KTP Coding Workshop

Coding in Python: Basics

Schedule

01

Overview

Why is Python useful?

- ➔ Easy to Learn
  - ◆ Syntax resembles natural language - easy to pick up and identify errors
- ➔ Many libraries to do many different things with
  - ◆ Data visualization, machine learning, web development and more
- ➔ Large community of users
  - ◆ Lots of resources to aid in learning and solution assistance
- ➔ Can handle large datasets

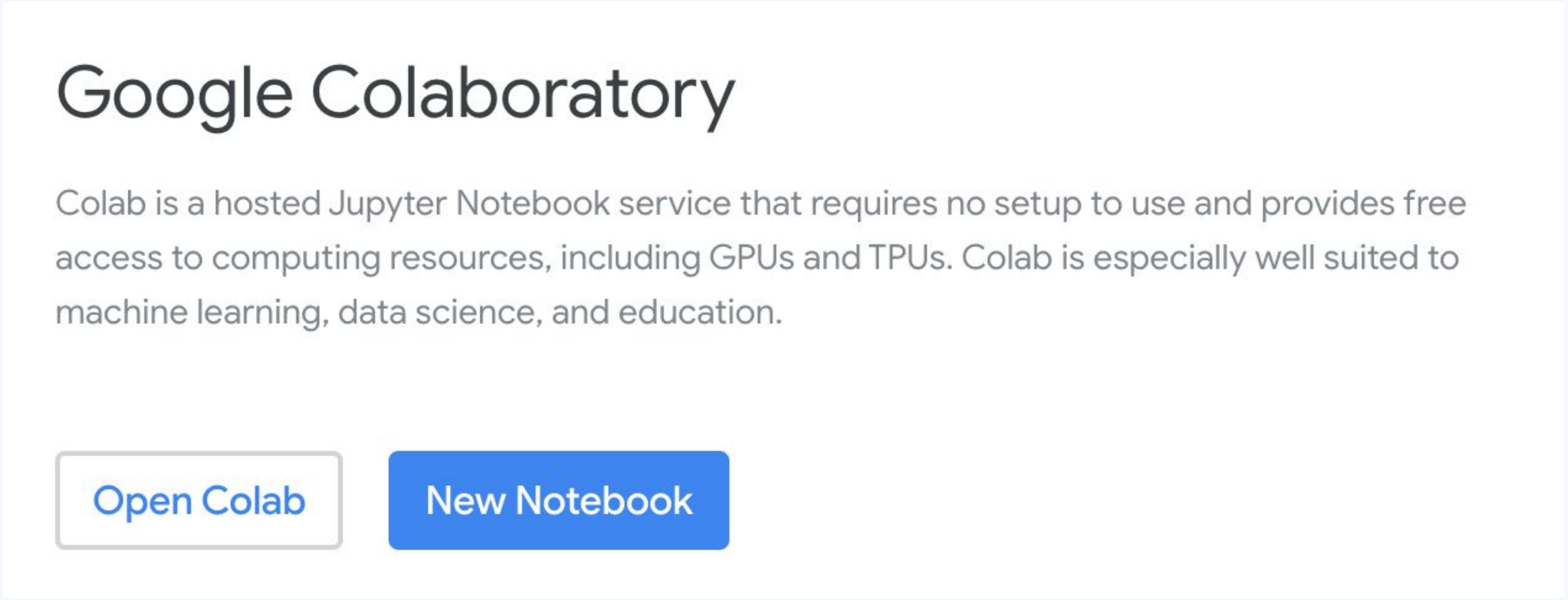
02

Jupyter Notebooks

So How Do You Use Python?

- Jupyter Notebooks
    - Open-source web-based application
- Google Colab
    - Free, cloud-based platform
    - You can run Python code in a Jupyter Notebook through Colab

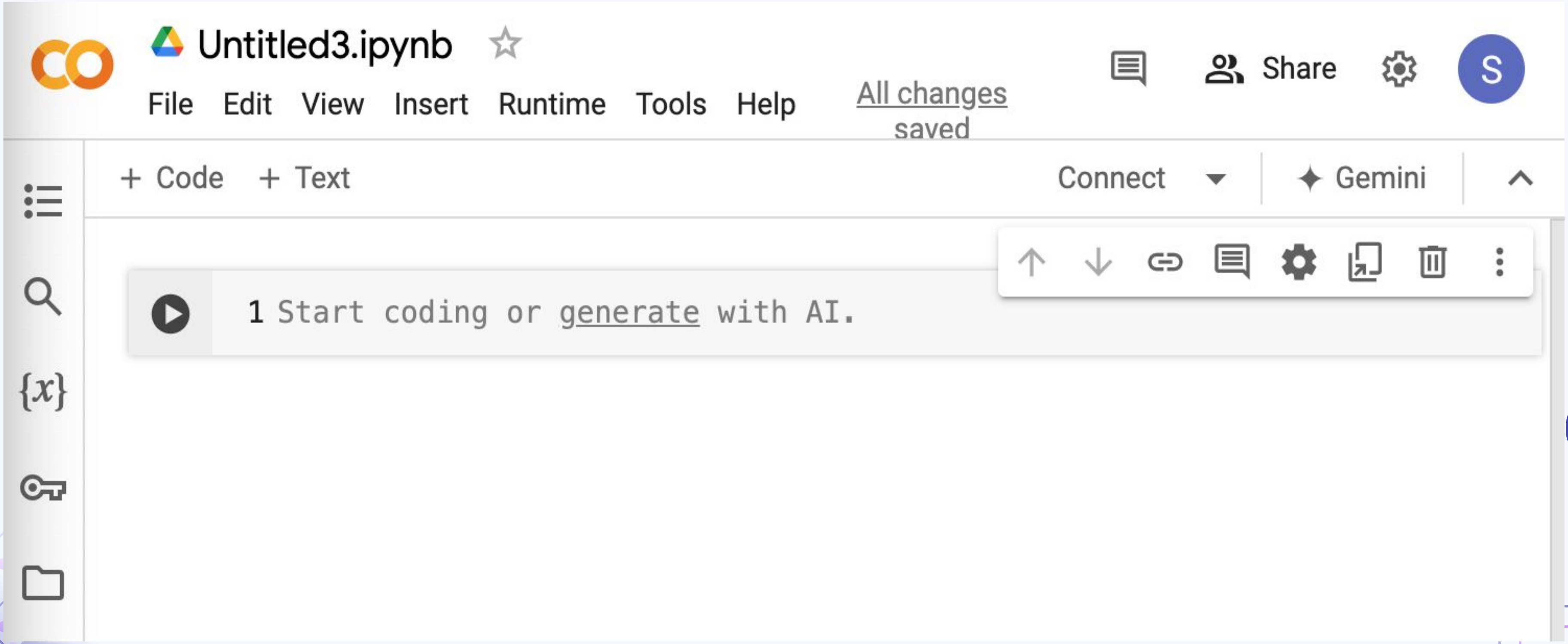
How to Open Google Colab



Google Colaboratory

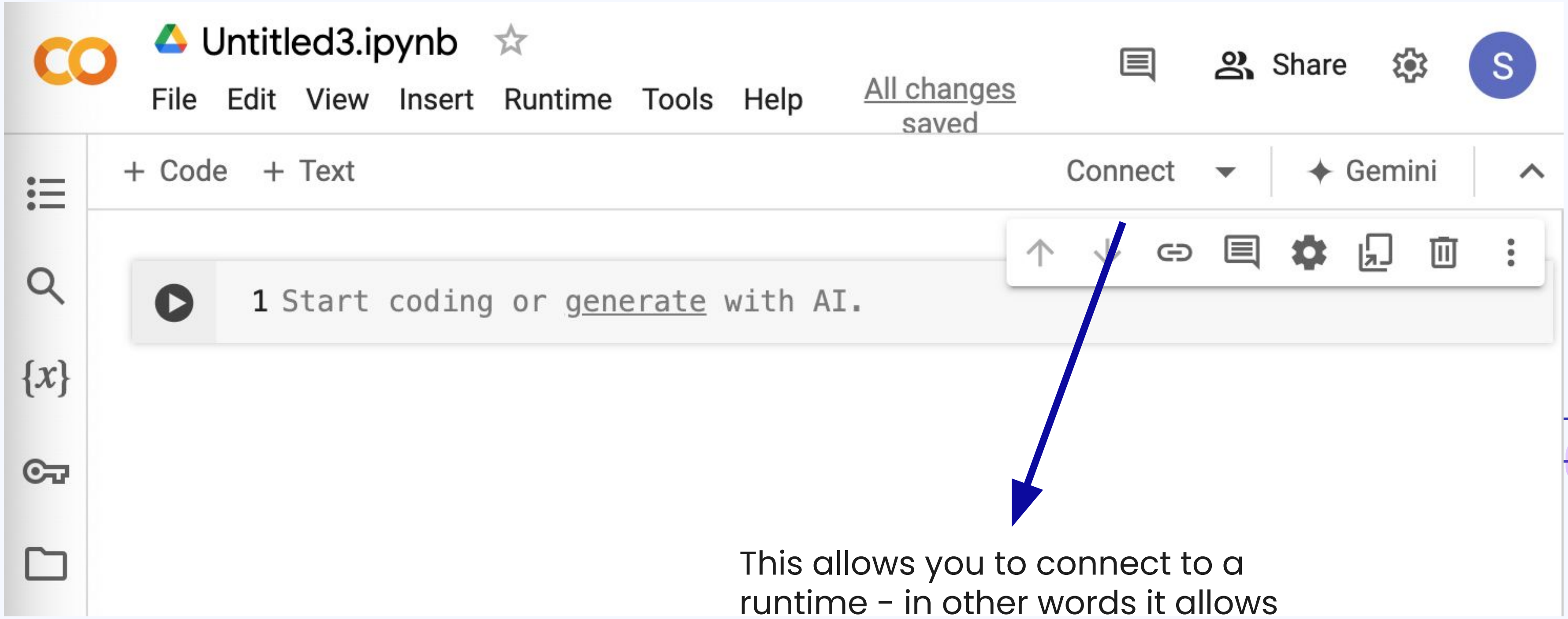
Colab is a hosted Jupyter Notebook service that requires no setup to use and provides free access to computing resources, including GPUs and TPUs. Colab is especially well suited to machine learning, data science, and education.

Should Look Like This Now



```
1 Start coding or generate with AI.
```

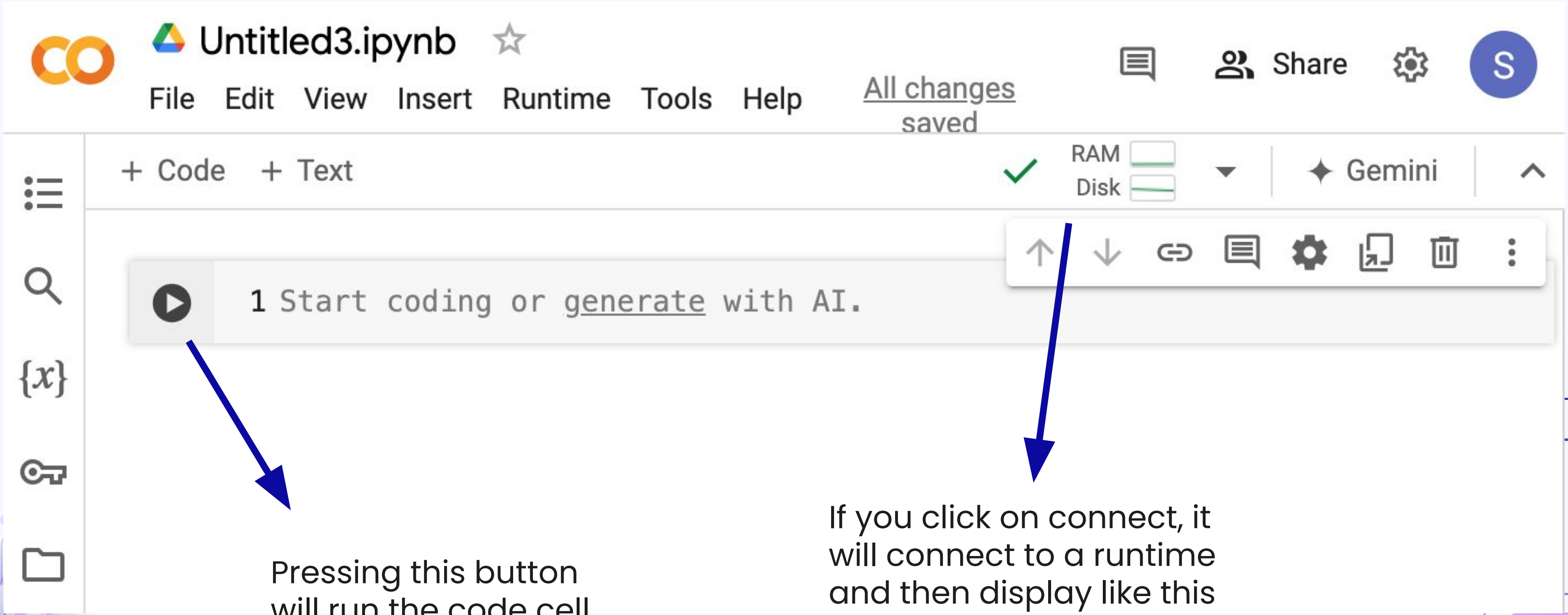
What Does Everything Mean?



```
Start coding or generate with AI.
```

This allows you to connect to a runtime - in other words it allows your code to process!

What Does Everything Mean?

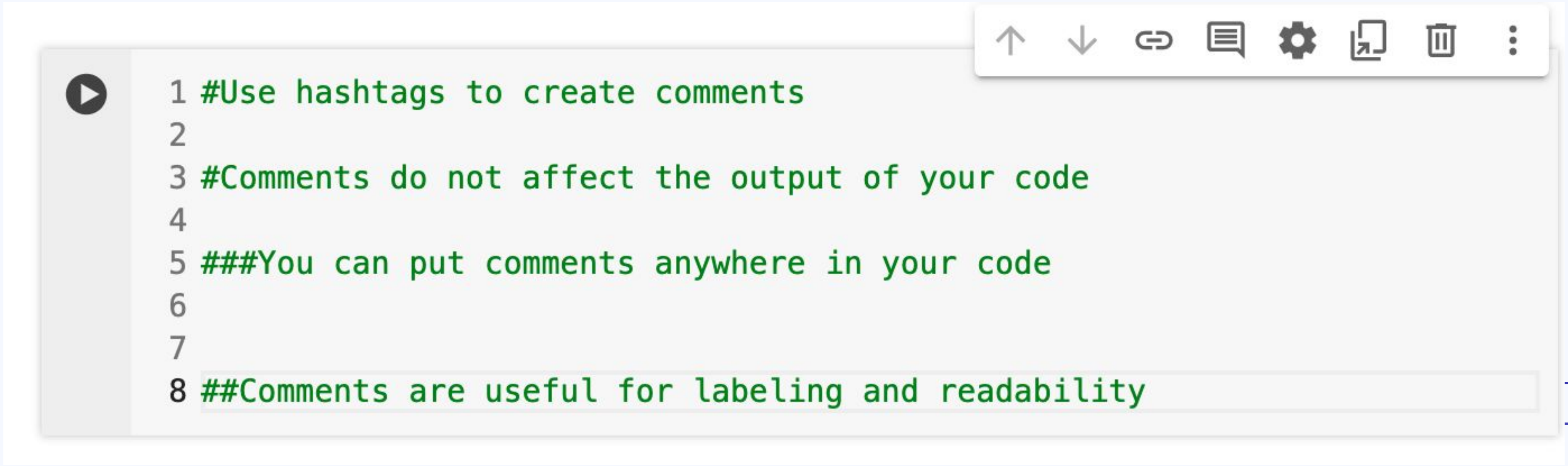


```
Start coding or generate with AI.
```

Pressing this button will run the code cell

If you click on connect, it will connect to a runtime and then display like this - now your code can run

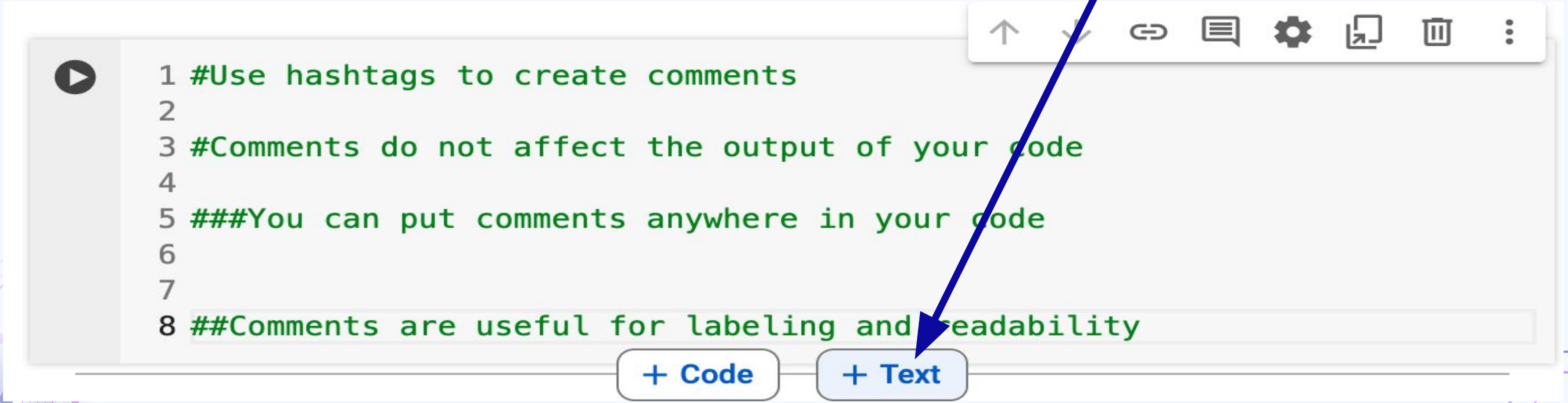
Creating Comments



```
#Use hashtags to create comments

#Comments do not affect the output of your code

###You can put comments anywhere in your code


##Comments are useful for labeling and readability
```

Headers and Text Boxes

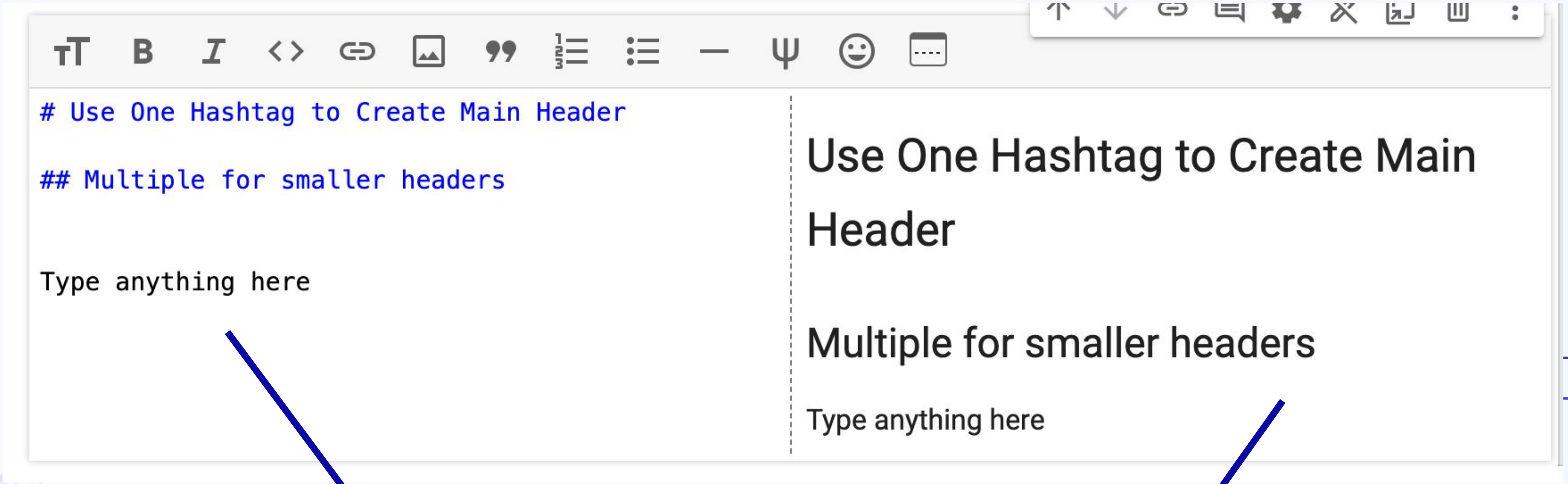
- To create a text cell
  - You can use Command MM (on a mac)
  - Hover near the top or bottom of your current cell and then click "Text"



```
#Use hashtags to create comments

#Comments do not affect the output of your code

###You can put comments anywhere in your code


##Comments are useful for labeling and readability
```

Add Text Box

Headers and Text Boxes



```
# Use One Hashtag to Create Main Header

## Multiple for smaller headers


Type anything here
```

Use One Hashtag to Create Main Header

Multiple for smaller headers

Type anything here

This is your code

This is the output

03

Basic Syntax

Data Types

int (integer)

Represents integer values (whole numbers)

3, -5, 0 are all integer values

float values

Represents a floating point number (decimal)

31.0, -0.987, 45.24

str (string)

Represents text (string)

Enclosed with quotes - single or double

“Hello”, ‘Python is Cool’, “3 is the best number”

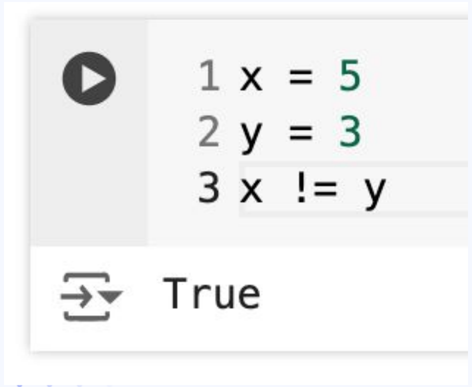
boolean

True or False value

Connected to comparison operators: <, >, !=, ==

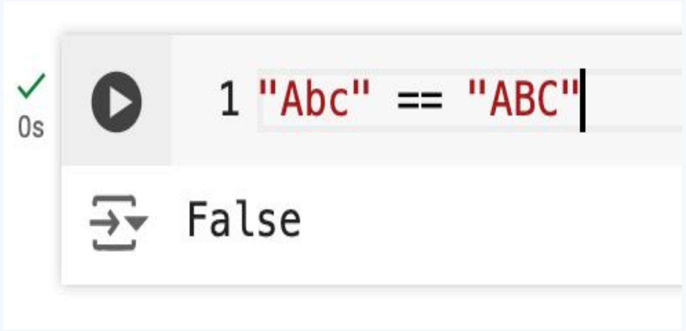
More on Boolean Values

- **Syntax**
  - True
  - False
- **Comparison Operators**
  - != means does not equal
  - == means equal
- **You can compare string, integers and float values**



```
x = 5
y = 3
x != y
```

```
True
```



```
"Abc" == "ABC"
```

```
False
```

Using ‘And’ and ‘Or’ To Compare

- **And**
  - True and True = True
  - True and False = False
  - False and False = False

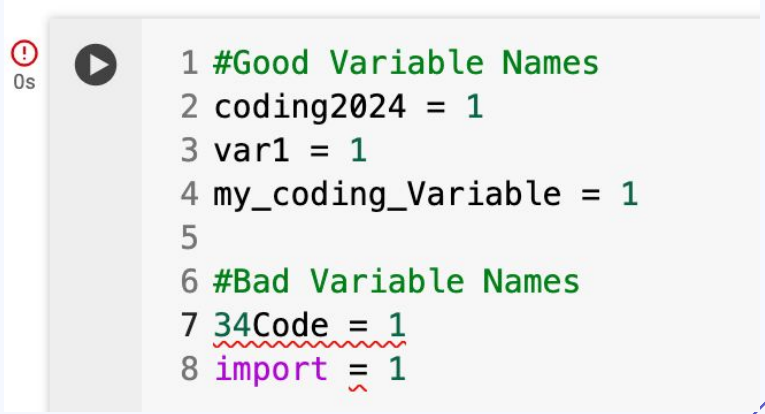
- **Or**
  - True or True = True
  - True or False = True
  - False or False = False

Naming Variables

➔ **Assign Variables using '='**
- ◆ Ex:
  - thisVariable = 4

➔ **Do's**
- ◆ Use letters and numbers
- ◆ Use underscores
- ◆ False or False = False
- ◆ Recognize case sensitivity

➔ **Don'ts**
- ◆ Start with a number
- ◆ Use Python keywords as a variable name



0s

```
#Good Variable Names
coding2024 = 1
var1 = 1
my_coding_Variable = 1

#Bad Variable Names
34Code = 1
import = 1
```

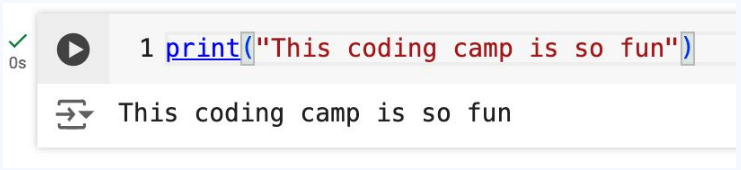
04

Basic Functions

Basic Functions

print()

Displays result to the screen

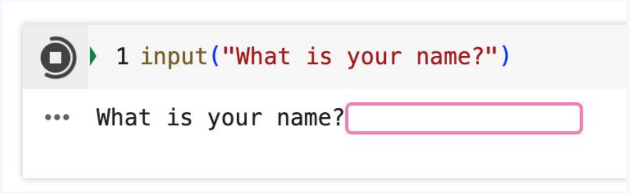


```
print("This coding camp is so fun")
```

```
This coding camp is so fun
```

input()

Allows user to enter input

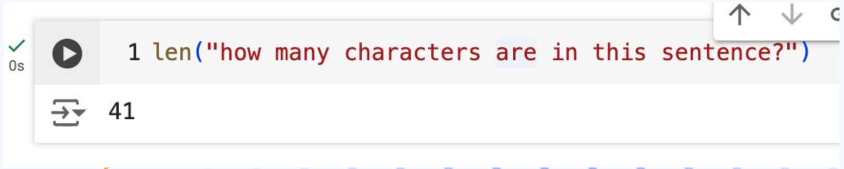


```
input("What is your name?")
```

```
What is your name?
```

len()

Returns the length of an object

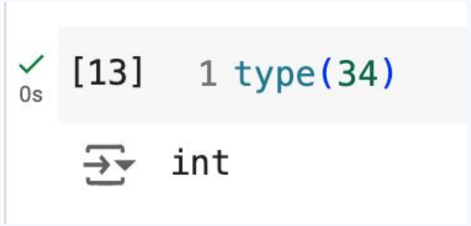


```
len("how many characters are in this sentence?")
```

```
41
```

type()

Returns the data type



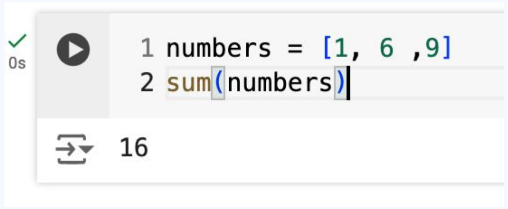
```
[13] type(34)
```

```
int
```

More Basic Functions

sum()

Returns the sum

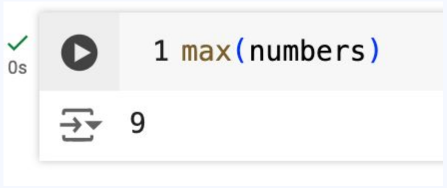


```
numbers = [1, 6 ,9]
sum(numbers)
```

```
16
```

max() and min()

Returns the max and min values

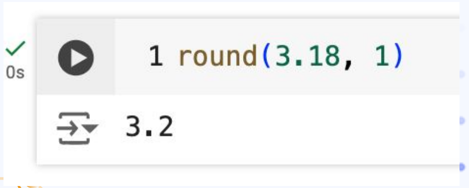


```
max(numbers)
```

```
9
```

round()

Rounds a float to specified number of decimals

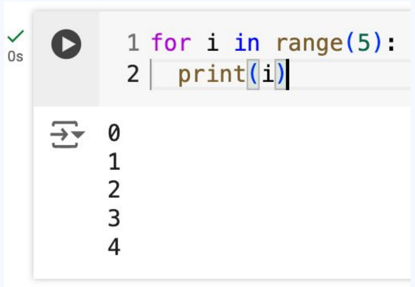


```
round(3.18, 1)
```

```
3.2
```

range()

Returns a sequence of numbers



```
for i in range(5):
  print(i)
```

```
0
1
2
3
4
```

05

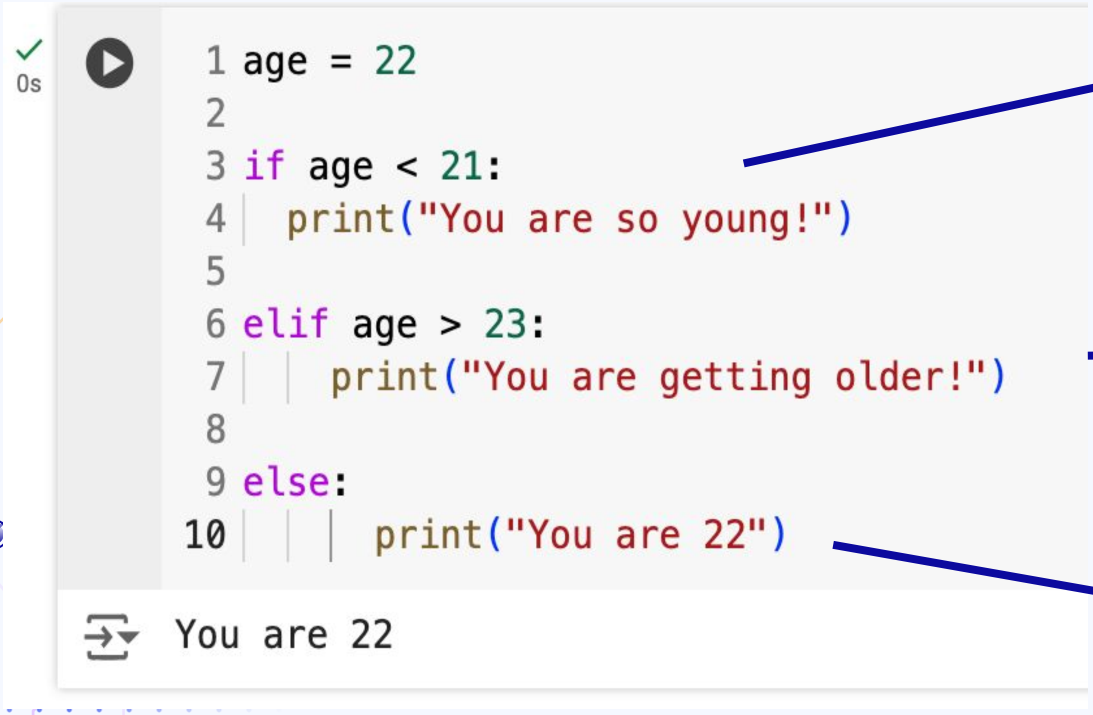
Control Flow

Control Flow - If Statements

- ➔ **Tells python what to execute and in what order using conditional statements**
  - ◆ If, else and elif
  - ◆ Elif means “else if”

- ➔ **Syntax is very important in control flow**
  - ◆ Use of colons
  - ◆ Use of indentation

Control Flow - Conditional Statements



```python
age = 22

if age < 21:
  print("You are so young!")

elif age > 23:
    print("You are getting older!")

else:
      print("You are 22")
```

```
You are 22
```

'If' checks a condition. If it is true, it will execute.

'Elif' checks additional conditions in the case that the if condition is not true

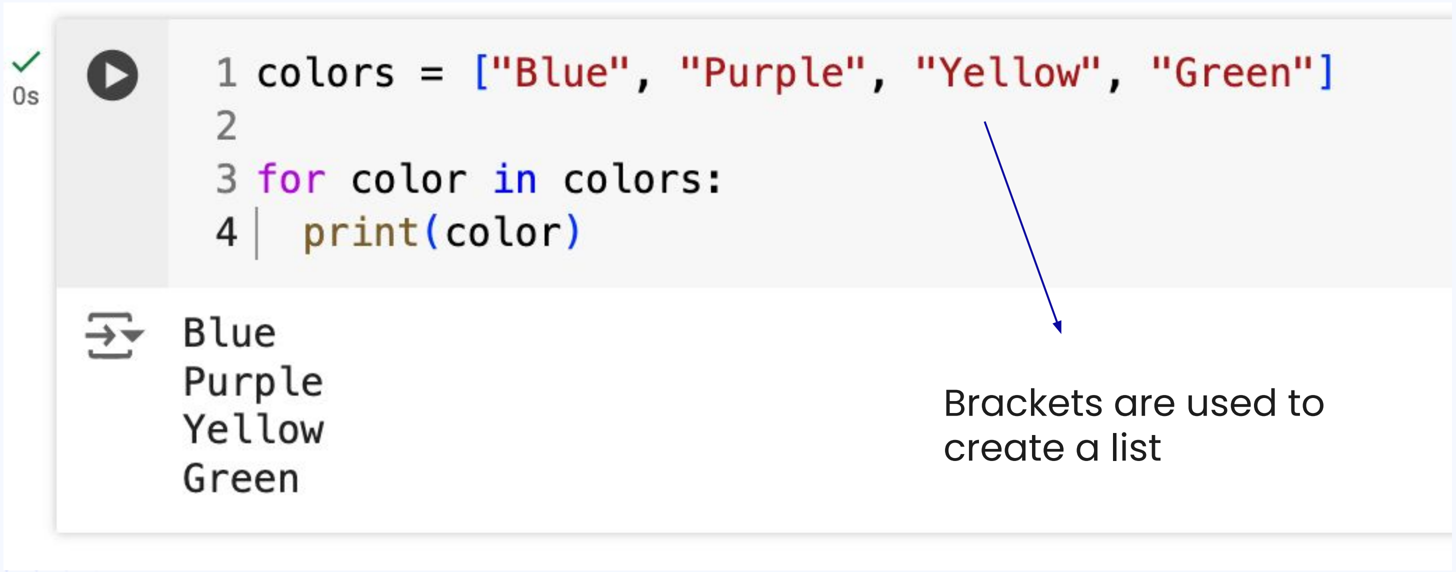
'Else' runs only if all other conditions are false

06

Loops

Control Flow - For Loops

- **For loops allow you to iterate over a sequence**
  - Useful if you want to print out something multiple times



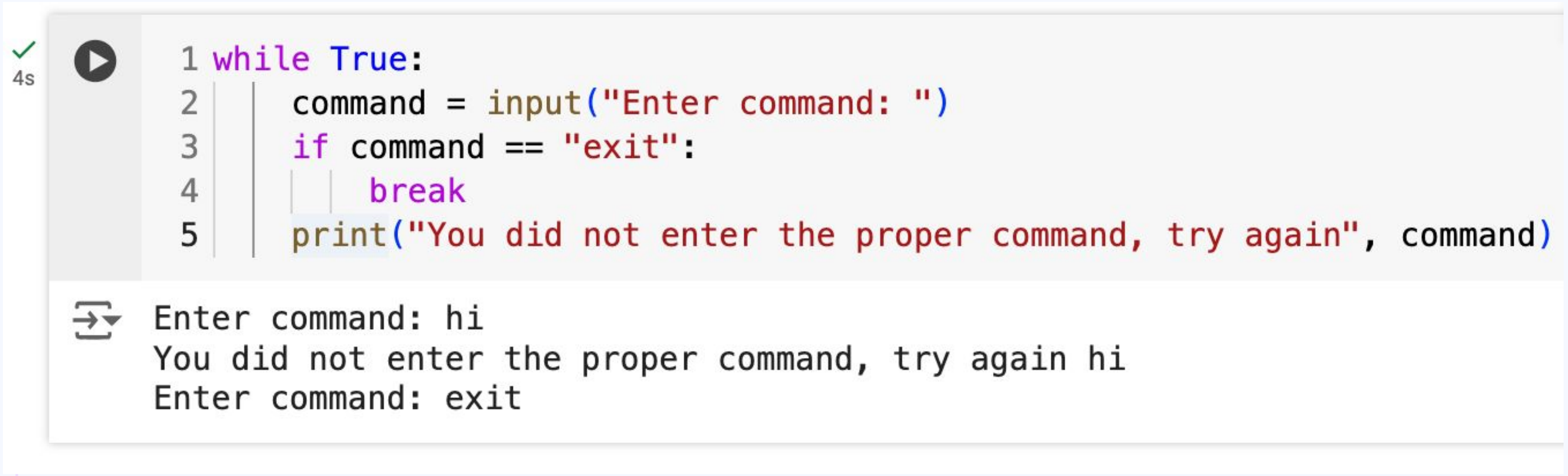
```python
colors = ["Blue", "Purple", "Yellow", "Green"]

for color in colors:
  print(color)
```

```
Blue
Purple
Yellow
Green
```

Brackets are used to create a list

Control Flow - While Loops

- ➔ **While loops will continue running - so long as a condition is true**
  - ◆ Useful if you are not sure how many times the loop will have to execute until you get the result you want



```python
while True:
    command = input("Enter command: ")
    if command == "exit":
        break
    print("You did not enter the proper command, try again", command)
```

```
Enter command: hi
You did not enter the proper command, try again hi
Enter command: exit
```

SO MUCH MORE TO LEARN!

- ➔ Object oriented programming
- ➔ Lists and Tuples
- ➔ Dictionaries
- ➔ Pandas
- ➔ Importing Files
- ➔ Data Visualization
- ➔ ...MORE!!!

- ➔ Helpful Resources
  - ◆ W3 Schools
  - ◆ Kaggle
  - ◆ LinkedIn Learning
  - ◆ Generate AI on Google Colab

THANKS FOR LISTENING!!!