

Python_Dictionary_Project_Hurricane Analysis

January 5, 2022

1 Hurricane Analysis

Overview This project is slightly different than others you have encountered thus far on Codecademy. Instead of a step-by-step tutorial, this project contains a series of open-ended requirements which describe the project you'll be building. There are many possible ways to correctly fulfill all of these requirements, and you should expect to use the internet, Codecademy, and other resources when you encounter a problem that you cannot easily solve.

Project Goals You will work to write several functions that organize and manipulate data about Category 5 Hurricanes, the strongest hurricanes as rated by their wind speed. Each one of these functions will use a number of parameters, conditionals, lists, dictionaries, string manipulation, and return statements.

```
[19]: # names of hurricanes
names = ['Cuba I', 'San Felipe II Okeechobee', 'Bahamas', 'Cuba II',
        ↪ 'CubaBrownsville', 'Tampico', 'Labor Day', 'New England', 'Carol', 'Janet',
        ↪ 'Carla', 'Hattie', 'Beulah', 'Camille', 'Edith', 'Anita', 'David', 'Allen',
        ↪ 'Gilbert', 'Hugo', 'Andrew', 'Mitch', 'Isabel', 'Ivan', 'Emily', 'Katrina',
        ↪ 'Rita', 'Wilma', 'Dean', 'Felix', 'Matthew', 'Irma', 'Maria', 'Michael']

# months of hurricanes
months = ['October', 'September', 'September', 'November', 'August',
        ↪ 'September', 'September', 'September', 'September', 'September',
        ↪ 'September', 'October', 'September', 'August', 'September', 'September',
        ↪ 'August', 'August', 'September', 'September', 'August', 'October',
        ↪ 'September', 'September', 'July', 'August', 'September', 'October',
        ↪ 'August', 'September', 'October', 'September', 'September', 'October']

# years of hurricanes
years = [1924, 1928, 1932, 1932, 1933, 1933, 1935, 1938, 1953, 1955, 1961,
        ↪ 1961, 1967, 1969, 1971, 1977, 1979, 1980, 1988, 1989, 1992, 1998, 2003,
        ↪ 2004, 2005, 2005, 2005, 2005, 2007, 2007, 2016, 2017, 2017, 2018]

# maximum sustained winds (mph) of hurricanes
max_sustained_winds = [165, 160, 160, 175, 160, 160, 185, 160, 160, 175, 175,
        ↪ 160, 160, 175, 160, 175, 175, 190, 185, 160, 175, 180, 165, 165, 160, 175,
        ↪ 180, 185, 175, 175, 165, 180, 175, 160]
```

```

# areas affected by each hurricane
areas_affected = [['Central America', 'Mexico', 'Cuba', 'Florida', 'The
↳Bahamas'], ['Lesser Antilles', 'The Bahamas', 'United States East Coast',
↳'Atlantic Canada'], ['The Bahamas', 'Northeastern United States'], ['Lesser
↳Antilles', 'Jamaica', 'Cayman Islands', 'Cuba', 'The Bahamas', 'Bermuda'],
↳['The Bahamas', 'Cuba', 'Florida', 'Texas', 'Tamaulipas'], ['Jamaica',
↳'Yucatn Peninsula'], ['The Bahamas', 'Florida', 'Georgia', 'The Carolinas',
↳'Virginia'], ['Southeastern United States', 'Northeastern United States',
↳'Southwestern Quebec'], ['Bermuda', 'New England', 'Atlantic Canada'],
↳['Lesser Antilles', 'Central America'], ['Texas', 'Louisiana', 'Midwestern
↳United States'], ['Central America'], ['The Caribbean', 'Mexico', 'Texas'],
↳['Cuba', 'United States Gulf Coast'], ['The Caribbean', 'Central America',
↳'Mexico', 'United States Gulf Coast'], ['Mexico'], ['The Caribbean', 'United
↳States East coast'], ['The Caribbean', 'Yucatn Peninsula', 'Mexico', 'South
↳Texas'], ['Jamaica', 'Venezuela', 'Central America', 'Hispaniola'],
↳'Mexico'], ['The Caribbean', 'United States East Coast'], ['The Bahamas',
↳'Florida', 'United States Gulf Coast'], ['Central America', 'Yucatn
↳Peninsula', 'South Florida'], ['Greater Antilles', 'Bahamas', 'Eastern
↳United States', 'Ontario'], ['The Caribbean', 'Venezuela', 'United States
↳Gulf Coast'], ['Windward Islands', 'Jamaica', 'Mexico', 'Texas'],
↳['Bahamas', 'United States Gulf Coast'], ['Cuba', 'United States Gulf
↳Coast'], ['Greater Antilles', 'Central America', 'Florida'], ['The
↳Caribbean', 'Central America'], ['Nicaragua', 'Honduras'], ['Antilles',
↳'Venezuela', 'Colombia', 'United States East Coast', 'Atlantic Canada'],
↳['Cape Verde', 'The Caribbean', 'British Virgin Islands', 'U.S. Virgin
↳Islands', 'Cuba', 'Florida'], ['Lesser Antilles', 'Virgin Islands', 'Puerto
↳Rico', 'Dominican Republic', 'Turks and Caicos Islands'], ['Central
↳America', 'United States Gulf Coast (especially Florida Panhandle)']]

# damages (USD($)) of hurricanes
damages = ['Damages not recorded', '100M', 'Damages not recorded', '40M', '27.
↳9M', '5M', 'Damages not recorded', '306M', '2M', '65.8M', '326M', '60.3M',
↳'208M', '1.42B', '25.4M', 'Damages not recorded', '1.54B', '1.24B', '7.1B',
↳'10B', '26.5B', '6.2B', '5.37B', '23.3B', '1.01B', '125B', '12B', '29.4B',
↳'1.76B', '720M', '15.1B', '64.8B', '91.6B', '25.1B']

# deaths for each hurricane
deaths =
↳[90,4000,16,3103,179,184,408,682,5,1023,43,319,688,259,37,11,2068,269,318,107,65,19325,51,1

```

Question 1 In order to complete this project, you should have completed the Loops and Dictionaries sections of the Learn Python 3 Course. This content is also covered in the Data Scientist Career Path.

Question 2 Hurricanes, also known as cyclones or typhoons, are one of the most powerful forces of nature on Earth. Due to climate change caused by human activity, the number and intensity of hurricanes has risen, calling for better preparation by the many communities that are devastated by them. As a concerned environmentalist, you want to look at data about the most powerful

hurricanes that have occurred.

Begin by looking at the damages list. The list contains strings representing the total cost in USD(\$) caused by 34 category 5 hurricanes (wind speeds 157 mph (252 km/h)) in the Atlantic region. For some of the hurricanes, damage data was not recorded (“Damages not recorded”), while the rest are written in the format “Prefix-B/M”, where B stands for billions (1000000000) and M stands for millions (1000000).

Write a function that returns a new list of updated damages where the recorded data is converted to float values and the missing data is retained as “Damages not recorded”.

Test your function with the data stored in damages.

```
[20]: #Question 1-2
conversion = {"M": 1000000,
              "B": 1000000000}

def convert_str_float(damages):
    float_damages=[float(i.replace('M','*1000000').replace('B','*1000000000').
    ↳split('*')[0])*float(i.replace('M','*1000000').replace('B','*1000000000').
    ↳split('*')[1]) if i !='Damages not recorded' else i for i in damages]
    return float_damages
float_damage=convert_str_float(damages)
```

Question 3 Additional data collected on the 34 strongest Atlantic hurricanes are provided in a series of lists. The data includes:

names: names of the hurricanes months: months in which the hurricanes occurred years: years in which the hurricanes occurred max_sustained_winds: maximum sustained winds (miles per hour) of the hurricanes areas_affected: list of different areas affected by each of the hurricanes deaths: total number of deaths caused by each of the hurricanes The data is organized such that the data at each index, from 0 to 33, corresponds to the same hurricane.

For example, names[0] yields the “Cuba I” hurricane, which occurred in months[0] (October) years[0] (1924).

Write a function that constructs a dictionary made out of the lists, where the keys of the dictionary are the names of the hurricanes, and the values are dictionaries themselves containing a key for each piece of data (Name, Month, Year, Max Sustained Wind, Areas Affected, Damage, Death) about the hurricane.

Thus the key “Cuba I” would have the value: {'Name': 'Cuba I', 'Month': 'October', 'Year': 1924, 'Max Sustained Wind': 165, 'Areas Affected': ['Central America', 'Mexico', 'Cuba', 'Florida', 'The Bahamas'], 'Damage': 'Damages not recorded', 'Deaths': 90}.

Test your function on the lists of data provided.

```
[21]: #Question 3
def
↳hurricane_dic(names,months,years,max_sustained_winds,areas_affected,damages,deaths):
↳
    hurricane={}

```

```

for i in range(len(names)):
    hurricane[names[i]]={'Name':names[i], 'Month': months[i], 'Year': years[i],
→'Max Sustained Wind': max_sustained_winds[i], 'Areas Affected':
→areas_affected[i], 'Damage':damages[i], 'Deaths': deaths[i]}
    return hurricane
hurricane_dic=hurricane_dic(names,months,years,max_sustained_winds,areas_affected,float_damage
print(hurricane_dic)

```

```

{'Cuba I': {'Name': 'Cuba I', 'Month': 'October', 'Year': 1924, 'Max Sustained
Wind': 165, 'Areas Affected': ['Central America', 'Mexico', 'Cuba', 'Florida',
'The Bahamas'], 'Damage': 'Damages not recorded', 'Deaths': 90}, 'San Felipe II
Okeechobee': {'Name': 'San Felipe II Okeechobee', 'Month': 'September', 'Year':
1928, 'Max Sustained Wind': 160, 'Areas Affected': ['Lesser Antilles', 'The
Bahamas', 'United States East Coast', 'Atlantic Canada'], 'Damage': 100000000.0,
'Deaths': 4000}, 'Bahamas': {'Name': 'Bahamas', 'Month': 'September', 'Year':
1932, 'Max Sustained Wind': 160, 'Areas Affected': ['The Bahamas', 'Northeastern
United States'], 'Damage': 'Damages not recorded', 'Deaths': 16}, 'Cuba II':
{'Name': 'Cuba II', 'Month': 'November', 'Year': 1932, 'Max Sustained Wind':
175, 'Areas Affected': ['Lesser Antilles', 'Jamaica', 'Cayman Islands', 'Cuba',
'The Bahamas', 'Bermuda'], 'Damage': 40000000.0, 'Deaths': 3103},
'CubaBrownsville': {'Name': 'CubaBrownsville', 'Month': 'August', 'Year': 1933,
'Max Sustained Wind': 160, 'Areas Affected': ['The Bahamas', 'Cuba', 'Florida',
'Texas', 'Tamaulipas'], 'Damage': 27900000.0, 'Deaths': 179}, 'Tampico':
{'Name': 'Tampico', 'Month': 'September', 'Year': 1933, 'Max Sustained Wind':
160, 'Areas Affected': ['Jamaica', 'Yucatn Peninsula'], 'Damage': 5000000.0,
'Deaths': 184}, 'Labor Day': {'Name': 'Labor Day', 'Month': 'September', 'Year':
1935, 'Max Sustained Wind': 185, 'Areas Affected': ['The Bahamas', 'Florida',
'Georgia', 'The Carolinas', 'Virginia'], 'Damage': 'Damages not recorded',
'Deaths': 408}, 'New England': {'Name': 'New England', 'Month': 'September',
'Year': 1938, 'Max Sustained Wind': 160, 'Areas Affected': ['Southeastern United
States', 'Northeastern United States', 'Southwestern Quebec'], 'Damage':
306000000.0, 'Deaths': 682}, 'Carol': {'Name': 'Carol', 'Month': 'September',
'Year': 1953, 'Max Sustained Wind': 160, 'Areas Affected': ['Bermuda', 'New
England', 'Atlantic Canada'], 'Damage': 2000000.0, 'Deaths': 5}, 'Janet':
{'Name': 'Janet', 'Month': 'September', 'Year': 1955, 'Max Sustained Wind': 175,
'Areas Affected': ['Lesser Antilles', 'Central America'], 'Damage': 65800000.0,
'Deaths': 1023}, 'Carla': {'Name': 'Carla', 'Month': 'September', 'Year': 1961,
'Max Sustained Wind': 175, 'Areas Affected': ['Texas', 'Louisiana', 'Midwestern
United States'], 'Damage': 326000000.0, 'Deaths': 43}, 'Hattie': {'Name':
'Hattie', 'Month': 'October', 'Year': 1961, 'Max Sustained Wind': 160, 'Areas
Affected': ['Central America'], 'Damage': 60300000.0, 'Deaths': 319}, 'Beulah':
{'Name': 'Beulah', 'Month': 'September', 'Year': 1967, 'Max Sustained Wind':
160, 'Areas Affected': ['The Caribbean', 'Mexico', 'Texas'], 'Damage':
208000000.0, 'Deaths': 688}, 'Camille': {'Name': 'Camille', 'Month': 'August',
'Year': 1969, 'Max Sustained Wind': 175, 'Areas Affected': ['Cuba', 'United
States Gulf Coast'], 'Damage': 1420000000.0, 'Deaths': 259}, 'Edith': {'Name':
'Edith', 'Month': 'September', 'Year': 1971, 'Max Sustained Wind': 160, 'Areas
Affected': ['The Caribbean', 'Central America', 'Mexico', 'United States Gulf

```

Coast'], 'Damage': 25400000.0, 'Deaths': 37}, 'Anita': {'Name': 'Anita', 'Month': 'September', 'Year': 1977, 'Max Sustained Wind': 175, 'Areas Affected': ['Mexico'], 'Damage': 'Damages not recorded', 'Deaths': 11}, 'David': {'Name': 'David', 'Month': 'August', 'Year': 1979, 'Max Sustained Wind': 175, 'Areas Affected': ['The Caribbean', 'United States East coast'], 'Damage': 1540000000.0, 'Deaths': 2068}, 'Allen': {'Name': 'Allen', 'Month': 'August', 'Year': 1980, 'Max Sustained Wind': 190, 'Areas Affected': ['The Caribbean', 'Yucatn Peninsula', 'Mexico', 'South Texas'], 'Damage': 1240000000.0, 'Deaths': 269}, 'Gilbert': {'Name': 'Gilbert', 'Month': 'September', 'Year': 1988, 'Max Sustained Wind': 185, 'Areas Affected': ['Jamaica', 'Venezuela', 'Central America', 'Hispaniola', 'Mexico'], 'Damage': 7100000000.0, 'Deaths': 318}, 'Hugo': {'Name': 'Hugo', 'Month': 'September', 'Year': 1989, 'Max Sustained Wind': 160, 'Areas Affected': ['The Caribbean', 'United States East Coast'], 'Damage': 10000000000.0, 'Deaths': 107}, 'Andrew': {'Name': 'Andrew', 'Month': 'August', 'Year': 1992, 'Max Sustained Wind': 175, 'Areas Affected': ['The Bahamas', 'Florida', 'United States Gulf Coast'], 'Damage': 26500000000.0, 'Deaths': 65}, 'Mitch': {'Name': 'Mitch', 'Month': 'October', 'Year': 1998, 'Max Sustained Wind': 180, 'Areas Affected': ['Central America', 'Yucatn Peninsula', 'South Florida'], 'Damage': 6200000000.0, 'Deaths': 19325}, 'Isabel': {'Name': 'Isabel', 'Month': 'September', 'Year': 2003, 'Max Sustained Wind': 165, 'Areas Affected': ['Greater Antilles', 'Bahamas', 'Eastern United States', 'Ontario'], 'Damage': 5370000000.0, 'Deaths': 51}, 'Ivan': {'Name': 'Ivan', 'Month': 'September', 'Year': 2004, 'Max Sustained Wind': 165, 'Areas Affected': ['The Caribbean', 'Venezuela', 'United States Gulf Coast'], 'Damage': 23300000000.0, 'Deaths': 124}, 'Emily': {'Name': 'Emily', 'Month': 'July', 'Year': 2005, 'Max Sustained Wind': 160, 'Areas Affected': ['Windward Islands', 'Jamaica', 'Mexico', 'Texas'], 'Damage': 1010000000.0, 'Deaths': 17}, 'Katrina': {'Name': 'Katrina', 'Month': 'August', 'Year': 2005, 'Max Sustained Wind': 175, 'Areas Affected': ['Bahamas', 'United States Gulf Coast'], 'Damage': 125000000000.0, 'Deaths': 1836}, 'Rita': {'Name': 'Rita', 'Month': 'September', 'Year': 2005, 'Max Sustained Wind': 180, 'Areas Affected': ['Cuba', 'United States Gulf Coast'], 'Damage': 12000000000.0, 'Deaths': 125}, 'Wilma': {'Name': 'Wilma', 'Month': 'October', 'Year': 2005, 'Max Sustained Wind': 185, 'Areas Affected': ['Greater Antilles', 'Central America', 'Florida'], 'Damage': 29400000000.0, 'Deaths': 87}, 'Dean': {'Name': 'Dean', 'Month': 'August', 'Year': 2007, 'Max Sustained Wind': 175, 'Areas Affected': ['The Caribbean', 'Central America'], 'Damage': 17600000000.0, 'Deaths': 45}, 'Felix': {'Name': 'Felix', 'Month': 'September', 'Year': 2007, 'Max Sustained Wind': 175, 'Areas Affected': ['Nicaragua', 'Honduras'], 'Damage': 720000000.0, 'Deaths': 133}, 'Matthew': {'Name': 'Matthew', 'Month': 'October', 'Year': 2016, 'Max Sustained Wind': 165, 'Areas Affected': ['Antilles', 'Venezuela', 'Colombia', 'United States East Coast', 'Atlantic Canada'], 'Damage': 15100000000.0, 'Deaths': 603}, 'Irma': {'Name': 'Irma', 'Month': 'September', 'Year': 2017, 'Max Sustained Wind': 180, 'Areas Affected': ['Cape Verde', 'The Caribbean', 'British Virgin Islands', 'U.S. Virgin Islands', 'Cuba', 'Florida'], 'Damage': 64800000000.0, 'Deaths': 138}, 'Maria': {'Name': 'Maria', 'Month': 'September', 'Year': 2017, 'Max Sustained Wind': 175, 'Areas Affected': ['Lesser Antilles', 'Virgin Islands', 'Puerto Rico', 'Dominican Republic', 'Turks and Caicos Islands'], 'Damage':

```
91600000000.0, 'Deaths': 3057}, 'Michael': {'Name': 'Michael', 'Month':
'October', 'Year': 2018, 'Max Sustained Wind': 160, 'Areas Affected': ['Central
America', 'United States Gulf Coast (especially Florida Panhandle)'], 'Damage':
25100000000.0, 'Deaths': 74}}
```

Question 4 In addition to organizing the hurricanes in a dictionary with names as the key, you want to be able to organize the hurricanes by year.

Write a function that converts the current dictionary of hurricanes to a new dictionary, where the keys are years and the values are lists containing a dictionary for each hurricane that occurred in that year.

For example, the key 1932 would yield the value: [{‘Name’: ‘Bahamas’, ‘Month’: ‘September’, ‘Year’: 1932, ‘Max Sustained Wind’: 160, ‘Areas Affected’: [‘The Bahamas’, ‘Northeastern United States’], ‘Damage’: ‘Damages not recorded’, ‘Deaths’: 16}, {‘Name’: ‘Cuba II’, ‘Month’: ‘November’, ‘Year’: 1932, ‘Max Sustained Wind’: 175, ‘Areas Affected’: [‘Lesser Antilles’, ‘Jamaica’, ‘Cayman Islands’, ‘Cuba’, ‘The Bahamas’, ‘Bermuda’], ‘Damage’: 40000000.0, ‘Deaths’: 3103}].

Test your function on your hurricane dictionary.

```
[22]: #Question 4
def hurricane_dic_year(hurricane_dic):
    years=[i['Year'] for i in hurricane_dic.values()]
    hurricane_year={}
    for i in years:
        hurricane_year[i]=[]
    for i in range(len(years)):
        hurricane_year[hurricane_dic[names[i]]['Year']] += [hurricane_dic[names[i]]]
    return hurricane_year
hurricane_year=hurricane_dic_year(hurricane_dic)
print(hurricane_year[1932])
```

```
[{'Name': 'Bahamas', 'Month': 'September', 'Year': 1932, 'Max Sustained Wind':
160, 'Areas Affected': ['The Bahamas', 'Northeastern United States'], 'Damage':
'Damages not recorded', 'Deaths': 16}, {'Name': 'Cuba II', 'Month': 'November',
'Year': 1932, 'Max Sustained Wind': 175, 'Areas Affected': ['Lesser Antilles',
'Jamaica', 'Cayman Islands', 'Cuba', 'The Bahamas', 'Bermuda'], 'Damage':
40000000.0, 'Deaths': 3103}]
```

Question 5 You believe that knowing how often each of the areas of the Atlantic are affected by these strong hurricanes is important for making preparations for future hurricanes.

Write a function that counts how often each area is listed as an affected area of a hurricane. Store and return the results in a dictionary where the keys are the affected areas and the values are counts of how many times the areas were affected.

Test your function on your hurricane dictionary.

```
[23]: #Question 5
def affected_area(hurricane_dic):
    areas_affected=[i['Areas Affected'] for i in hurricane_dic.values()]
```

```

areas_affected_comp=[]
for i in range(len(areas_affected)):
    areas_affected_comp+=areas_affected[i]
areas_count={}
for i in areas_affected_comp:
    areas_count[i]=areas_count.get(i,0)+1
return areas_count

affected_area_dic=affected_area(hurricane_dic)
print(affected_area_dic)

```

```

{'Central America': 9, 'Mexico': 7, 'Cuba': 6, 'Florida': 6, 'The Bahamas': 7,
'Lesser Antilles': 4, 'United States East Coast': 3, 'Atlantic Canada': 3,
'Northeastern United States': 2, 'Jamaica': 4, 'Cayman Islands': 1, 'Bermuda':
2, 'Texas': 4, 'Tamaulipas': 1, 'Yucatn Peninsula': 3, 'Georgia': 1, 'The
Carolinas': 1, 'Virginia': 1, 'Southeastern United States': 1, 'Southwestern
Quebec': 1, 'New England': 1, 'Louisiana': 1, 'Midwestern United States': 1,
'The Caribbean': 8, 'United States Gulf Coast': 6, 'United States East coast':
1, 'South Texas': 1, 'Venezuela': 3, 'Hispaniola': 1, 'South Florida': 1,
'Greater Antilles': 2, 'Bahamas': 2, 'Eastern United States': 1, 'Ontario': 1,
'Windward Islands': 1, 'Nicaragua': 1, 'Honduras': 1, 'Antilles': 1, 'Colombia':
1, 'Cape Verde': 1, 'British Virgin Islands': 1, 'U.S. Virgin Islands': 1,
'Virgin Islands': 1, 'Puerto Rico': 1, 'Dominican Republic': 1, 'Turks and
Caicos Islands': 1, 'United States Gulf Coast (especially Florida Panhandle)':
1}

```

Question 6 Write a function that finds the area affected by the most hurricanes, and how often it was hit. Test your function on your affected area dictionary.

```

[24]: #Question 6
def area_most_affected(affected_area_dic):
    return max(affected_area_dic,key=affected_area_dic.get)

area_most_affected=area_most_affected(affected_area_dic)
print(area_most_affected)

```

Central America

Question 7 Write a function that finds the hurricane that caused the greatest number of deaths, and how many deaths it caused.

Test your function on your hurricane dictionary.

```

[25]: #Question 7
def deaths_most_area(hurricane_dic):
    death=[i['Deaths'] for i in hurricane_dic.values()]
    names=hurricane_dic.keys()
    death_dic={key:value for key,value in zip(names,death)}
    return max(death_dic,key=death_dic.get)
print(deaths_most_area(hurricane_dic))

```

Mitch

Question 8 Just as hurricanes are rated by their windspeed, you want to try rating hurricanes based on other metrics.

Write a function that rates hurricanes on a mortality scale according to the following ratings, where the key is the rating and the value is the upper bound of deaths for that rating.

mortality_scale = {0: 0, 1: 100, 2: 500, 3: 1000, 4: 10000} For example, a hurricane with a 1 mortality rating would have resulted in greater than 0 but less than or equal to 100 deaths. A hurricane with a 5 mortality rating would have resulted in greater than 10000 deaths.

Store the hurricanes in a new dictionary where the keys are mortality ratings and the values are lists containing a dictionary for each hurricane that falls into that mortality rating.

Test your function on your hurricane dictionary.

```
[26]: #Question 8
def mortality_scale(hurricane_dic):
    hurricanes_by_mortality = {0: [], 1: [], 2: [], 3: [], 4: [], 5: []}
    for i in list(hurricane_dic):
        if hurricane_dic[i]['Deaths']==0:
            hurricanes_by_mortality[0]+=[i]
        elif 0<hurricane_dic[i]['Deaths']<=100:
            hurricanes_by_mortality[1]+=[i]
        elif 100<hurricane_dic[i]['Deaths']<=500:
            hurricanes_by_mortality[2]+=[i]
        elif 500<hurricane_dic[i]['Deaths']<=1000:
            hurricanes_by_mortality[3]+=[i]
        elif 1000<hurricane_dic[i]['Deaths']<=10000:
            hurricanes_by_mortality[4]+=[i]
        else:
            hurricanes_by_mortality[5]+=[i]
    return hurricanes_by_mortality
mortality_scale_dic=mortality_scale(hurricane_dic)
print(mortality_scale_dic)
```

```
{0: [], 1: ['Cuba I', 'Bahamas', 'Carol', 'Carla', 'Edith', 'Anita', 'Andrew',
'Isabel', 'Emily', 'Wilma', 'Dean', 'Michael'], 2: ['CubaBrownsville',
'Tampico', 'Labor Day', 'Hattie', 'Camille', 'Allen', 'Gilbert', 'Hugo', 'Ivan',
'Rita', 'Felix', 'Irma'], 3: ['New England', 'Beulah', 'Matthew'], 4: ['San
Felipe II Okeechobee', 'Cuba II', 'Janet', 'David', 'Katrina', 'Maria'], 5:
['Mitch']}
```

Question 9 Write a function that finds the hurricane that caused the greatest damage, and how costly it was.

Test your function on your hurricane dictionary.

```
[27]: #Question 9
def deaths_most_area(hurricane_dic):
```



```

damage=[i['Damage'] if i['Damage']!='Damages not recorded' else 0 for i in_
↳hurricane_dic.values()]
names=hurricane_dic.keys()
damage_dic={key:value for key,value in zip(names,damage)}
print(damage_dic)
return max(damage_dic,key=damage_dic.get)
print(deaths_most_area(hurricane_dic))

```

```

{'Cuba I': 0, 'San Felipe II Okeechobee': 100000000.0, 'Bahamas': 0, 'Cuba II':
40000000.0, 'CubaBrownsville': 27900000.0, 'Tampico': 5000000.0, 'Labor Day': 0,
'New England': 306000000.0, 'Carol': 2000000.0, 'Janet': 65800000.0, 'Carla':
326000000.0, 'Hattie': 60300000.0, 'Beulah': 208000000.0, 'Camille':
1420000000.0, 'Edith': 25400000.0, 'Anita': 0, 'David': 1540000000.0, 'Allen':
1240000000.0, 'Gilbert': 7100000000.0, 'Hugo': 10000000000.0, 'Andrew':
26500000000.0, 'Mitch': 6200000000.0, 'Isabel': 5370000000.0, 'Ivan':
23300000000.0, 'Emily': 1010000000.0, 'Katrina': 125000000000.0, 'Rita':
120000000000.0, 'Wilma': 29400000000.0, 'Dean': 1760000000.0, 'Felix':
720000000.0, 'Matthew': 15100000000.0, 'Irma': 64800000000.0, 'Maria':
91600000000.0, 'Michael': 25100000000.0}

```

Katrina

Question 10 Lastly, you want to rate hurricanes according to how much damage they cause.

Write a function that rates hurricanes on a damage scale according to the following ratings, where the key is the rating and the value is the upper bound of damage for that rating.

damage_scale = {0: 0, 1: 100000000, 2: 1000000000, 3: 10000000000, 4: 50000000000} For example, a hurricane with a 1 damage rating would have resulted in damages greater than 0 USD but less than or equal to 100000000 USD. A hurricane with a 5 damage rating would have resulted in damages greater than 50000000000 USD (talk about a lot of money).

Store the hurricanes in a new dictionary where the keys are damage ratings and the values are lists containing a dictionary for each hurricane that falls into that damage rating.

Test your function on your hurricane dictionary.

```

[28]: #Question 10
def damage_scale(hurricane_dic):
    hurricanes_by_damage = {0: [], 1: [], 2: [], 3: [], 4: [], 5: []}
    for i in list(hurricane_dic):
        if (hurricane_dic[i]['Damage']==0) or (hurricane_dic[i]['Damage']=='Damages_
↳not recorded'):
            hurricanes_by_damage[0].append(i)
        elif 0<hurricane_dic[i]['Damage']<=100000000:
            hurricanes_by_damage[1].append(i)
        elif 100000000<hurricane_dic[i]['Damage']<=1000000000:
            hurricanes_by_damage[2].append(i)
        elif 1000000000<hurricane_dic[i]['Damage']<=10000000000:
            hurricanes_by_damage[3].append(i)
        elif 10000000000<hurricane_dic[i]['Damage']<=50000000000:

```

```
        hurricanes_by_damage[4]+=[i]
    else:
        hurricanes_by_damage[5]+=[i]
    return hurricanes_by_damage
damage_scale_dic=damage_scale(hurricane_dic)
print(damage_scale_dic)
```

```
{0: ['Cuba I', 'Bahamas', 'Labor Day', 'Anita'], 1: ['San Felipe II Okeechobee',
'Cuba II', 'CubaBrownsville', 'Tampico', 'Carol', 'Janet', 'Hattie', 'Edith'],
2: ['New England', 'Carla', 'Beulah', 'Felix'], 3: ['Camille', 'David', 'Allen',
'Gilbert', 'Hugo', 'Mitch', 'Isabel', 'Emily', 'Dean'], 4: ['Andrew', 'Ivan',
'Rita', 'Wilma', 'Matthew', 'Michael'], 5: ['Katrina', 'Irma', 'Maria']}
```

[]: