

NNSA HSAP Computational Laboratory 2 - 2026 Update

[1] Using Python in “interactive” mode. Part One.

Open a terminal. To do this you need first to log into the computer. You should see a terminal icon (small grey box) on the left of your screen. Click on it.

Open up python. To this type *python* in the terminal window.

get “prompt” `>>>`

(python is ready for you to enter commands)

Type `1+1`, followed by `<enter>`

what do you get?

Type `print('Hello World!')` followed by `<enter>`

what do you get?

Try some basic arithmetic operations

`40+2`

`43-1`

`6*7`

`84/2`

Notice an asterisk `*` is used for multiplication not a ‘times sign’ (`x`).

Exponentiation: Different computer languages have different format for this.

[EX: 2-1] Which of the following is the correct way to do an exponential in python?

`5^3`

`5**3`

`pow(5,3)`

[EX: 2-2] What are the results of the following in an interactive python window?

a) `type(2)`

b) `type(2.0)`

c) `type('Hello!')`

d) `type('2.0')`

What does the command ‘type’ ask python to tell you?

Explain the difference between a) and b).

Explain the difference between b) and d).

[EX: 2-3] Try this sequence of commands:

`n=7`

`type(n)`

`n=2.0`

`type(n)`

[EX: 2-4] Try these commands:

```
7%3
```

```
8%3
```

```
9%3
```

```
10%3
```

```
11%3
```

```
12%3
```

What is python doing?

In python, an assignment statement creates a new variable and gives it a value.

[EX: 2-5] Try this in your python terminal:

```
n = 17
```

```
print(n)
```

Then try this

```
n+25
```

What happens?

Now try

```
N+25
```

How would you discuss/explain what happened?

[EX: 2-6] You can also assign a variable a string. Try this command sequence:

```
a='Hagrid likes dragons'
```

```
print(a)
```

```
type(a)
```

```
a+25
```

What happens? Why?

```
b=' and Norbert is actually Norberta'
```

```
a+b
```

What happens? Why?

```
print(A)
```

What happens?

Why?

[EX: 2-7] Math interlude! What does python give for

```
16**0.5
```

```
25**0.5
```

```
36**0.5
```

```
49**0.5
```

Explain why it makes sense for the 0.5 power be a square root.

[EX: 2-8] For $a = \text{'Hagrid likes dragons'}$

Do the commands

$a-4$

$a/4$

$a**4$

have any meaning? Why or why not?

In real life we perform different tasks depending on the numerical values of some quantity:

If temperature $T > 60^\circ F$, then wear sweater.

If gas gauge $G < 1/4$ full, then buy gas.

If exam grade $S < 80$, then stay home and study more.

Similarly when we write programs, we need the ability to tell the computer to perform different tasks depending on values of some variable.

We first need to understand how python does ‘logical statements’, that is when does python think something is true or false?

[EX: 2-9] Try giving python these commands:

```
5==5
```

```
5==6
```

Can you explain what python is using ‘==’ for?

Why not use ‘=’?

[EX: 2-10] Try giving python these commands:

```
5 > 6
```

```
5 < 6
```

```
5 >= 5
```

```
17 <= 17
```

```
23 != 2
```

```
23 != 23
```

Can you explain what python is using ‘!=’ for?

[EX: 2-11] Now try this:

(Note 1: hit the ‘enter’ key after each line.)

(Note 2: Try it with and without indenting before the ‘print’ statement.)

```
x=5
```

```
if x>0:
```

```
    print(' x is positive ')
```

Why do you think python might use indentation in such an “if” statement?

[EX: 2-12] Describe what happens with this sequences of python commands:

```
x=-3
```

```
if x>0:
```

```
    print(' x is positive ')
```

[EX: 2-13] Let's write something that begins to look like a piece of code!

```
x = int(input("Please enter an integer: "))
if x%2 ==0:
    print('x is even')
else:
    print('x is odd')
```

Try playing around with indenting and not indenting.

In python, indentation tells the computer when 'if statements' begin and end. This will also be true of loops.