**Name: Session:**

**Programming I**

**Lab Exercise 9/1/2021**

**Getting Started: Python and IDLE**  
**Using IDLE**

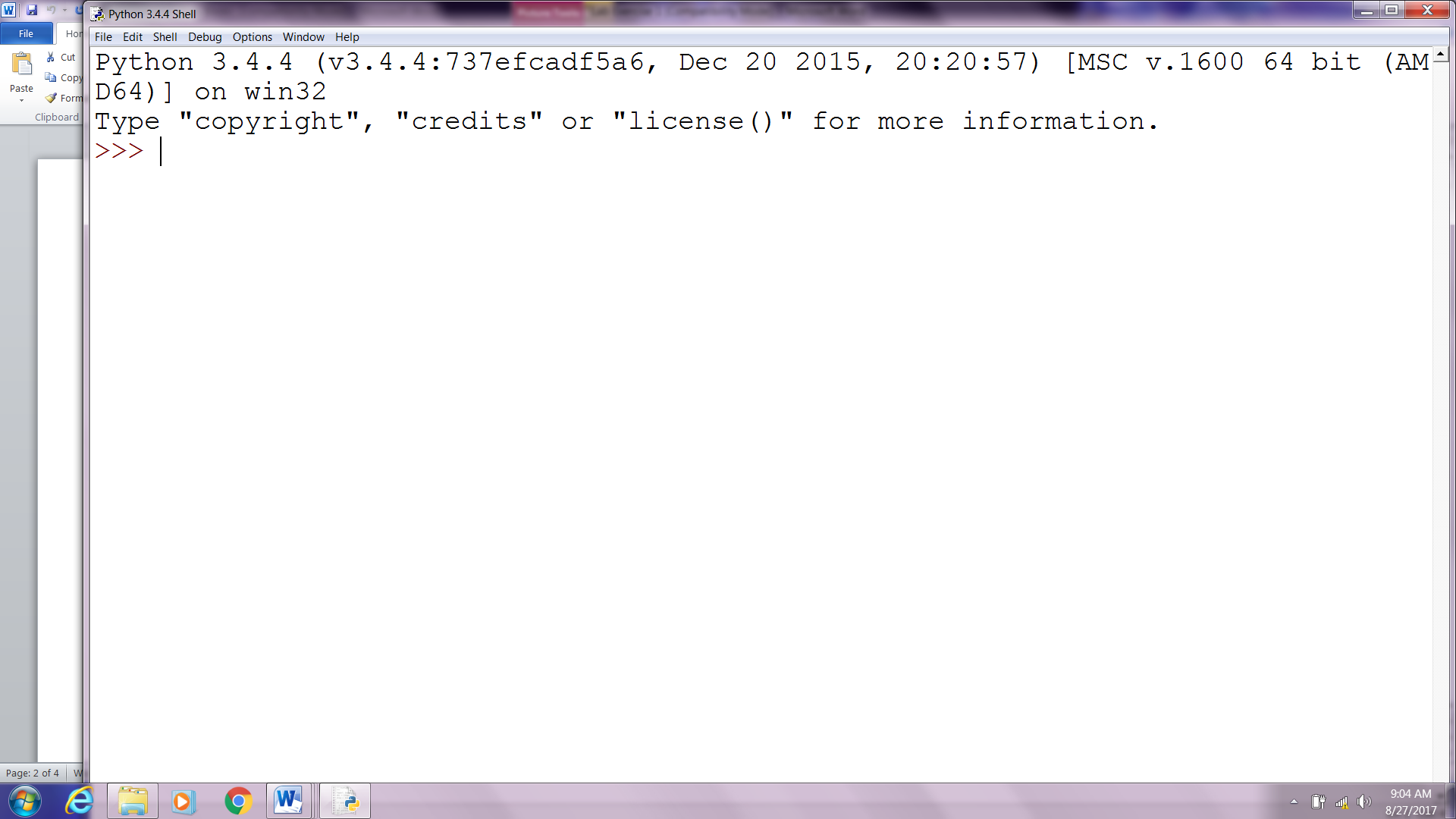
IDLE is the standard Python development environment. Its name is an acronym of "**I**ntegrated **D**eve**L**opment **E**nvironment". It works well on both Unix and Windows platforms.

It has a Python shell window, which gives you access to the Python interactive mode. It also has a file editor that lets you create and edit existing Python source files.

During the following discussion of IDLE's features, instead of passively reading along, you should start IDLE and try to replicate the screenshots.

**Interactive Python shell**

When you start up IDLE, a window with an interactive Python shell will pop up:



You can type Python code directly into this shell, at the '>>>' prompt. Whenever you enter a complete code fragment, it will be executed. For instance, typing:

|  |  |
| --- | --- |
| >>> print ("hello world") |  |

and pressing ENTER, will cause the following to be displayed:

|  |  |
| --- | --- |
| hello world |  |

Using the IDLE shell, evaluate the following expressions:

Integer Expressions

1. 2 + 4 \_\_\_\_\_\_\_
2. 6 – 4 \_\_\_\_\_\_\_
3. 6 \* 3 \_\_\_\_\_\_\_
4. 6 / 3 \_\_\_\_\_\_\_
5. 6 % 3 \_\_\_\_\_\_\_
6. 6 // 3 \_\_\_\_\_\_\_
7. 3\*\*2 \_\_\_\_\_\_\_

Floating Point Expressions

1. 2.0 + 4.0 \_\_\_\_\_\_\_
2. 6.0 - 4.0 \_\_\_\_\_\_\_
3. 6.0 \* 3.0 \_\_\_\_\_\_\_
4. 6.0 / 3.0 \_\_\_\_\_\_\_
5. 6.0 % 3.0 \_\_\_\_\_\_\_
6. 6.0 // 3.0 \_\_\_\_\_\_\_
7. 3.0\*\*2.0 \_\_\_\_\_\_\_

# mixing data types in expressions

# mixed type expressions are "converted up"

# converted up means to take the data type with the greater storage

# float has greater storage (8 bytes) than a regular int (4 bytes)

1. 2 + 4.0 \_\_\_\_\_\_\_
2. 6 - 4.0 \_\_\_\_\_\_\_
3. 6 \* 3.0 \_\_\_\_\_\_\_
4. 6 / 3.0 \_\_\_\_\_\_\_
5. 6 % 3.0 \_\_\_\_\_\_\_
6. 6 // 3.0 \_\_\_\_\_\_\_
7. 3\*\*2.0 \_\_\_\_\_\_\_

# these are Boolean expressions which result in a value of

# true or false

# Note that Python stores true as integer 1, and false as integer 0

# but outputs 'true' or 'false' from print statements

1. 7 > 10 \_\_\_\_\_\_\_
2. 4 < 16 \_\_\_\_\_\_\_
3. 4 == 4 \_\_\_\_\_\_\_
4. 4 <= 4 \_\_\_\_\_\_\_
5. 4 >= 4 \_\_\_\_\_\_\_
6. 4 != 4 \_\_\_\_\_\_\_
7. 4 <> 4 \_\_\_\_\_\_\_

Now let’s try some examples with strings of characters:

1. ‘hello’ \* 3 \_\_\_\_\_\_\_\_\_\_\_\_\_\_
2. ‘hello’ + “world” \_\_\_\_\_\_\_\_\_\_\_\_\_\_
3. ‘hello’ + ‘ ‘ + ‘world’ \_\_\_\_\_\_\_\_\_\_\_\_\_\_
4. ‘hello’ + ‘\n‘ + ‘world’ \_\_\_\_\_\_\_\_\_\_\_\_\_\_
5. print 'hello' + '\n' + 'world' \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Addition (+), subtraction (-), multiplication (\*), division (/), floor division (//), modulo (%) and power (\*\*) operators are built into the Python language. This means you can use them right away. If you want to use a square root in your calculation, you can either raise something to the power of 0.5 or you can *import* the *math* module. Do not worry about what it means right now, we will cover this later during the course. Below are two examples of square root calculation:

|  |  |
| --- | --- |
| >>> 16\*\*0.5 4.0 >>> import math >>> math.sqrt(16) 4.0 |  |

The math module allows you to do a number of useful operations:

|  |  |
| --- | --- |
| >>> math.log(16, 2) 4.0 >>> math.cos( 0 ) 1.0 |  |

Note that you only need to execute the import command once after you start IDLE; however you will need to execute it again if you *restart* the shell, as restarting resets everything back to how it was when you opened IDLE. Don't worry too much about this right now; we'll cover it more in depth soon!

**Exercise**

*(this is just for practice, solutions will not be graded or collected in class)*  
Use IDLE to calculate:

1. 6+4\*10
2. (6+4)\*10 (Compare this to #1, and note that Python uses parentheses just like you would in normal math to determine order of operations!)
3. 23.0 to the 5th power
4. Positive root of the following equation:  
   34x2 + 68x - 510 = 0  
   Recall:  
   ax2 + bx + c = 0  
   x1 = ( - b + sqrt ( b\*b - 4\*a\*c ) ) / ( 2\*a)
5. Calculate how many days old you are.
6. Calculate how many weekends you have experienced.

**When you have completed everything on this worksheet, turn it in by placing it into the turn-in folder.**