cis122 Project 3

Work with lists, "iteration", choosing what to do, with some more plotting.

Name your notebook **Project3**.

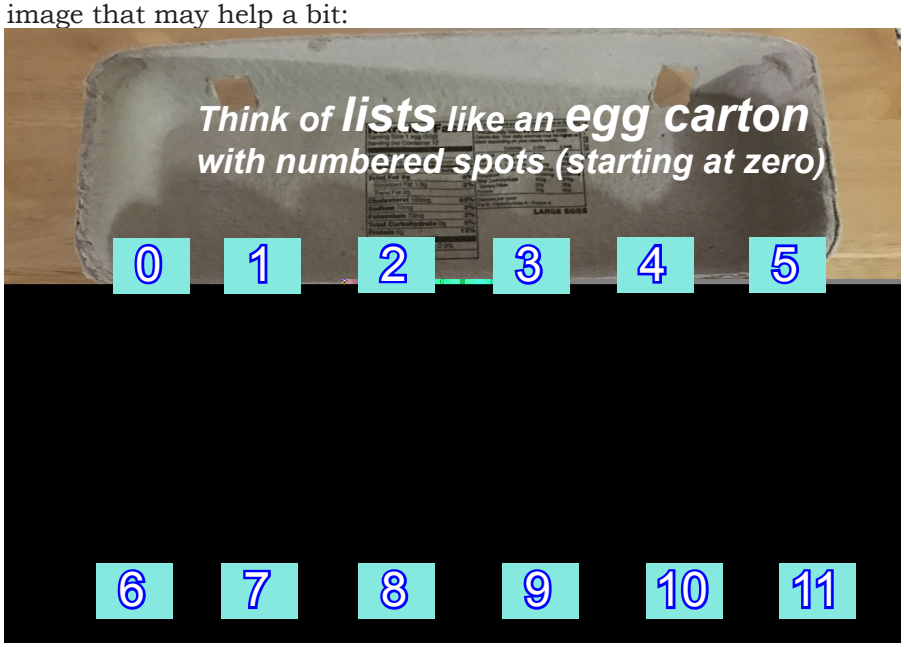
30 points total

When you finish, **save** your notebook (click the "diskette" save icon on the far left),
test your project; make sure each part works correctly,
then go into **Canvas**, **Assignments**, **Project 3**, and **Upload** your **project3.ipynb** file to Canvas.
Click **Submit**.

Download project3.ipynb from Canvas, Assignments - it has some cells for each part of the project.

The cell for part b has some data in a list, so you won't have to re-type the numbers.

Mental image that may help a bit:



Part A

6 points **Get input data, convert to a number as needed**

(1 point for input data, 1 point for conversion if any)
Using the `input( )` function to get data from a user

```
my_major = input("Type your major: ")
Type your major: # (Now the person using the program types something like MATH)
```

Converting typed data (str) to a whole number (int)

```
my_answer = input("How many laptop computers do you own? ")
print(type(my_answer))

# convert to an int (whole number)
computers = int(my_answer)
print(type(computers))
```

```
# User sees
How many computers do you own? 1

print("You own", computers, "laptops")
<class str>  # from print(type(my_answer))
<class int>  # from print(type(computers))
You own 1 laptops
```

Get the **name** of a person (it is a **string** so no conversion needed)
Get the person's **age** and **convert** the answer to an **int** (whole number)
Get the **price** of an item and **convert** it to a **float** (decimal number)

Print the person's **name** and **age**
Print how old the person **will be 3 years from now**
Print the **total due** after adding a **10% sales tax** to the price

Hint
Be sure to convert input strings into either float or int values when you get number answers.

Part B Work with static lists; iterate

6 points

```
# Official low daily temperatures, Eugene, 2017 through Jan 19

low_temperatures = [32, 29, 24, 21, 13, 13, 24, 28, 35, 34,
                    29, 26, 24, 21, 21, 24, 27, 45, 40]
```

a) Iterate through the list displaying each temperature on a separate line
1 point (iterate)

b) Iterate again, but add a comment 'Freeze' for each day the temperature is 32 or lower.
2 points (iterate, Freeze)

c) Iterate again, adding a comment "Coldest" on the coldest day or days and adding a comment "Warmest" on the warmest day or days
3 points (iterate, Coldest, Warmest)

Hints
The **min( )** and **max( )** functions can help you find the coldest and warmest temperatures in your list.
if and **else** can help you compare a temperature with warmest or coldest.
for is great for iterating through the list.

Part c) Modify some values in a static list – work with index values to modify the contents of your list

6 points
Use index values to **modify** values of **individual items** in a list.
Why are we setting limits on what you can do to clean up our list? Later, we will be reading files or web pages to gather lots of data; some of it will need fixing.
If you just type corrections to the data and fix, say 20, bad names but then need to rerun your program, you will have to type all 20 fixes yet again.
Instead, using your **program** to make fixes is a better idea. So that is what we are practicing in this exercise.

```
#            0             1          2         3           4
cities = ['Portland', 'Salen', 'Eugene', 'Medford', 'Beaverton']
```
Do NOT directly MODIFY the line above.

Instead write a Python statement to modify the item with 'Salen'; update it to have the value 'Salem'
2 points

Write Python to find the length (number of items) in the cities list.
1 point

Write Python to change the last city in the list to 'Ashland' instead of 'Beaverton'.
2 points

Iterate through the list, printing each city from the modified list, one city per line
1 point

Part d) Append data items to a list.
6 points

```
# Some sample code for appending data to a list
my_list = [ ] # start with an empty list

animal_1 = 'Elephant'
my_list.append(animal_1) # "Append" (add on to the end of) my_list

animal_2 = 'Giraffe'
my_list.append(animal_2)

print(my_list)  # prints['Elephant', 'Giraffe']
```

Create an empty list to hold your 6 favorite kinds of trees.
Ask a user to **input** her favorite kinds of **trees**.
Append each answer to the list.

When you have the favorites, iterate through the list to print each on a separate line.

1 point	well-named empty list
1 point	**for** loop that runs **6** times :
1 point	call **input( )** to get tree typed in by user
2 points	**append** tree to your list
1 iterate	print each tree in list on a separate line

Part e) Iteration with selective action for several states

6 points
Select a group of states from a `states_list` such as `'OK'`, `'MT'`
iterate through the list, adding a comment `'great place to visit'` to the selected states (OK and MT in this example)

Hint:

Iterate using a **for** loop, Select using something like `if state in ['OK', 'MT']:`

```
states_list = ['AK', 'AZ', 'CA', 'CO', 'HI', 'MI', 'MN', 'MT',
               'ND', 'NM', 'NV', 'OH', 'OK', 'OR', 'SD', 'UT',
               'WA', 'WI', 'WY']
```

Iterate through this states_list,
If the state is any one of `AZ, CA, UT` or `NV`, create a comment 'great place to visit'
Print each state along with a comment when needed, each state and its comment on separate lines.
Sample print

```
AK
AZ great place to visit
CA great place to visit
CO
```

1 point, iterate
3 points if to check for any of the states needing special attention
2 points print state code and when needed, a comment

Bonus

2 points
Display this data as a **Pie Chart**
One year for University of Prineville

```
Tuition       4,675.00
Books           432.75
Room & Board  9,674.00
Misc.         1,500.00
```

GTF challenge Your GTF may also offer a 1 or 2 point challenge.