In [1]:
```
import pandas as pd
```

In [2]:
```
df1 = pd.read_csv('respondents.csv')
df2 = pd.read_csv('states.csv')
```

In [4]:
```
df1
```

Out[4]:

	Person_Key	Age_Key	Gender_Key	State_Key	Children	Salary	Opinion_Key
0	1	2	2	9	2	63017	5
1	2	2	2	10	3	100302	1
2	3	2	2	2	0	144043	5
3	4	1	2	2	0	36025	4
4	5	2	1	9	0	97543	3
...	...	...	...	...	...	...	...
394	395	2	2	2	0	60715	3
395	396	2	1	10	2	91760	2
396	397	2	2	1	1	82558	1
397	398	2	1	7	1	84880	5
398	399	2	2	4	2	76933	5

399 rows × 7 columns

In [5]:
```
df2
```

Out[5]:

	State_Key	State_Name
0	1	Arizona
1	2	California
2	3	Florida
3	4	Illinois
4	5	Michigan
5	6	Minnesota
6	7	New York
7	8	Ohio
8	9	Texas
9	10	Virginia

In [6]:
```
result = pd.merge(df1,df2, on=["State_Key", "State_Key"])
```

In [8]:
```
result
```

Out[8]:

	Person_Key	Age_Key	Gender_Key	State_Key	Children	Salary	Opinion_Key	State_Name
0	1	2	2	9	2	63017	5	Texas
1	5	2	1	9	0	97543	3	Texas
2	8	2	2	9	1	101894	4	Texas
3	14	3	2	9	2	67748	3	Texas
4	15	1	2	9	2	47172	3	Texas
...	...	...	...	...	...	...	...	...
394	371	3	1	8	0	64879	3	Ohio
395	373	2	1	8	2	90675	4	Ohio
396	374	2	2	8	2	109417	4	Ohio
397	386	3	1	8	0	80256	3	Ohio
398	387	3	2	8	2	87059	3	Ohio

399 rows × 8 columns

In [9]:
```
result = pd.merge(df1, df2, how='left', on=["State_Key", "State_Key"])
#can specify type of join with how option 'right', 'left', 'outer', 'inner'
```

In [10]:
```
result.head()
```

Out[10]:

	Person_Key	Age_Key	Gender_Key	State_Key	Children	Salary	Opinion_Key	State_Name
0	1	2	2	9	2	63017	5	Texas
1	2	2	2	10	3	100302	1	Virginia
2	3	2	2	2	0	144043	5	California
3	4	1	2	2	0	36025	4	California
4	5	2	1	9	0	97543	3	Texas

In [11]:
```
df3 = pd.read_csv('states2.csv')
```

In [12]:
```
df3
```

Out[12]:

	State_Key	State_Name
0	11	Alabama

	State_Key	State_Name
1	12	Connecticut
2	13	Georgia
3	14	Indiana
4	15	Mississippi
5	16	Montana
6	17	New Jersey
7	18	Oklahoma
8	19	Tennessee
9	20	West Virginia

In [13]:
```
df4 = df2.append(df3, ignore_index=True)
df4
```

Out[13]:

	State_Key	State_Name
0	1	Arizona
1	2	California
2	3	Florida
3	4	Illinois
4	5	Michigan
5	6	Minnesota
6	7	New York
7	8	Ohio
8	9	Texas
9	10	Virginia
10	11	Alabama
11	12	Connecticut
12	13	Georgia
13	14	Indiana
14	15	Mississippi
15	16	Montana
16	17	New Jersey
17	18	Oklahoma
18	19	Tennessee
19	20	West Virginia

In [14]:
```
df5 = pd.read_csv('states3.csv')
df5
```

Out[14]:

	State_Key	State_Capital
0	1	Phoenix
1	2	Sacremento
2	3	Tallahassee
3	4	Springfield
4	5	Lansing
5	6	Saint Paul
6	7	Albany
7	8	Columbus
8	9	Austin
9	10	Richmond

In [15]:
```
df6 = pd.concat([df2,df5])
df6
```

Out[15]:

	State_Key	State_Name	State_Capital
0	1	Arizona	NaN
1	2	California	NaN
2	3	Florida	NaN
3	4	Illinois	NaN
4	5	Michigan	NaN
5	6	Minnesota	NaN
6	7	New York	NaN
7	8	Ohio	NaN
8	9	Texas	NaN
9	10	Virginia	NaN
0	1	NaN	Phoenix
1	2	NaN	Sacremento
2	3	NaN	Tallahassee
3	4	NaN	Springfield
4	5	NaN	Lansing
5	6	NaN	Saint Paul
6	7	NaN	Albany

	State_Key	State_Name	State_Capital
7	8	NaN	Columbus
8	9	NaN	Austin
9	10	NaN	Richmond

In [16]:
```
df6 = pd.concat([df2,df5],axis=1)
df6
```

Out[16]:

	State_Key	State_Name	State_Key	State_Capital
0	1	Arizona	1	Phoenix
1	2	California	2	Sacremento
2	3	Florida	3	Tallahassee
3	4	Illinois	4	Springfield
4	5	Michigan	5	Lansing
5	6	Minnesota	6	Saint Paul
6	7	New York	7	Albany
7	8	Ohio	8	Columbus
8	9	Texas	9	Austin
9	10	Virginia	10	Richmond

In [17]:
```
df6 = pd.merge(df2,df5, on=["State_Key", "State_Key"])
```

In [18]:
```
df6
```

Out[18]:

	State_Key	State_Name	State_Capital
0	1	Arizona	Phoenix
1	2	California	Sacremento
2	3	Florida	Tallahassee
3	4	Illinois	Springfield
4	5	Michigan	Lansing
5	6	Minnesota	Saint Paul
6	7	New York	Albany
7	8	Ohio	Columbus
8	9	Texas	Austin
9	10	Virginia	Richmond

Color palettes for plotting:

1) https://jiffyclub.github.io/palettable/matplotlib/ 2) https://jiffyclub.github.io/palettable/colorbrewer/ 3) https://github.com/dsc/colorbrewer-python

Plotting libraries:

1) https://analyticsindiamag.com/top-5-python-libraries-for-data-visualization/ 2) https://mode.com/blog/python-data-visualization-libraries/ 3) https://mode.com/blog/python-interactive-plot-libraries/

In [19]:
```
import datetime

x = datetime.datetime.now()
print(x)
```

```
2022-03-21 18:45:29.508182
```

In [20]:
```
print(x.year)
print(x.strftime("%A"))
```

```
2022
Monday
```

In [21]:
```
print(x.strftime("%b"))
```

```
Mar
```

In [22]:
```
x = datetime.datetime(2020, 5, 17)

print(x)
```

```
2020-05-17 00:00:00
```

In [23]:
```
from datetime import date, time, datetime
date(year=2020, month=1, day=31)
```

Out[23]:
```
datetime.date(2020, 1, 31)
```

In [24]:
```
today = date.today()
today
```

Out[24]:
```
datetime.date(2022, 3, 21)
```

In [25]:
```
date(year=2022, month=2, day=29)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-25-6a62e1c7f5d2> in <module>
----> 1 date(year=2022, month=2, day=29)

ValueError: day is out of range for month
```

In [26]:
```
datetime(year=2020, month=1, day=31, hour=13, minute=14, second=31)
```

Out[26]:
```
datetime.datetime(2020, 1, 31, 13, 14, 31)
```

In [27]:
```
x = datetime.now()
```

In [28]:
```
y=datetime.now()
```

In [29]:
```
y-x
```

Out[29]:
```
datetime.timedelta(seconds=12, microseconds=236854)
```

In [30]:
```
date.fromisoformat("2020-01-31")
```

Out[30]:
```
datetime.date(2020, 1, 31)
```

In [31]:
```
date_string = "01-31-2020 14:45:37"
format_string = "%m-%d-%Y %H:%M:%S"
datetime.strptime(date_string, format_string)
```

Out[31]:
```
datetime.datetime(2020, 1, 31, 14, 45, 37)
```

In [32]:
```
import numpy as np
mu, sigma = 0, 1000 # mean and standard deviation
N=1000
s = np.random.normal(mu, sigma, N)
```

In [33]:
```
df7=pd.read_excel('employee_data.xlsx')
```

In [34]:
```
df7 = pd.DataFrame(s, columns = ['Error'])
result['Error']=round(df7['Error'],0)
result.head()
```

Out[34]:

	Person_Key	Age_Key	Gender_Key	State_Key	Children	Salary	Opinion_Key	State_Name	Error
0	1	2	2	9	2	63017	5	Texas	804.0
1	2	2	2	10	3	100302	1	Virginia	1725.0
2	3	2	2	2	0	144043	5	California	535.0
3	4	1	2	2	0	36025	4	California	-744.0
4	5	2	1	9	0	97543	3	Texas	400.0

In [35]:
```
result['Salary_New']=result['Salary']+result['Error']
result.head()
```

Out[35]:

	Person_Key	Age_Key	Gender_Key	State_Key	Children	Salary	Opinion_Key	State_Name	Error	S
0	1	2	2	9	2	63017	5	Texas	804.0	
1	2	2	2	10	3	100302	1	Virginia	1725.0	
2	3	2	2	2	0	144043	5	California	535.0	
3	4	1	2	2	0	36025	4	California	-744.0	
4	5	2	1	9	0	97543	3	Texas	400.0	

In [ ]:

In [ ]:

In [ ]: