

# Why Python?

## Introduction

Making Friends with Python – DI Team (22-23 May, 2017)

mrchakra@microsoft .com



Mrinal Chakraborty  
Data Insights Team  
Microsoft India

- ✓ **Programing:** SAS, R-Server, TensorFlow and Scala
- ✓ **Big-Data:** Cloudera Hadoop certification and Spark Ecosystem
- ✓ **Machine learning:** Logistic Regression, Neural Networks, Support vector machines, XGBoost, Classification and Association rules
- ✓ **Allied Analytics skills:** Visualisation, Marketing & Web analytics
- ✓ **Certifications:** PMP, Certified Scrum Master & Certified in Business analytics from **Indian School of Business** <http://www.isb.edu/cba/>

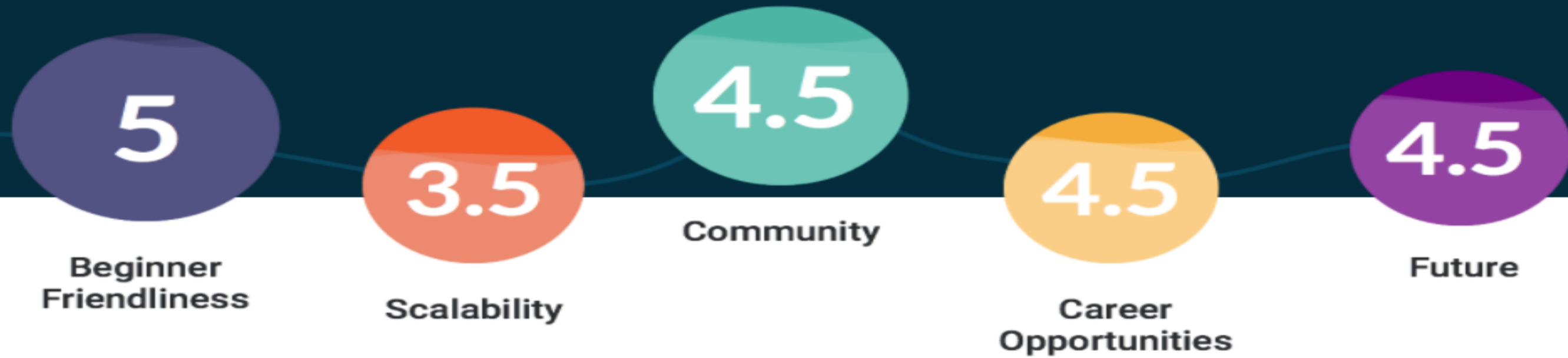
# Why Learn Python ?

Surendra's & Manmeet's Motivation

Your Reasons

My Thoughts ...

# Why Learn Python ?



5th Largest StackOverflow Community

4th Most-Used Language at GitHub

3rd Largest Meetup Community

Popular sites built with Python



Java programs end up being longer and more painful to write than the equivalent Python programs


```
name = raw_input('Enter file:')
handle = open(name, 'r')
text = handle.read()
words = text.split()


counts = dict()
for word in words:
    counts[word] = counts.get(word,0) + 1
bigcount = None
bigword = None

for word,count in counts.items():
    if bigcount is None or count >
bigcount:
        bigword = word
        bigcount = count
print bigword, bigcount
```

python words.py  
Enter file: words.txt  
to 16

python words.py  
Enter file: clown.txt  
the 7







[All](#)
[Images](#)
[News](#)
[Videos](#)
[More](#)
[Settings](#)
[Tools](#)

About 10,10,000 results (0.76 seconds)


Shop for raspberry pi in india price on Google




PICAMERA  
Raspberry Pi ...  
₹ 2,332  
Amazon India




Raspberry Pi  
5MP Camera ...  
₹ 1,625  
Amazon India



Raspberry Pi 3  
Model B 1.2 GHz  
₹ 3,000  
RS Components



Quad Store(TM) -  
Ultra kit for ...  
₹ 2,999  
Amazon India




Protocentral  
Raspberry Pi 3 ...  
₹ 5,995  
Amazon India

[Raspberry Pi 3 Model B Kit - Computer Components Online - amazon.in](#)  
[www.amazon.in/computers](#)  
 Buy Processors, Graphic Cards, Motherboards & More. Easy Returns. Pay COD\*.  
 Free Shipping\* · Great Value for Money · 100% Purchase Protection · Huge Selection · Low Prices  
 Types: Adapters & Cables, Memory Card, External Hard Drives, Pen Drives, Keyboard & Mouse Sets, ...  
[Hard Disks & Pen Drives](#)   [Student Offers - Laptops](#)  
[Used Laptops](#)   [Tablets](#)

Raspberry Pi 2 Model B 1GB -The Complete Kit

Plan (months)	EMI (payable to provider)	Total Cost (payable to provider)
3	1,648.07	No Cost
6	837.35	No Cost
9	567.21	No Cost
12	432.21	No Cost



[Raspberry Pi 2 Model B 1GB -The Complete Kit: Amazon.in: Amazon.in](#)  
[www.amazon.in/Raspberry-Pi-Model-1GB-Complete/dp/B00T2U7R7I](#)

YES !

You could start coding in Python with just

~ Rs. 1,625

of upfront hardware cost!

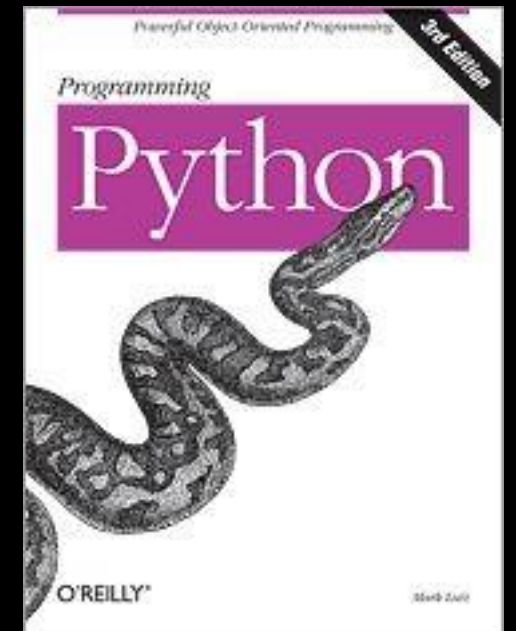


An interpreted **language**, **Python** has a design philosophy which emphasizes code readability (notably using whitespace indentation to delimit code blocks rather than curly braces or keywords), and a syntax which allows programmers to express concepts in fewer lines of code than possible in **languages** such as C++ or Java.



[www.python.org](http://www.python.org)

[Python \(programming language\) - Wikipedia](https://en.wikipedia.org/wiki/Python_(programming_language))  
[https://en.wikipedia.org/wiki/Python\\_\(programming\\_language\)](https://en.wikipedia.org/wiki/Python_(programming_language))



**Guido van Rossum** is the author of the Python programming language. In the Python community, Van Rossum is known as a "Benevolent Dictator For Life" (BDFL), meaning that he continues to oversee the Python development process, making decisions where necessary.



# Talking to Python

UbuntuMachine on MRINAL-PC - Virtual Machine Connection

File Action Media Clipboard View Help

Terminal Terminal File Edit View Search Terminal Help

```
mrinal@UbuntuSys: ~  
mrinal@UbuntuSys:~$ python3.5  
Python 3.5.2 (default, Nov 17 2016, 17:05:23)  
[GCC 5.4.0 20160609] on linux  
Type "help", "copyright", "credits" or "license" for more information.  
>>> quit()  
mrinal@UbuntuSys:~$ python  
Python 3.6.0 |Anaconda 4.3.1 (64-bit)| (default, Dec 23 2016, 12:22:00)  
[GCC 4.4.7 20120313 (Red Hat 4.4.7-1)] on linux  
Type "help", "copyright", "credits" or "license" for more information.  
>>> quit()  
mrinal@UbuntuSys:~$ Python2.7  
Python2.7: command not found  
mrinal@UbuntuSys:~$ python2.7  
Python 2.7.12 (default, Nov 19 2016, 06:48:10)  
[GCC 5.4.0 20160609] on linux2  
Type "help", "copyright", "credits" or "license" for more information.  
>>>
```

You can call-out different version of Python...



```
mrinal@UbuntuSys: ~  
mrinal@UbuntuSys:~$ python  
Python 3.6.0 |Anaconda 4.3.1 (64-bit)| (default, Dec 23 2016, 12:22:00)  
[GCC 4.4.7 20120311 (Red Hat 4.4.7-1)] on linux  
Type "help", "copyright", "credits" or "license" for more information.  
>>> Test = 1+2  
>>> Test  
3  
>>>
```

This is a good test to make sure that you have Python correctly installed. Note that `quit()` also works to end the interactive session.

# What Do We Say?

# Elements of Python

- **Vocabulary / Words** - Variables and Reserved words (Chapter 2)
- **Sentence structure** - valid syntax patterns (Chapters 3-5)
- **Story structure** - constructing a program for a purpose

```
name = raw_input('Enter file:')
handle = open(name, 'r')
text = handle.read()
words = text.split()

counts = dict()
for word in words:
    counts[word] = counts.get(word,0) + 1
bigcount = None
bigword = None

for word,count in counts.items():
    if bigcount is None or count >
bigcount:
        bigword = word
        bigcount = count
print bigword, bigcount
```

A short “story” about  
how to count words  
in a file in Python

**python words.py**  
**Enter file: words.txt**  
**to 16**

# Reserved Words

- You cannot use **reserved words** as variable names / identifiers

and del for is raise assert elif from  
lambda return break else global  
not try class except if or while  
continue exec import pass yield  
def finally in print as with



# Sentences or Lines

<code>x</code>	<code>=</code>	<code>2</code>	←	Assignment statement		
<code>x</code>	<code>=</code>	<code>x</code>	<code>+</code>	<code>2</code>	←	Assignment with expression
<code>print</code>	<code>x</code>	←	Print statement			

Variable

Operator

Constant

Reserved  
Word

# Programming Paragraphs

# Python Scripts

- Interactive Python is good for experiments and programs of 3-4 lines long.
- Most programs are much longer, so we type them into a file and tell Python to run the commands in the file.
- In a sense, we are “giving Python a script”.
- As a convention, we add “.py” as the suffix on the end of these files to indicate they contain Python.

# Writing a Simple Program

# Interactive versus Script

- Interactive

- > You type directly to Python one line at a time and it responds

- Script

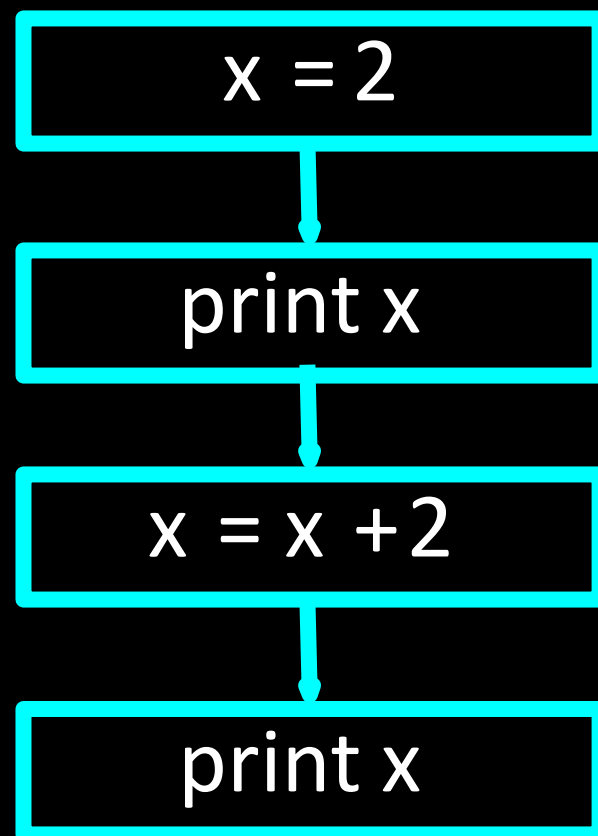
- > You enter a sequence of statements (lines) into a file using a text editor and tell Python to execute the statements in the file



# Program Steps or Program Flow

- Like a recipe or installation instructions, a program is a **sequence** of steps to be done in order.
- Some steps are **conditional** - they may be skipped.
- Sometimes a step or group of steps are to be **repeated**.
- Sometimes we store a set of steps to be used over and over as needed several places throughout the program (Chapter 4).

# Sequential Steps



Program:

`x = 2`

`print x`

`x = x + 2`

`print x`

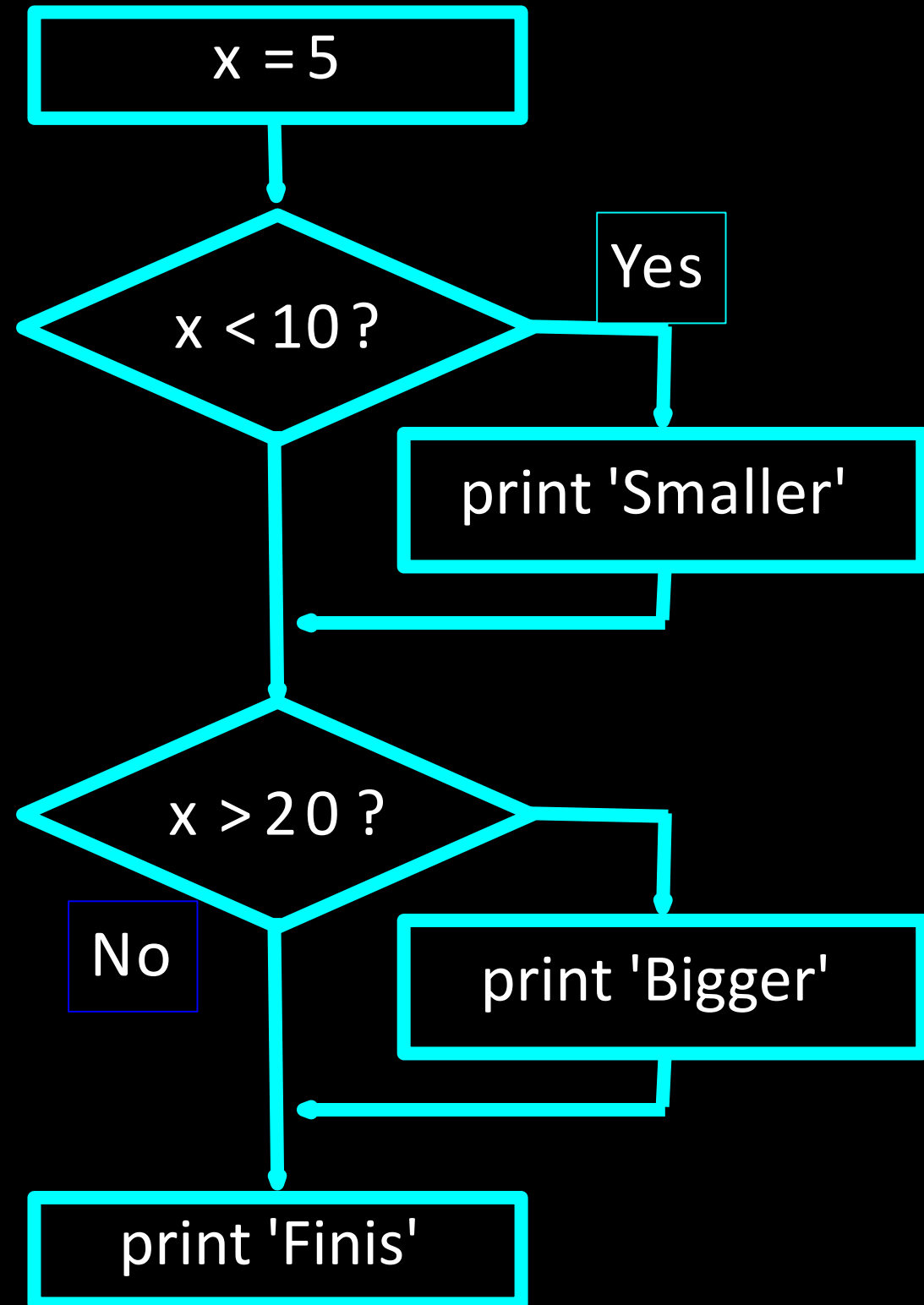
Output:

2

4

When a program is running, it flows from one step to the next.  
As programmers, we set up “paths” for the program to follow.

# Conditional Steps

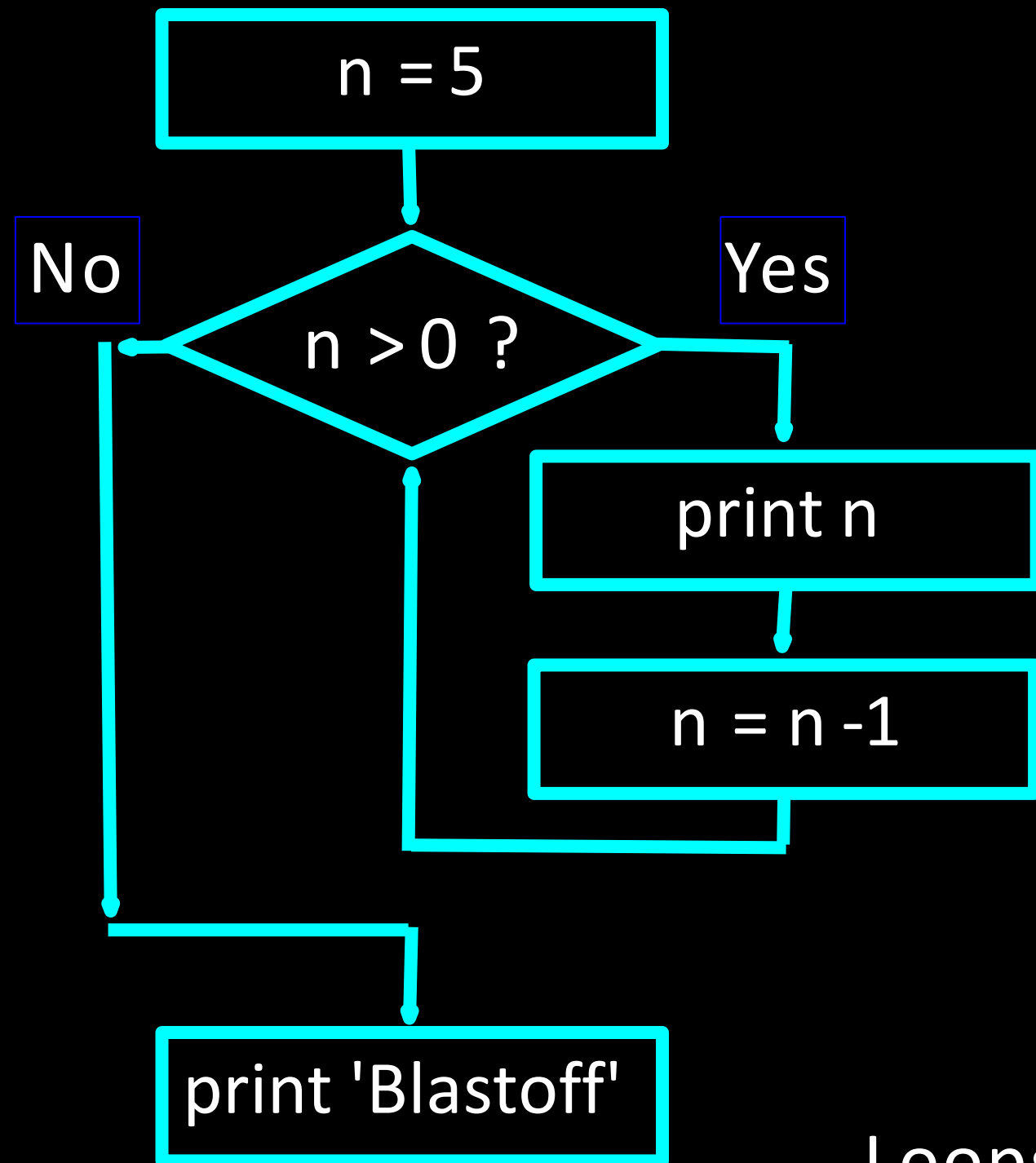


Program:

```
x = 5
if x < 10:
    print 'Smaller'
if x > 20:
    print 'Bigger'
print 'Finis'
```

Output:

Smaller  
Finis



# Repeated Steps

Program:

```
n = 5
while n > 0 :
    print n
    n = n - 1
print 'Blastoff!'
```

Output:

5  
4  
3  
2  
1  
Blastoff!

Loops (repeated steps) have **iteration variables** that change each time through a loop. Often these **iteration variables** go through a sequence of numbers.

```
name = raw_input('Enter file:')
handle = open(name, 'r')
text = handle.read()
words = text.split()

counts = dict()
for word in words:
    counts[word] = counts.get(word, 0) + 1
bigcount = None
bigword = None

for word, count in counts.items():
    if bigcount is None or count >
bigcount:
        bigword = word
        bigcount = count

print bigword, bigcount
```

Sequential

Repeated

Conditional



```
name = raw_input('Enter file:')  
handle = open(name, 'r')  
text = handle.read()  
words = text.split()  
counts = dict()  
for word in words:  
    counts[word] = counts.get(word, 0) + 1
```

```
bigcount = None  
bigword = None  
for word, count in counts.items():  
    if bigcount is None or count >  
bigcount:  
    bigword = word  
    bigcount = count
```

```
print bigword, bigcount
```

A short Python “Story”  
about how to count  
words in a file

A word used to read  
data from a user

A sentence about  
updating one of the  
many counts

A paragraph about how  
to find the largest item  
in a list

# Summary

- This is a quick overview of **Topic # 1**
- We will revisit these concepts throughout the course
- After the break, lets focus on the big-picture

# Acknowledgements / Contributions

These slides are Copyright 2017 University of Michigan School of Information and [open.umich.edu](https://open.umich.edu) and made available under a Creative Commons Attribution 4.0 License. Please maintain this last slide in all copies of the document to comply with the attribution requirements of the license. If you make a change, feel free to add your name and organization to the list of contributors on this page as you republish the materials.

