idle-

up- PREVIOUS (alt p for Windows)
down- NEXT (alt n for Windows)
tab - completes request (to the best of the computer's abilities, double tab it will give you options)
to go back or undo a command by one at a time - cd ..

terminal -

python3 to enter
>>> is the beginning of the code

'a' + 'c' = 'ac'
'5' + '7' = 57
5 + 7 = 12
'5' + 7 = error
'a' + c = error

+ with ' is just combining those two items, without them is literal addition

quit() will take you out, python3 will come back

autocomplete is tab after it has learned something (auto type)

—

pwd is where is this program? (print working directory)
cd is to "change direct[ory]" - example: cd Desktop
and then check that it moved by typing pwd

ls (LS) gives you a rundown of what's on the desktop

—

to run a file, do "python3 filename"

—

>>> interactive !

—

spaces need a \ before the space because backslash is to denote a special character - you can also do this if you want an apostrophe in a sentence, ex: "I don\'t want bananas" since the character now reads as "special"

—

"print" means to print a statement - like "print(a)" will print
a
And - print('a\ta') is a, special character (backslash), t is for tab, a and will therefore print like this:
a a
and then print('a\na') will print as
a
a
because n is next!

—

touch as a command creates a new file

—

touch a b will create a file called "a" and a file called "b", touch a \ b will create a file called "a b"

—

variable name = input(“question?\n:”)
print(variable name) enter becomes whatever they answered that question with

—

variable = “words”
second_variable = “different words”
variable = variable + second_variable
print(variable)
wordsdifferentwords

—

on Atom, creating a file:
line 1: #whatever_the_name_is.py
line2: print(“Words”)
line3: thing1 = input(“Question? >”)
line4: print(“Words” + thing1)

how this would read:
Hello there! What is your name? > (user says “Rosie”)
then it prints “Words Rosie”

—

on Python you can enter >>>
and then go
movie1 = ‘Hot Rod’

name1 = ‘Bob’
movie2 = ‘High School Musical’
name2 = ‘Larson’

f strings can go as follows:
print(f”{name1} likes to watch {movie1}”)
And then it will print
Bob likes to watch Hot Rod

—

another way could be, with those variables defined,

out_string = ‘’
out_string = out_string + movie1
etc

—