

# Python Foundations

# A computer does two things and two things only...

...it performs calculations and it remembers the results of those calculations.  
(Guttag, p1)

- Your job is to tell the computer HOW to do those calculations to achieve your goals
- Goals must be expressed as calculations, usually very small and structured calculations
- The structuring of calculations and how one calculation follows another are governed by control statements
- Python has the ability to **calculate** and **control**

# Python Calculations

## Operators

- `+`, `-`, `*`, `**`, `/`, `//`, `%`
- `+=`, etc

# Python Control

for

while

if

else

def

# Code example 1

```
sum = 0
count = 0
data = [1,5,8,2,0,9,10,4]
for number in data:
    sum += number
    count += 1
print(sum/count)
```

```
sum = 0
count = 0
data = [1,5,8,2,0,9,10,4]
for number in data:
    sum += number
    count += 1
print(sum/count)
```

## Code example 2

```
sum = 0
count = 0
file = open("data.csv")
for number in file:
    n = int(number)
    sum += n
    count += 1
print(sum/count)
```

```
sum = 0
count = 0
file = open("data.csv")
for number in file:
    n = int(number)
    sum += n
    count += 1
print(sum/count)
```



# Code example 3



```
def average(filename):  
    sum = 0  
    count = 0  
    file = open(filename)  
    for number in file:  
        n = int(number)  
        sum += n  
        count += 1  
    return (sum/count)
```

```
def average(filename):  
    sum = 0  
    count = 0  
    file = open(filename)  
    for number in file:  
        n = int(number)  
        sum += n  
        count += 1  
    return (sum/count)
```

## Code example 4



```
from pathlib import Path
```

```
def average(filename: Path) -> float:
    """Compute average of numbers in a file named filename."""
    sum = 0
    count = 0
    file = open(filename)
    for number in file:
        n = int(number)
        sum += n
        count += 1
    return(sum/count)
```

```
from pathlib import Path
def average(filename: Path) ->
float:
    """Compute average of numbers in a
file named filename."""
    sum = 0
    count = 0
    file = open(filename)
    for number in file:
        n = int(number)
        sum += n
        count += 1
    return(sum/count)
```



# Building Blocks of Code

Variables

Arithmetic Operators

Conditional Logic

Iteration

Functions

# Building Blocks of Code

Variables (essentially just storage)

Arithmetic Operators (for calculations)

Conditional Logic (for control)

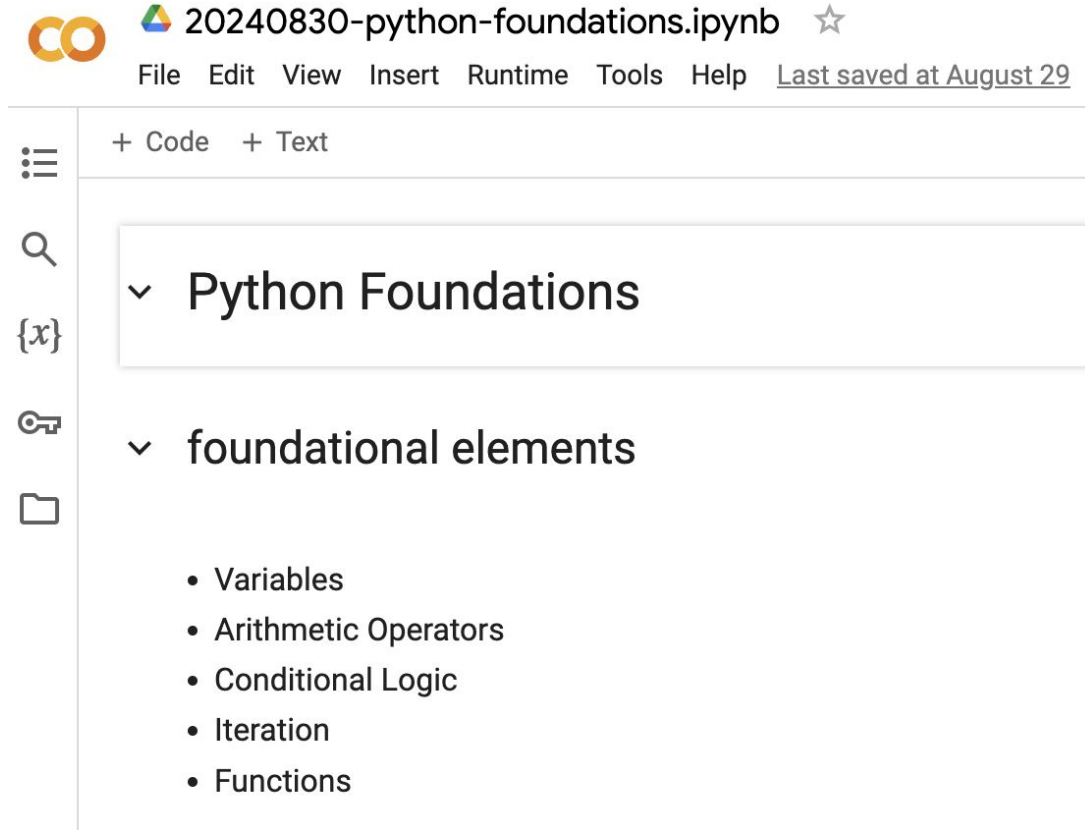
Iteration (for controlled calculations)

Functions (for convenience)

# Real Examples

# Google Colab

[https://colab.research.google.com/github/allegHENY-collegE-cmpsc-101-fall-2024/course-materials/blob/main/notes/20240830\\_python\\_foundations.ipynb](https://colab.research.google.com/github/allegHENY-collegE-cmpsc-101-fall-2024/course-materials/blob/main/notes/20240830_python_foundations.ipynb)



The screenshot shows the Google Colab interface. At the top, the notebook title is "20240830-python-foundations.ipynb" with a star icon. Below the title is a menu bar with "File", "Edit", "View", "Insert", "Runtime", "Tools", and "Help". The status bar indicates "Last saved at August 29". The left sidebar contains a file explorer with icons for a menu, search, a file named "{x}", a key, and a folder. The main area shows the notebook content with a code cell containing the following text:

```
+ Code + Text
```

Python Foundations

foundational elements

- Variables
- Arithmetic Operators
- Conditional Logic
- Iteration
- Functions