Last name _____

First name _____

LARSON—MATH 310–CLASSROOM WORKSHEET 02
Getting Started with CoCalc.

1. Create a CoCalc account.

   (a) Start the Chrome browser.

   (b) Go to `https://cocalc.com`

   (c) "Create new account" using **your VCU email address** .

   (d) You should see an existing Project for our class. Click on that.

   (e) Make sure you are in your Home directory (if you work in your Handouts directory, your work could get overwritten).

   (f) Click "New", then "Jupyter Notebook", then call it **310-c02**.

   (g) Make sure you have PYTHON as the *kernel*.

2. Code the following *procedure* and try it with some inputs:

```
def mul(p,q):
    return p*q
```

3. What is the difference between a *procedure* and a *function*?

Coding the Matrix - Simple Expressions 0.5.1

4. Code and run.

```
44+11*4-6/11
```

5. (**Task 0.5.1**) Use Python to find the number of minutes in a week.

6. (**Task 0.5.2**) Use Python to find the remainder of 2304811 divided by 47 without using the modulo operator $\%$. (Hint: Use `//`.)

Comparisons.
You can compare values (strings and numbers, for example) using the operators $==$, $<$, $>$, $<=$, $>=$, and $!=$. (The operator $!=$ is inequality.)

7. Code and run.

```
5 == 4
5 != 4
```

```
4 == 4
4 != 4
```

8. (**Task 0.5.3**) Enter a Boolean expression to test whether the sum of 673 and 909 is divisible by 3.

Assignment statements

9. Code and run.

```
mynum = 4+1
print(mynum)
print(type(mynum))
```

```
mynum = "Brown"
print(mynum)
print(type(mynum))
```

Conditional statements.

10. (**Task 0.5.4**) Assign the value -9 to x and $1/2$ to y. Predict the value of the following expression, then enter it to check your prediction:

```
2**(y+1/2) if x+10<0 else 2**(y-1/2)
```

Sets.

11. Code and run.

```
{1+2, 3, "a"}

```

```
S = {1+2, 3, "a"}
print(S)
#find the cardinality or length of S
len(S)

```

Set membership.

12. Code and run.

```
S={1,2,3}
2 in S

```

```
4 in S

```

```
4 not in S

```

Set comprehensions.
Python provides for expressions called *comprehensions* that let you build collections out of other collections. We will be using comprehensions a lot because they are useful in constructing an expression whose value is a collection, and they mimic traditional mathematical notation.

13. Code and run.

```
{2*x for x in {1,2,3}}

```

Getting your classwork recorded

When you are done, before you leave class...

(a) Click the "Print" menu choice (under "File") and make a pdf of this worksheet.

(b) Send me an email (`clarson@vcu.edu`) with an informative header like "Math 310 - c02 worksheet attached" (so that it will be properly recorded).

(c) Remember to attach today's classroom worksheet!