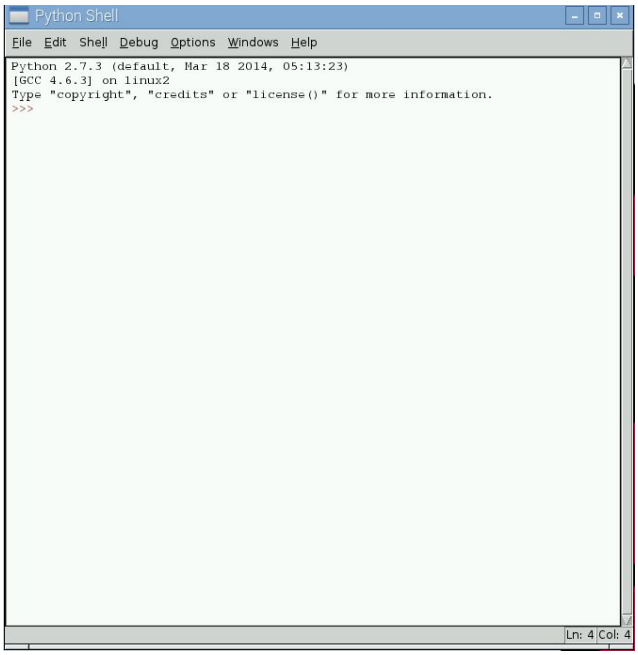
We will be using **Python 2**. Remember that Codecademy has prepared you for some of the programing concepts in Python, you just have to build off of them!

Open Python on the Raspberry Pi. Go to: Menu > Programing > Python 2

Once open, you will see the shell (commonly referred to as IDLE):

The Python shell is an interactive tool used for writing and evaluating code.

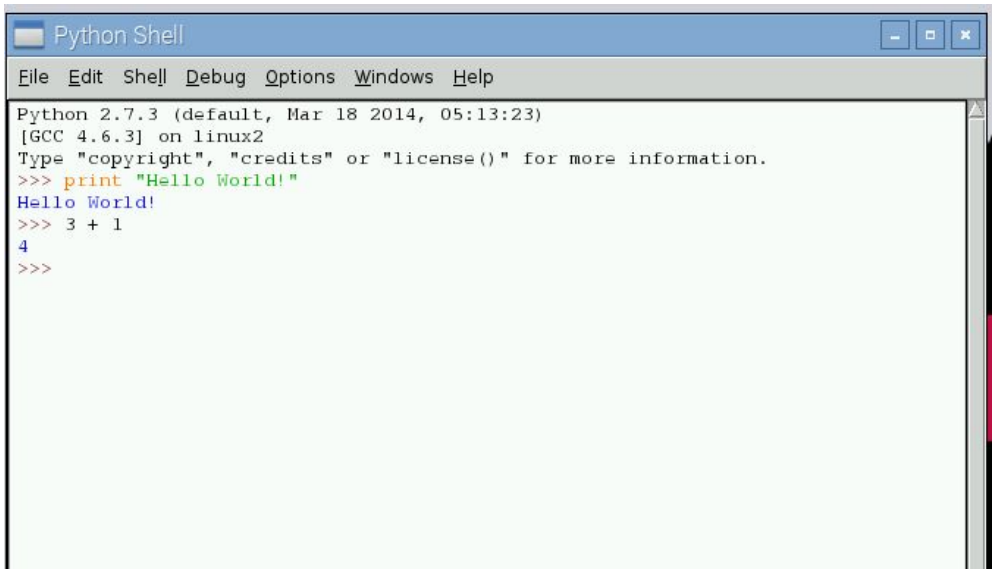


Try entering something into the shell:

- Type the following then press enter:

  print "Hello World"
- Now try a simple expression:

  3 + 1



- Do you remember what variables are? Try creating some:

  x = 5

  y = 5

  print x

  print y

```
Python 2.7.3 (default, Mar 18 2014, 05:13:23)
[GCC 4.6.3] on linux2
Type "copyright", "credits" or "license()" for more information.
>>> x = 5
>>> y = 5
>>> print x
5
>>> print y
5
>>>
```

- **Try some of the following**:

  Add x and y together by entering *x + y*. What is the output?

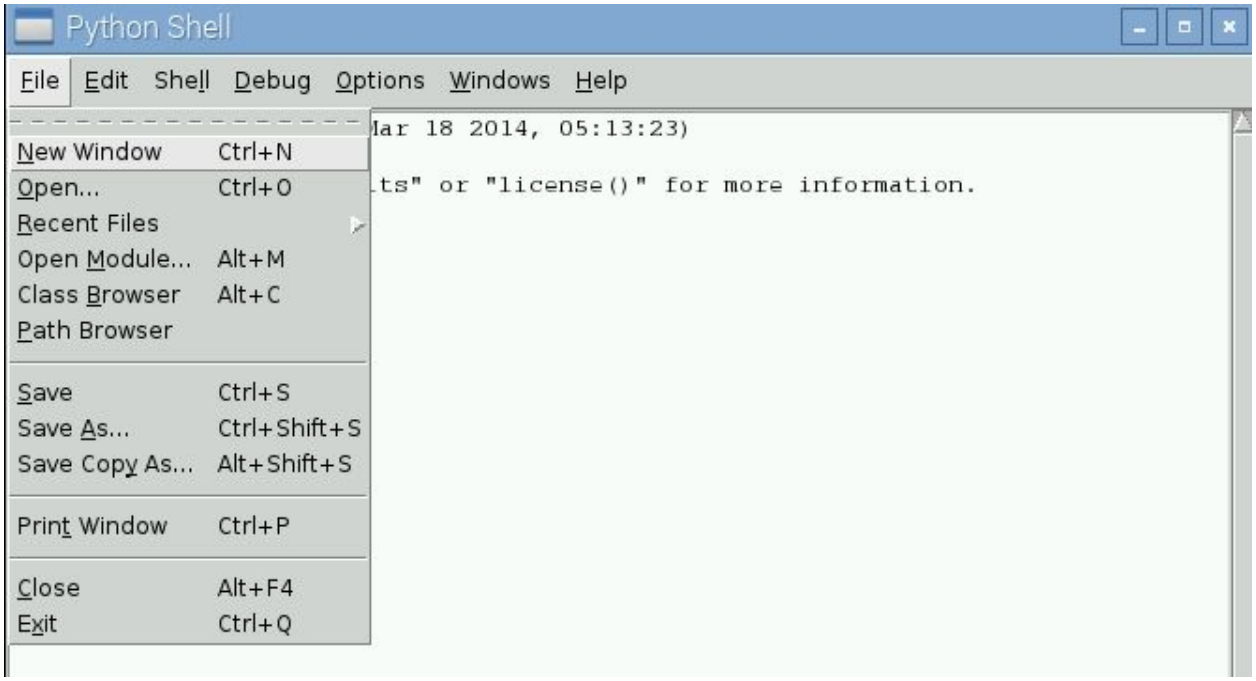
  Enter: *((3 + 2) * 5) - 3* You should get 22 as the answer.

  Try: *2**2* (Means 2 to the power of 2) which should equal 4.

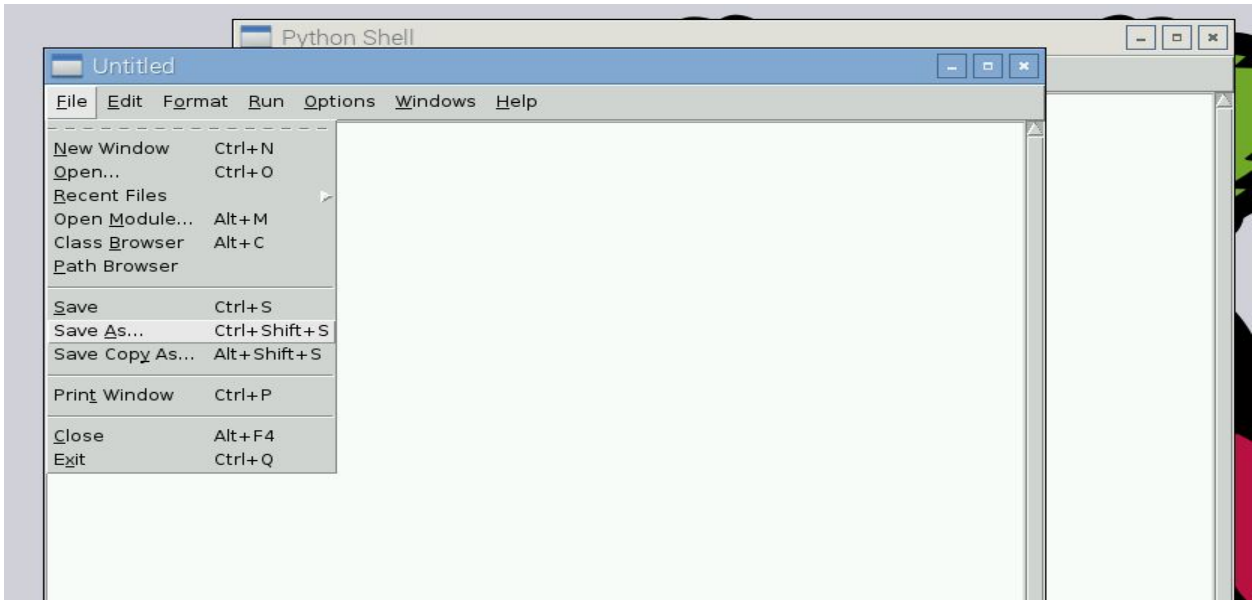
The Editor:

This is important. This is where you will be writing your code for your Minecraft project. Here you will be able to save your python code, and then run it.

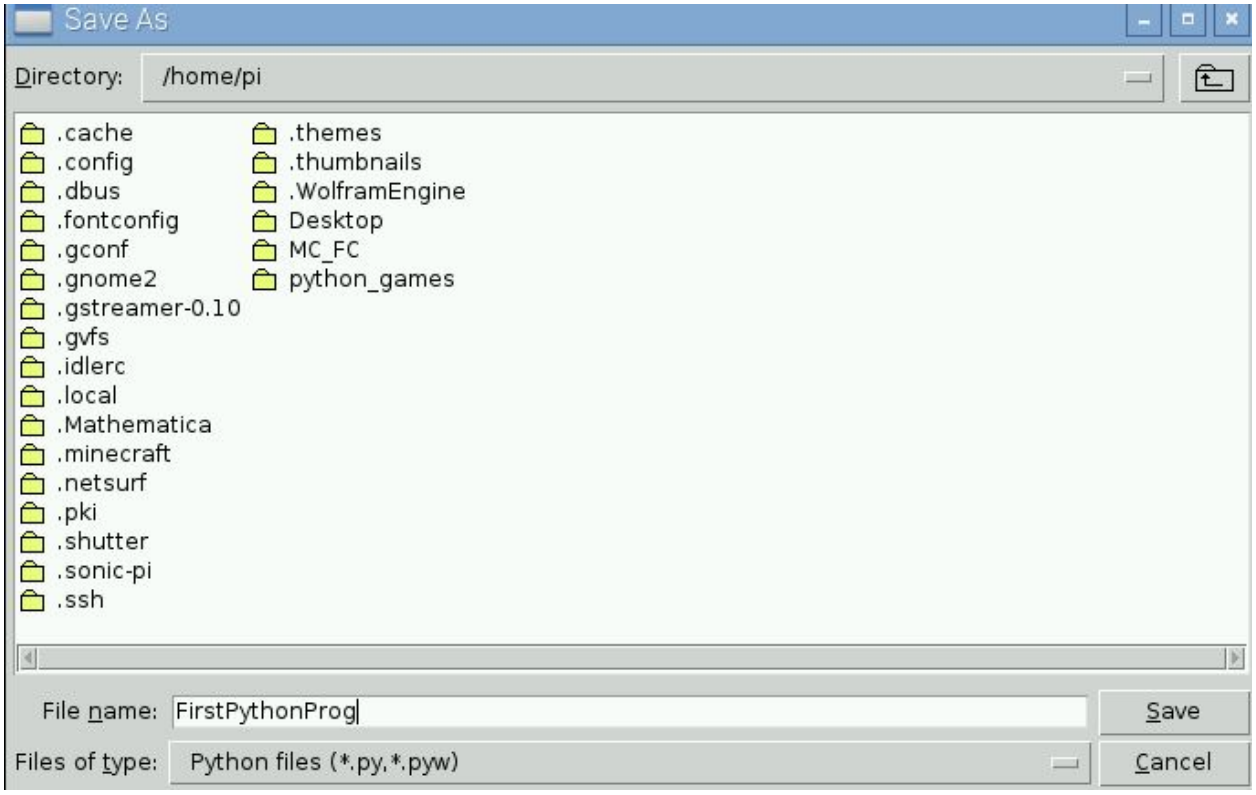
To open the editor go to: File > New Window



Save the file (important) go to: File> Save As…



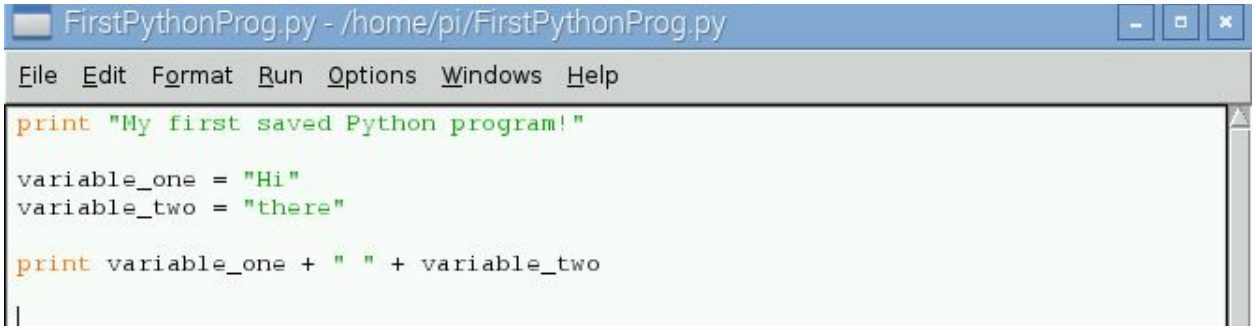
You will see a new window, here you need to save the file and give it a name you will remember!



- The directory you should save to is: */home/pi*
- **Save often by going to:** File > Save! (on the keyboard you can enter ctrl + s)
- If you ever close the Python window out, you can open your file by going to File > Open and selecting your saved file, and then pressing open

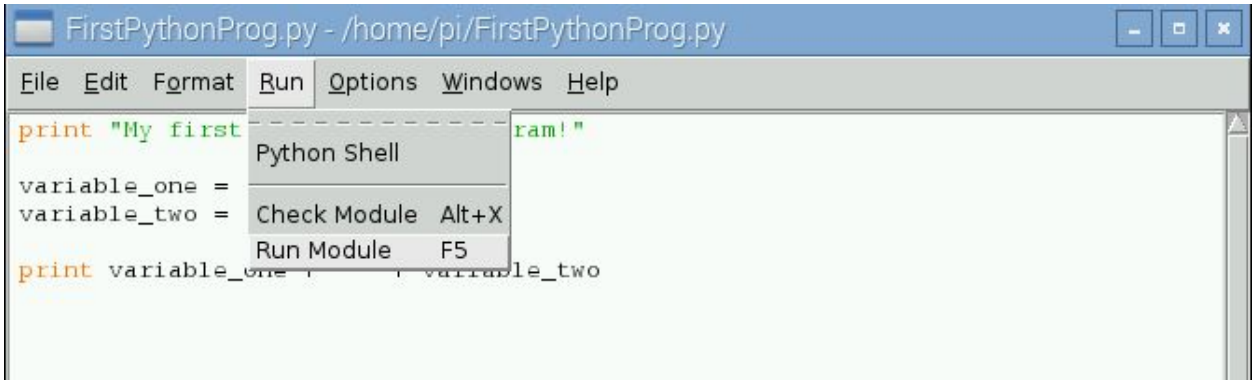
Start off big:

Type this into your new saved editor:



Make sure to save it, then run the program (see how below).

Run a saved Python Program (important)
You can do this by going to Run > Run Module



For Loops:

Codeacademy did not cover *for loops* very well, here is a brief intro:

- They are used when you want to repeat steps a number of times
- For loops run for a fixed amount of time, unlike while loops
- We will use these a lot in Minecraft
    - You will see examples of “nested” for loops in Minecraft. These are essentially for loops within for loops.
- Remember that in computer science numbering sequences start at 0!
- The “range” function is an essential part in for loops. You might also see xrange as well.

Basic Syntax:

```
For i in range (start, end):
    Do something
```

In this case “i” is the variable that is used to hold the value at each step from start to end. Range specifies how many times the loop should run.

Try these:

```
for i in range(0,3):
    print "How many times will this run"
```

How many times do you think this loop will print?
Your output should be:

```
How many times will this run
How many times will this run
How many times will this run
```

What will this one print out?

```
z = 0
for i in range(0,3):
    z = i + z
    print z
```

The output should be:

```
>>>
0
1
3
>>>
```

Why does it print 3 instead of two? This is because when z = 1, i will = 2 therefore adding to 3

Try changing the range values of the for loop, and anything else you want!

These are the basics of using the Python editor!
Try using concepts from Codeacademy with what you just learned!

Helpful things to know and try:

Comments:

A comment is straightforward, it is a piece of text that is ignored by the programing language. It is used to write comments about code.

Python Comments are:

#

The “hashtag”, number sign or pound sign etc.

```
# this line will be ignored when the python program is run
```

\'\'\'text\'\'\'

Three single quotes are multi line comments, you may see these used.

```
'''This is a multi line comment
    and used for explaining long lines of code
'''
```

[ctrl + c]

If the python program ever gets stuck or does not end press this key combination to end the program.

Module

Is a another python file (with the .py extension). We will be using these in Minecraft

Library

Provide a wide array for tools to use in Python. Some tools can be modules!

Import

You will see these at the top of your Python programs. They essentially allow us to use the modules and Libraries that Python and the Minecraft API provide us.

Variable:

We will see these a lot. They are items that hold a piece of data. You saw examples in both codeacademy and from the previous page, try making one yourself.

Definition:

These are important, and allow you to run long lines of code multiple times.

```
def this_is_a_definition(parameter):
    print ("a definition can take a parameter, or a piece of data " + parameter)

#Calling the definition:
this_is_a_definition("HELLO THERE!")
```

The output of the definition:

```
>>>
a definition can take a parameter, or a piece of data HELLO THERE!
>>>
```

Loops:

Loops will repeat many steps and stop when a condition is met.

Use them on repetitive tasks or when you find yourself writing the same lines of code many different times.

Try these:

```
x = 1
while (x < 4):
    x = x + 1
    print x
```

Try this for loop (it's a little different):

```
for x in range (0,4):
    x = x + 1
    print x
```

Try and figure out what each loop is doing.

You can change around values of the code to see different outcomes. Or add another variable.

Loops can get stuck, remember [ctrl + c] to end out of control loops!

Control Statements:

They can test if a condition is met. Also known as if statements

```
x = -1
if x < 0:
    print "x is less than zero"
elif x > 0:
    print "x is greater than 0"
else:
    print "x must be zero"
```

Other Syntax:

== Both sides are equal to one another.

!= Both sides are not equal to one another.

<, > Less than and greater than.

<=, >= Less than or equal to. Greater than or equal to.

and Both sides are true.

not Returns true for false statements and False for true statements.

or One side or the other side is true.