TERRAPIN WORKS

Introduction to Python

https://go.umd.edu/TWSP25

Students please sign in for the TW Workshop!

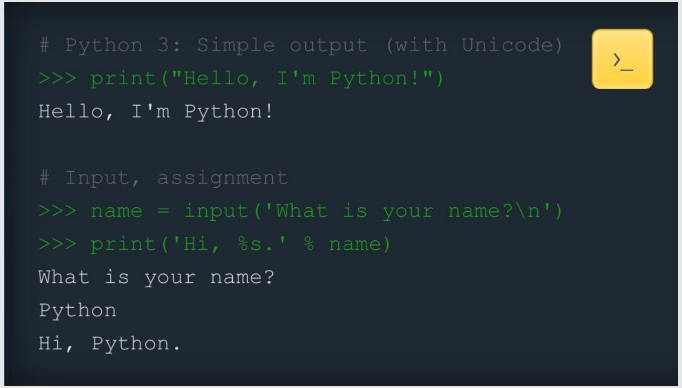
Sign Into Replit

replit.com/languages/python3

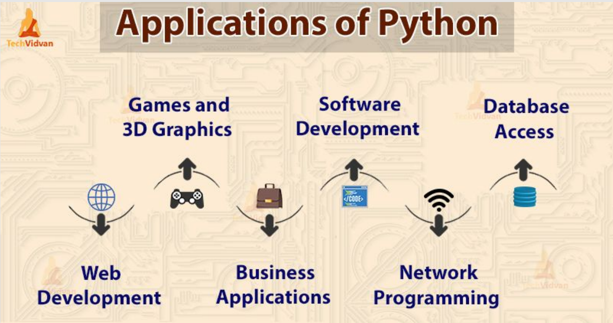
Replit is an integrated development environment (IDE) or an online platform that allows users to create online programming and run, edit, and share code.

What is Python?

Programming language with many uses including creating websites, software, automating tasks, data analysis, and more.

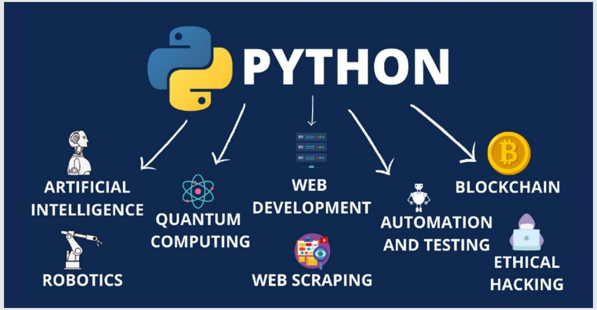


```
# Python 3: Simple output (with Unicode)
>>> print("Hello, I'm Python!")
Hello, I'm Python!

# Input, assignment
>>> name = input('What is your name?\n')
>>> print('Hi, %s.' % name)
What is your name?
Python
Hi, Python.
```



What’s so Special About Python?

- ⭐ Emphasizes code readability and allows you to use English keywords without using punctuation.
- ⭐ Used by applications/websites like Instagram, Uber, and many video games.
- ⭐ Provides extensive libraries, and a supportive community
- ⭐ Can create custom applications

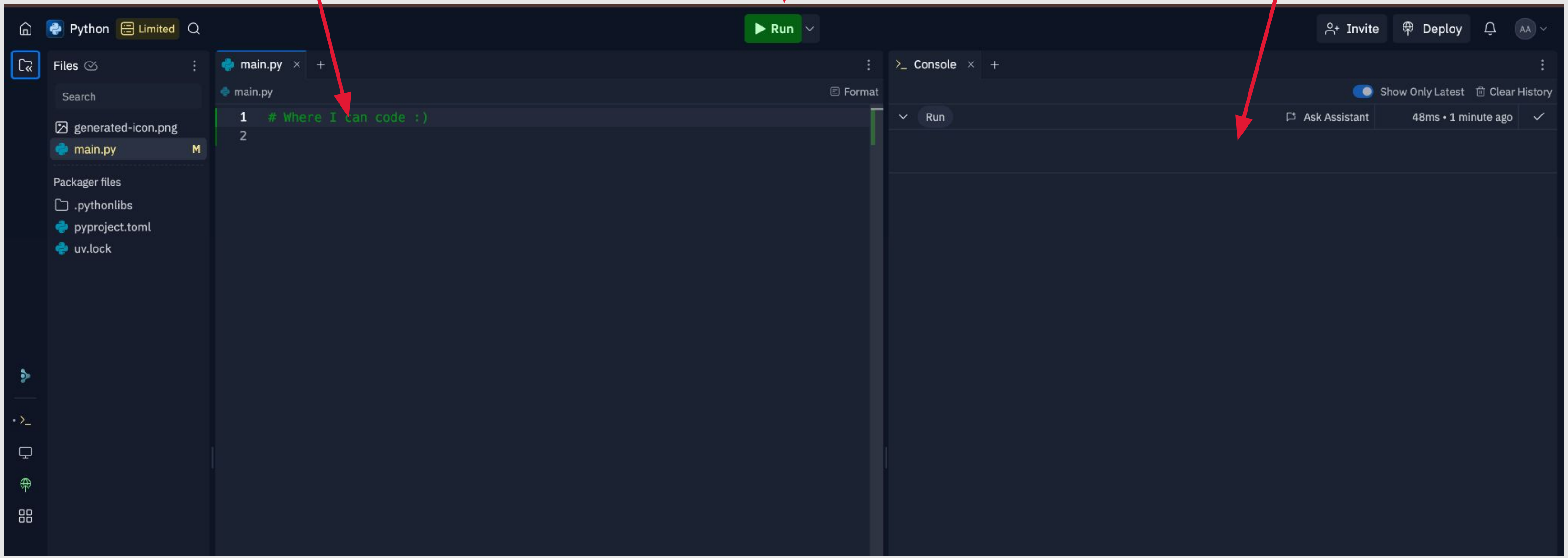


Replit Work Area

Coding Area

Hitting the run button executes your code.

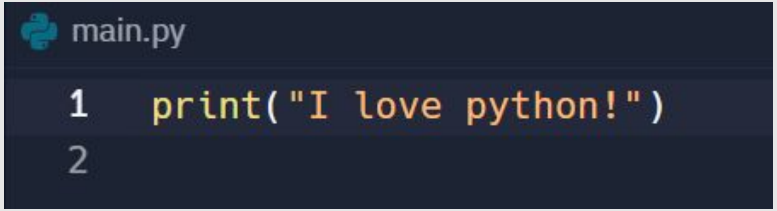
Console: Where Python programs run. The result of running your code, or errors are displayed.



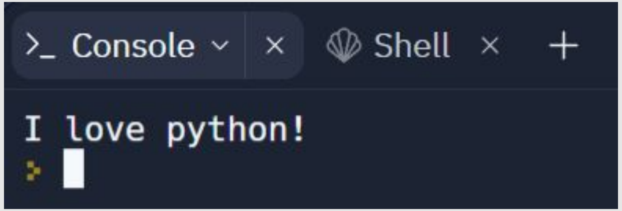
```
# Where I can code :)
```

Print Statements

print() is a built-in Python function that takes in a value or argument in between the parentheses, and outputs the argument to the console.



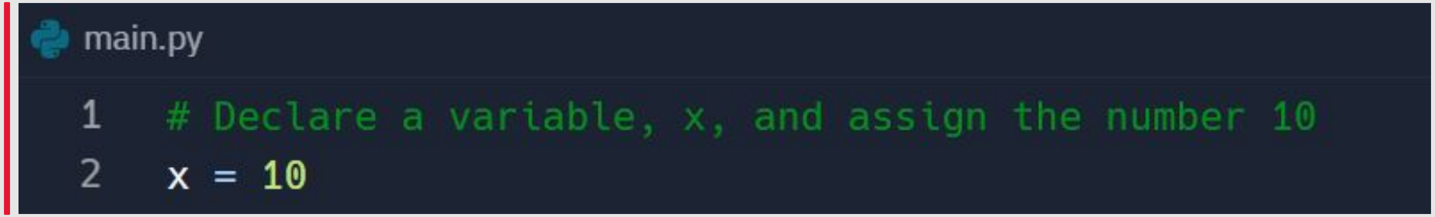
```
print("I love python!")
```



```
I love python!
```

syntax	console output
print(“hi”) or print(‘hi’)	hi
print(25)	25
print(10 * 4)	40

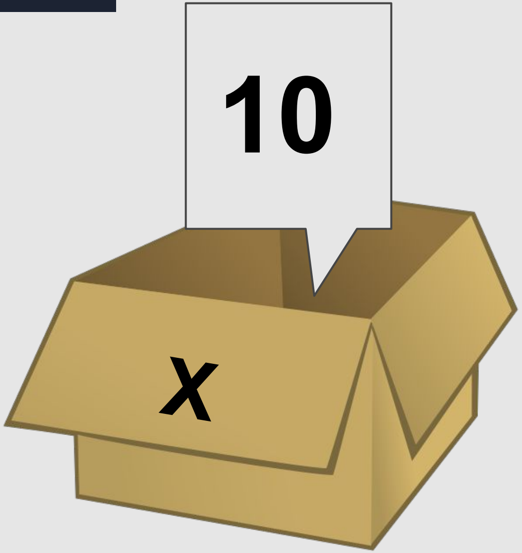
Variables



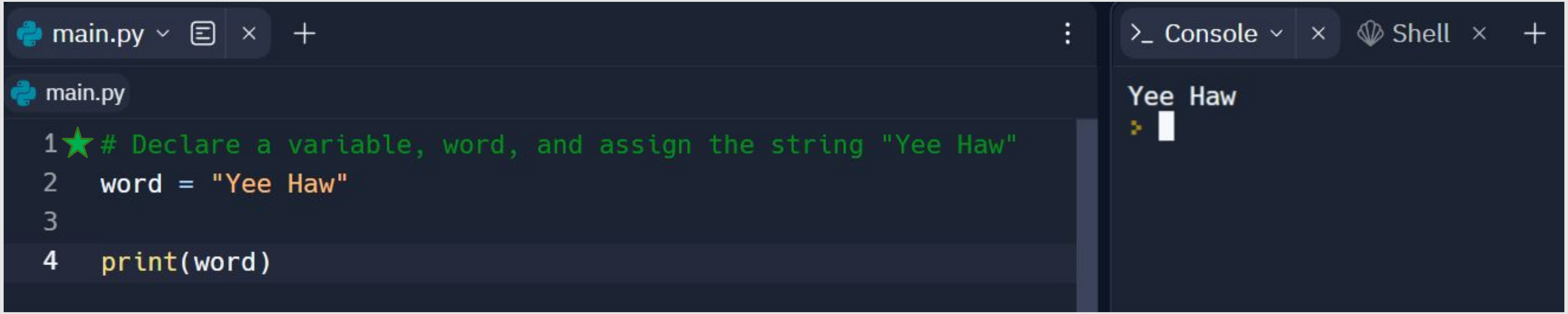
main.py

```python
# Declare a variable, x, and assign the number 10
x = 10
```

Variables store data that can be later referenced or used.



Variables: Another Example



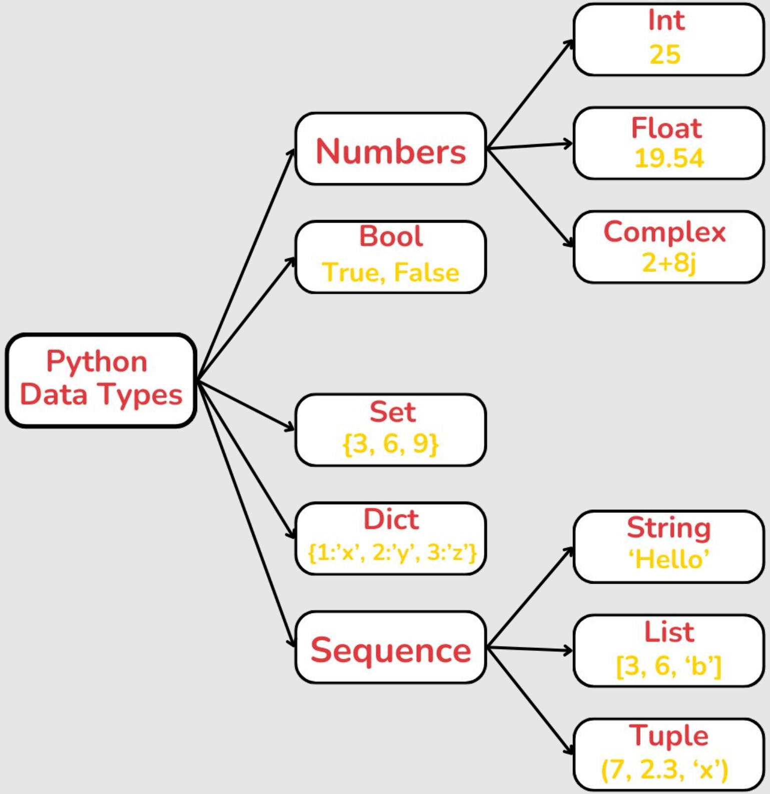
```python
# Declare a variable, word, and assign the string "Yee Haw"
word = "Yee Haw"

print(word)
```

```
Yee Haw
>
```

When naming variables, keep in mind that python has variable name restrictions.

# marks a single line comment. Comments do not affect the execution of code.

Variable Types



Python Data Types

Numbers

Int
25

Float
19.54

Complex
2+8j

Bool
True, False

Set
{3, 6, 9}

Dict
{1:'x', 2:'y', 3:'z'}

Sequence

String
'Hello'

List
[3, 6, 'b']

Tuple
(7, 2.3, 'x')

- Python has data types built-in by default.

- You do not have to specify the type when assigning a variable because Python infers the type.

- The type(*variable_name*) function can show the type of a variable.

temp = 80.4 print(type(temp)) → float

flag = False print(type(flag)) → bool

Operators

Arithmetic Operators

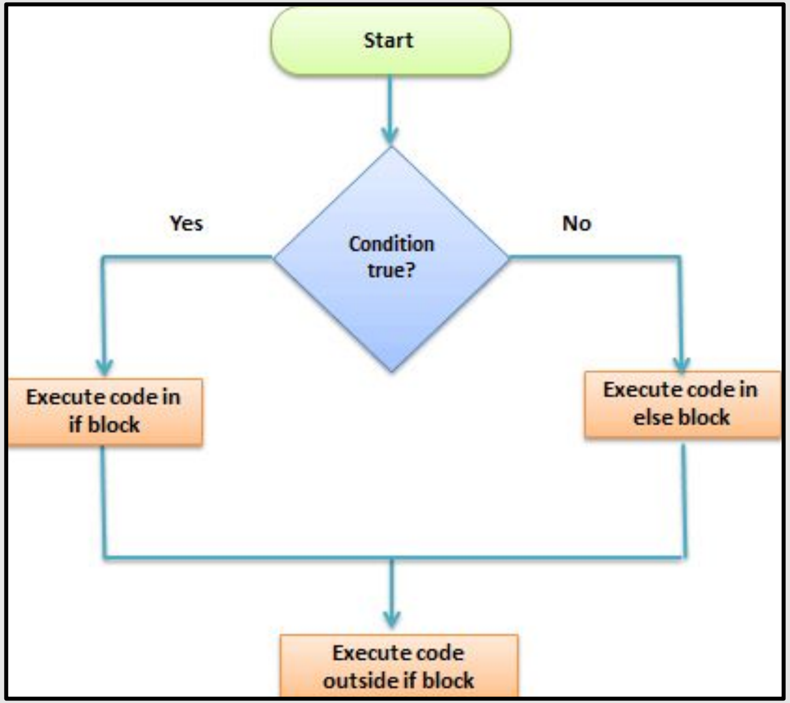
Multiply	Subtract	Add	Divide	Exponent
*	-	+	/	**

Conditional Operators

Greater	Less	Equality	Less,Equal	Greater,Equal
>	<	==	<=	>=

And	Or	Not
and	or	not

Conditional Statements



- A conditional is a situation where you see the pattern of:

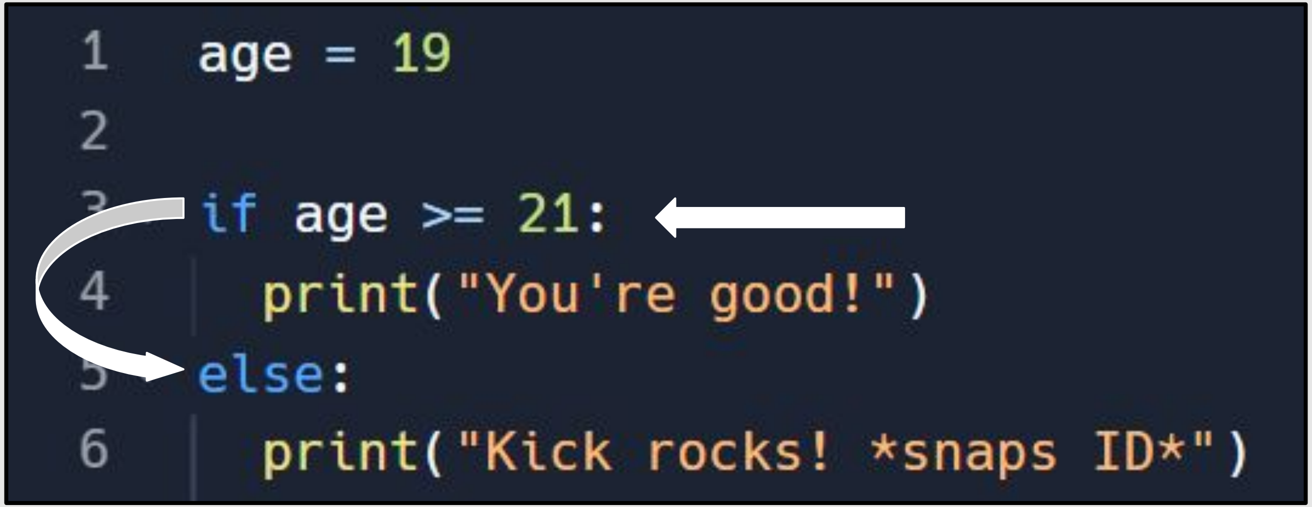
  if __, then __.

- If a condition is met, something will happen as a result.

Conditional Statements: Execution Pattern

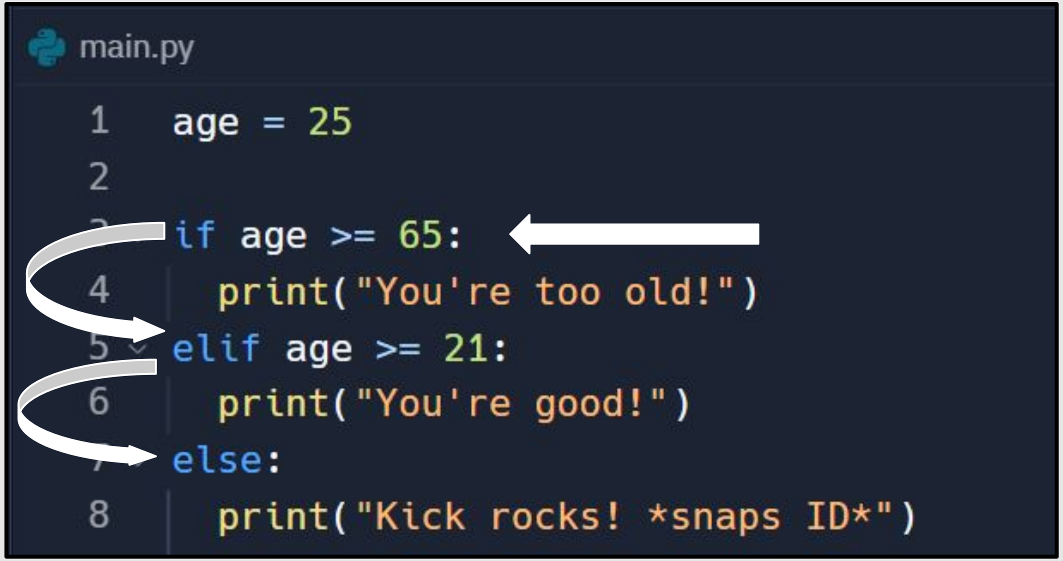
Conditional Keyword + Syntax	What it does?
if condition: body	If the condition is **met** then the body will be executed. If not, then the body will be skipped.
elif condition: body	If none of the previous conditions were true, execute the body. You can have multiple elif statements.
else: body	If none of the previous conditions were true, execute the body.

Conditional Statements: Bouncer Example



```python
age = 19

if age >= 21:
    print("You're good!")
else:
    print("Kick rocks! *snaps ID*")
```

Conditional Statements: Bouncer Example + A Twist



```python
age = 25

if age >= 65:
  print("You're too old!")
elif age >= 21:
  print("You're good!")
else:
  print("Kick rocks! *snaps ID*")
```

Try It, What Would be Printed?

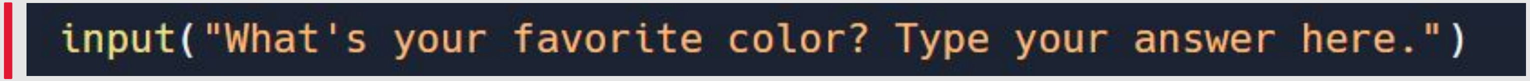
```
x = 7
y = 10

if x + y > 15:
    if x == 35 and y == 12:
        print("A")
    elif x - y <= 0 or x + 7 > 30:
        print("B")
    else:
        print("C")
elif x + y > 0:
    if x + y == 17:
        print("C")
else:
    print("D")
```

Answer:

B

- Note that the outer elif and else clauses are never reached in this case.

Capturing User Input



```
input("What's your favorite color? Type your answer here.")
```

Syntax: **input**(“message”)

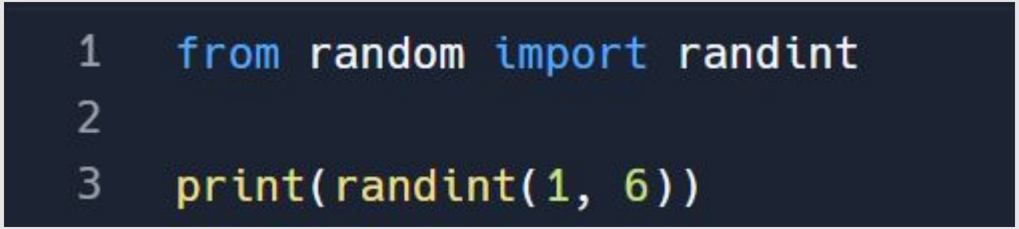
The input function is similar to the print function. However, the input function subsequently asks the user to put in an answer.

The console will print the *message* in quotation marks, and then wait for user input.

Generating Random Numbers

Chooses a random number inside of a range.

Syntax: **randint**(*min number, max number*)



```
from random import randint

print(randint(1, 6))
```

Randomly generates a number:
1, 2, 3, 4, 5, or 6

Review: What We’ve Learned

Print Function

print(“*message*”)

Variables

x = “terrapin”
y = 1738

Conditionals

if, else, elif

Input Function

input(“*message*”)

randint()

randint(*min*, *max*)

Now it’s Your Turn: Rock Paper Scissors

Rules:

1. Each player randomly chooses: Rock, Paper, or Scissors
   - Players are playing against the computer which also randomly chooses what to play.
2. Rock beats Scissors
3. Scissors beats Paper
4. Paper beats Rock
5. If both the player and computer select the same thing, it’s a Tie

Things to Keep in Mind:

1. Ensure that your program allows the user to enter what they want to play (rock, paper, or scissors)

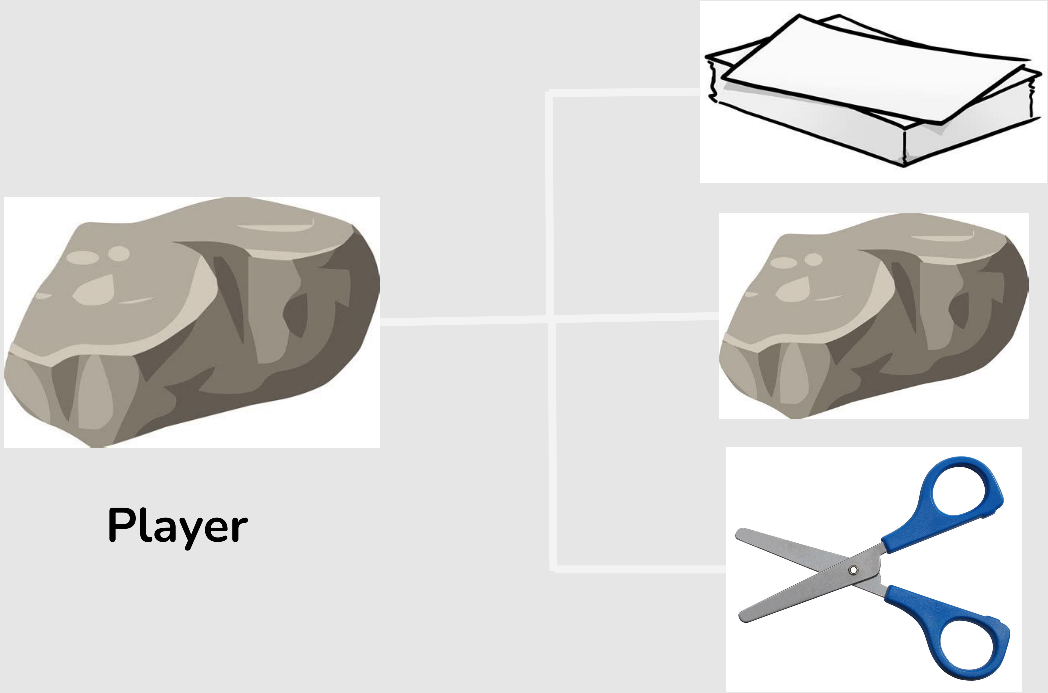
2. Be sure you account for the different cases in what both players can play.

   (ie. Player chooses Paper, computer chooses rock, Player wins.)

3. Ensure that the computer choice of what to play is random.

4. Print out the winner at the end.

Possible Outcomes



You Lose ☹

Tie

You Win!

Player

Computer

Example of a Completed Rock, Paper, Scissors Code

https://replit.com/@JasonSteinberg/CharmingHorizontalCore#main.py

Thank You

Check our our other **introductory workshops!**

- Soldering
- Motors & Servos
- Basic Electronics
- Sensors