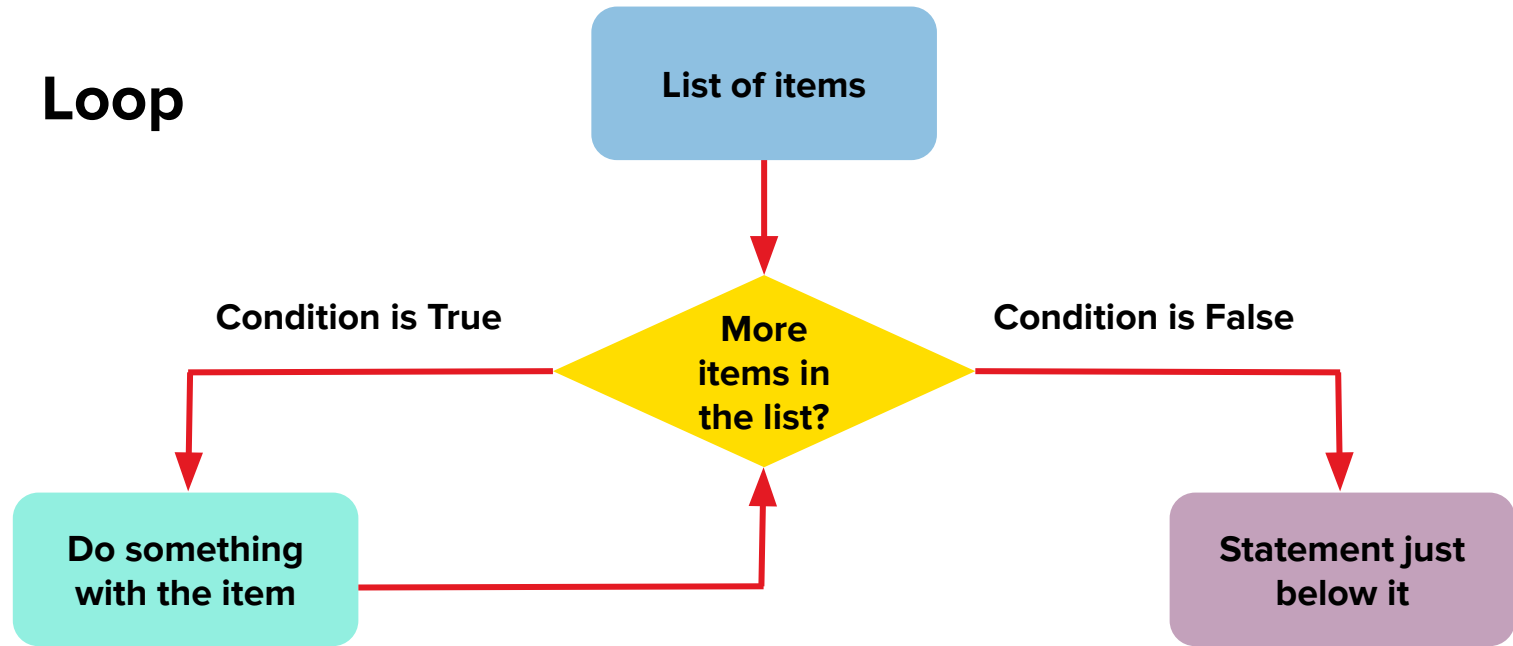
if...elif...else Statement

Syntax

```
if (condition 1 is True):  
    ____Do something special  
elif (condition 2 is True):  
    ____Do some other thing  
else:  
    ____Do something else  
-----  
Statement just below it
```

Can you draw it?

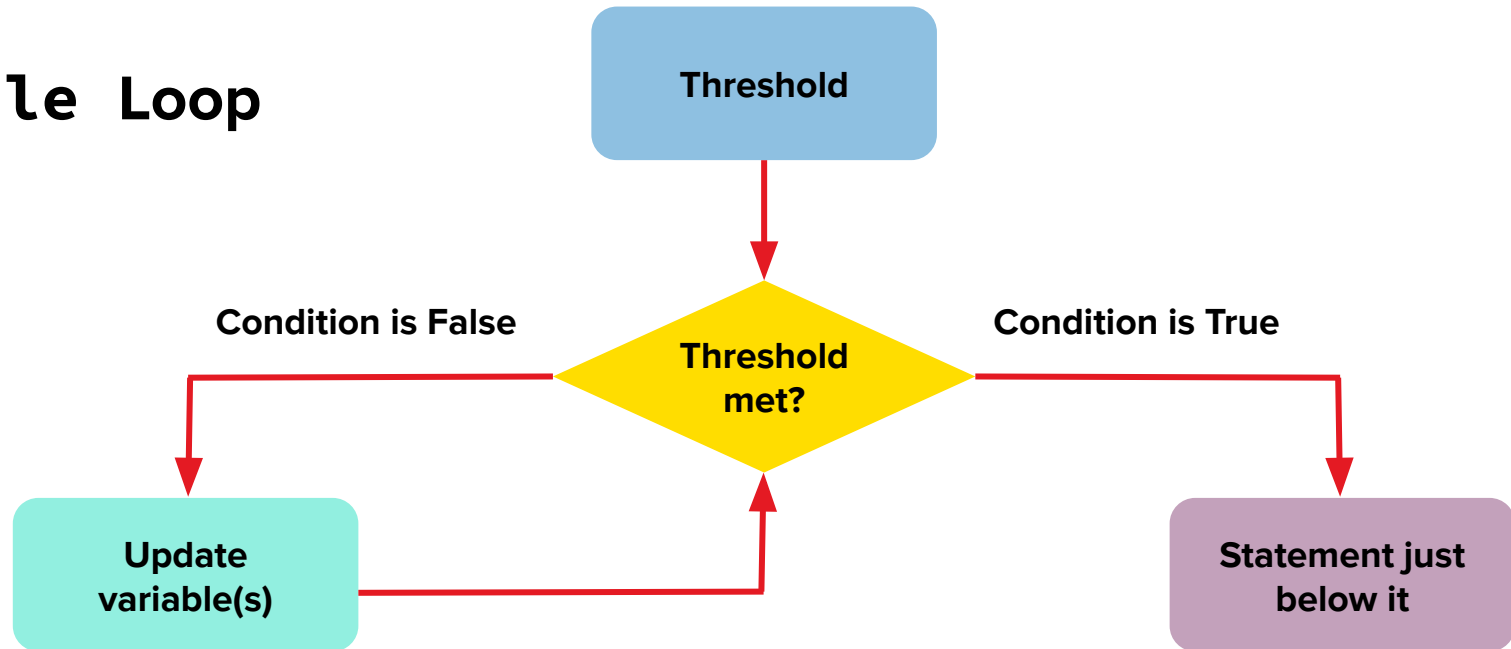
for Loop



Syntax

```
for (item in list):  
    ____Do something with the item  
    -----  
    Statement just below it
```


while Loop



Syntax

```
while (threshold is not met):  
    ____Update variable(s)  
    -----  
    Statement just below it
```

Data Science



Conclusion



Discussion: Important Takeaways

5 minutes



Let's Debrief

- What did you learn in this lesson?
- What was challenging about this topic?
- What questions do you still have?
- What are some next steps you could take with this information?



Ask Me Anything!



Before Next Class:

- A. Get into habit of using Git and GitHub**
- B. Practice your Python!**

See you next time!



Thank you!

