

Handout for Session 7 (Solutions Only)

1. For loops and dictionaries

Q1: Given the following dictionaries containing word counts (total and current), use a for loop to iterate through the dictionary current and add the counts to the dictionary total. (If the word is not found in total, you have to first initialize the value in total to zero before adding.)

```
[4]: total={'happy':51,'cheap':30}
     current={'happy':2,'amazing':1,'price':2}
```

```
[5]: for key in current:
     if key not in total:
         total[key]=0
     total[key]+=current[key]
     total
```

```
{'happy': 53, 'cheap': 30, 'amazing': 1, 'price': 2}
```

2. Breaking Down Case 7a from Last Session (4 Step Method)

Q2: Apply the above 4 step method to solve case 7b) from last session.

Solution:

Describe: Obtain the count of each domain from the "From" lines of a mail log, and print the counts.

Decompose:

A: Traverse through and filter for the "From" lines (same as in 7a).

B: Obtain the domain name of each line (same as in 7a).

C: Maintain the count of each word using a dictionary (Q6 from last session).

D: Print the dictionary.

Translate:

```
[12]: # C: Maintain the count of each word
     count={'berkeley.edu':1}
     domain='uct.ac.za'
     if domain not in count:
         count[domain]=0
     count[domain]+=1
     count
```

```
{'berkeley.edu': 1, 'uct.ac.za': 1}
```

```
[13]: domain='uct.ac.za'
     if domain not in count:
         count[domain]=0
     count[domain]+=1
     count
```

```
{'berkeley.edu': 1, 'uct.ac.za': 2}
```

```
[14]: # D: Print the dictionary
     count={'berkeley.edu': 1, 'uct.ac.za': 2}
     for domain in sorted(count.keys()):
         print(domain,count[domain])
```

```
berkeley.edu 1
uct.ac.za 2
```

Combine:

```
[15]: filename='mbox-short.txt'
      file=open(filename,'r')
      count={}
      for line in file:
          line=line.rstrip()
          if line.startswith('From:'):
              domain=line.split('@')[1]
              if domain not in count:
                  count[domain]=0
              count[domain]+=1
      for domain in sorted(count.keys()):
          print(domain,count[domain])
```

```
caret.cam.ac.uk 1
gmail.com 1
iupui.edu 8
media.berkeley.edu 4
uct.ac.za 6
umich.edu 7
```

3. Pandas DataFrame Basics

Q3-a: Create the following DataFrame and name it phones.

```
[31]: products=pd.DataFrame({'price':{'iPhone XR':749,'Samsung S9':619,'iPhone 8':599},\
                             'screen size':{'iPhone XR':6.1,'Samsung S9':5.8,'iPhone 8':4.7}})
      products
```

| | price | screen size |
|------------|-------|-------------|
| Samsung S9 | 619 | 5.8 |
| iPhone 8 | 599 | 4.7 |
| iPhone XR | 749 | 6.1 |

Q3-b: Sort the columns in descending order by screen size.

```
[32]: products.sort_values(by='screen size',ascending=False)
```

| | price | screen size |
|------------|-------|-------------|
| iPhone XR | 749 | 6.1 |
| Samsung S9 | 619 | 5.8 |
| iPhone 8 | 599 | 4.7 |

Q3-c: Obtain only the first two rows of the DataFrame (after sorting by screen size).

```
[33]: products.sort_values(by='screen size',ascending=False).head(2)
```

| | price | screen size |
|------------|-------|-------------|
| iPhone XR | 749 | 6.1 |
| Samsung S9 | 619 | 5.8 |

Q3-d: Create a scatter plot where x axis is screen size and y axis is price.

```
[34]: import matplotlib.pyplot as plt
      products.plot(x='screen size',y='price',kind='scatter')
      plt.show()
```

