

You will find a private GitHub repo called `<github-username>-hw` where you will submit all your homework assignments. Clone this repo and create a `hw2` directory inside. Add this directory to the repo using `$ git add hw2`. All your code should appear in a file called `hw2.py` that lives inside the `hw2` directory. Make sure to add `hw2.py` to the repo and commit your changes with `$ git commit -a -m "good log message"`.

**Question 1** (5 points). *Without using the python interpreter, but with the use of documentation, what does `l` equal after all the operations are performed? Verify your answer on the computer. Were you right? Give your guess and whether you were right in a comment (i.e., a line starting with `#`) in `hw2.py`.*

```
>>> l = [list(range(k+1)) for k in range(3)]
>>> l2 = l[1]
>>> l2.append(2)
>>> l3 = sorted(l2,reverse=True)
>>> l[0] = l3
>>> l3.remove(0)
>>> l[0].reverse()
>>> l3.insert(0,0)
```

**Question 2** (5 points). *Write a function called `find_item(item, iterable, key=lambda x: x)` that searches `iterable` for `item`. The function takes an optional third parameter `key`, which is a function. By default, `key` is the identify function, but you can use it to transform an object in `iterable` before comparing it against `item`.*

```
>>> l = [('brent', 38), ('courtney', 40), ('oscar', 5), ('george', 1)]
>>> find_item('george', l, 1)
('george', 1)
>>> find_item('courtney', l, key=operator.itemgetter(0))
('courtney', 40)
>>> find_item(5, l, key=operator.itemgetter(1))
('oscar', 5)
```

**Question 3** (5 points). *Write a class called `PieChart` that represents a traditional pie chart. Each slice in the pie chart is a pair (`label, amount`). You should store the items in an instance variable called `slices`. You may also elect to have an instance variable called `total` that stores the total amount of things labelled in the chart. You will find that reading <https://mkaz.com/2012/10/10/python-string-format/> is helpful when writing the `__str__` method.*

```
1 class PieChart:
2
3     def __init__(self):
4
5     def add_slice(self, label, amount):
6
7     def percentage(self, label):
8
9     def __str__(self):
```

```
>>> p = PieChart()
>>> p.add_slice("Entertainment", 10000)
>>> print(p)
Entertainment: 10000 (1.00)
>>> p.add_slice("Travel", 5000)
>>> p.add_slice("Food", 20000)
>>> print(p)
Entertainment: 10000 (0.29)
Travel: 5000 (0.14)
Food: 20000 (0.57)
```