In [ ]:
```
#----------------------- Geeting files ready ---------------------------------

# Importing of the modules
import pandas as pd

# Reading the file:
dataset = pd.read_csv("salaries_by_college_major.csv")
```

In [11]:
```
#==================================Make the rows and columns easy to read and ch
    # Renaming columns
dataset.rename(columns  = {"Undergraduate Major":"u_major"}, inplace= True )

    # Check type of data
print(dataset.dtypes)
        # If error , use this code
# dataset[["Starting Median Salary", "Mid-Career Median Salary",
#          "90th Percentile Salary", "10th Percentile Salary"]] = (
#     dataset[["Starting Median Salary", "Mid-Career Median Salary",
#              "90th Percentile Salary", "10th Percentile Salary"]]
#     .apply(pd.to_numeric, errors="coerce")
# )
```

```
u_major                              object
Starting Median Salary              float64
Mid-Career Median Salary            float64
Mid-Career 10th Percentile Salary   float64
Mid-Career 90th Percentile Salary   float64
Group                                object
dtype: object
```

In [12]:
```
#--------------------------- Exploring the data and N/A-----------------------

# Top and last 10
dataset.head(10)
dataset.tail(10)

# Rows and Columns
dataset.shape
dataset.columns

# Get the N/A proportions:
    # N/A Across all dataset
missing_values = dataset.isna().sum()
print(missing_values)


# Removing the number of n/a vlaues using listwise deletion or  partial deletion
## Need to add the inplace function.
dataset.dropna( how= "any", inplace= True)
total_rows =print(f"Total rows:{len(dataset)}")
```

```
u_major                              0
Starting Median Salary               1
Mid-Career Median Salary             1
Mid-Career 10th Percentile Salary    1
Mid-Career 90th Percentile Salary    1
Group                                1
dtype: int64
Total rows:50
```

In [13]:
```
#=======================================Accessing new values: iloc and loc====


# Can pull from the whole dataset using list slicing. This will pull 0 to 4 :
print(dataset[0:5])

#Can give a column to row combination. Pulls the column as a vector and then the
dataset["u_major"][2]


# Using the Loc function dataset[ row_name: column_name] :
dataset['u_major'].loc[43]
dataset.loc[43 ,"u_major" ]                                                    #  Al
dataset.loc[[10, 20, 30], "u_major"]                                           # Mul
dataset.loc[43, ["u_major", "Group"]]                                          # Mul
dataset.loc[10:20, "u_major":"Group"]                                          # Usi
dataset.loc[dataset["Starting Median Salary"] > 80, ["u_major", "Group"]] # Usin

# Using the iloc function dataset[ row_position: column_position] :
dataset.iloc[2]                                                                 # Pu
dataset.iloc[:2]                                                                # Ge
dataset.iloc[0:2, 0:2]                                                          # fi
dataset.iloc[[0, 3], [1]]                                                       # Ro
```

```
                u_major  Starting Median Salary  Mid-Career Median Salary  \
0            Accounting                 46000.0                   77100.0
1  Aerospace Engineering                57700.0                  101000.0
2           Agriculture                 42600.0                   71900.0
3          Anthropology                 36800.0                   61500.0
4          Architecture                 41600.0                   76800.0

   Mid-Career 10th Percentile Salary  Mid-Career 90th Percentile Salary  \
0                            42200.0                           152000.0
1                            64300.0                           161000.0
2                            36300.0                           150000.0
3                            33800.0                           138000.0
4                            50600.0                           136000.0

     Group
0  Business
1     STEM
2  Business
3     HASS
4  Business
```

Out[13]:

	Starting Median Salary
0	46000.0
3	36800.0

In [14]:
```
#====================================Accessing Insight Q1: idmax or access val

# What  major had the hightes mid career salary? What was the amount ? ( Use of
                # Find the exact salary number
mid_career_max_salary = dataset["Mid-Career Median Salary"].max()
                # Using it to find the index value
mid_career_max_salary_index  = dataset[dataset["Mid-Career Median Salary"]==mid_
                # The earlier statement returnd a dataframe so need to access it
major = mid_career_max_salary_index["u_major"].values[0]

                #Pulling the undergraduate degree and presenting the data
print(f"Pathway 1: The highest mid-career salary is ${mid_career_max_salary} for

# Or:
max_salary_id= dataset["Mid-Career Median Salary"].idxmax()
amount = dataset["Mid-Career Median Salary"][max_salary_id]
major =  dataset["u_major"][max_salary_id]
print(f"Pathway 2:  The highest mid-career salary is ${amount} for {major} ")
```

```
Pathway 1: The highest mid-career salary is $107000.0 for Chemical Engineering
Pathway 2:  The highest mid-career salary is $107000.0 for Chemical Engineering
```

In [15]:
```
#====================================Accessing Insight Q2====================

# What  major had the lowest starting salaryy? What was the amount ? ( Use of ID
        # Get the id for the minimum
lowest_salary_index = dataset["Starting Median Salary"].idxmin()

        # Use the idea to get the salary and the occupation
low_salary_major = dataset["u_major"][lowest_salary_index]
low_salary_amount = dataset["Starting Median Salary"][lowest_salary_index]
        # Final sentence
print(f" The lowest mid-career salary is ${low_salary_amount} for {low_salary_ma
```

```
The lowest mid-career salary is $34000.0 for Spanish
```

In [16]:
```
#========================================Low risk majors using insert functio

    # Catching the difference between the 10th and the 90th percentile
diff_10_and_90_salary = dataset["Mid-Career 90th Percentile Salary"] - dataset["

    # Placing it into the dataset( where you want it, name of it , vector)
dataset.insert(1, "Spread", diff_10_and_90_salary )
dataset.head()

    # Sort the values in ascending order
dataset.sort_values( by= "Spread", ascending=True)
```

Out[16]:

	u_major	Spread	Starting Median Salary	Mid-Career Median Salary	Mid-Career 10th Percentile Salary	Mid-Career 90th Percentile Salary	Group
40	Nursing	50700.0	54200.0	67000.0	47600.0	98300.0	Business
43	Physician Assistant	57600.0	74300.0	91700.0	66400.0	124000.0	STEM
41	Nutrition	65300.0	39900.0	55300.0	33900.0	99200.0	HASS
49	Spanish	65400.0	34000.0	53100.0	31000.0	96400.0	HASS
27	Health Care Administration	66400.0	38800.0	60600.0	34600.0	101000.0	Business
47	Religion	66700.0	34100.0	52000.0	29700.0	96400.0	HASS
23	Forestry	70000.0	39100.0	62600.0	41000.0	111000.0	Business
32	Interior Design	71300.0	36100.0	53200.0	35700.0	107000.0	HASS
18	Education	72700.0	34900.0	52000.0	29300.0	102000.0	HASS
15	Criminal Justice	74800.0	35000.0	56300.0	32200.0	107000.0	HASS
26	Graphic Design	76000.0	35700.0	59800.0	36000.0	112000.0	HASS
31	Information Technology (IT)	84500.0	49100.0	74800.0	44500.0	129000.0	STEM
10	Civil Engineering	84600.0	53900.0	90500.0	63400.0	148000.0	STEM
4	Architecture	85400.0	41600.0	76800.0	50600.0	136000.0	Business
48	Sociology	87300.0	36500.0	58200.0	30700.0	118000.0	HASS
29	Hospitality & Tourism	88500.0	37800.0	57500.0	35500.0	124000.0	Business
24	Geography	92000.0	41200.0	65500.0	40000.0	132000.0	HASS
46	Psychology	95400.0	35900.0	60400.0	31600.0	127000.0	HASS
12	Computer Engineering	95900.0	61400.0	105000.0	66100.0	162000.0	STEM
5	Art History	96200.0	35800.0	64900.0	28800.0	125000.0	HASS
1	Aerospace Engineering	96700.0	57700.0	101000.0	64300.0	161000.0	STEM
13	Computer Science	98000.0	55900.0	95500.0	56000.0	154000.0	STEM
6	Biology	98100.0	38800.0	64800.0	36900.0	135000.0	STEM
19	Electrical Engineering	98700.0	60900.0	103000.0	69300.0	168000.0	STEM
38	Mechanical Engineering	99300.0	57900.0	93600.0	63700.0	163000.0	STEM

	u_major	Spread	Starting Median Salary	Mid-Career Median Salary	Mid-Career 10th Percentile Salary	Mid-Career 90th Percentile Salary	Group
20	English	99600.0	38000.0	64700.0	33400.0	133000.0	HASS
35	Management Information Systems (MIS)	100700.0	49200.0	82300.0	45300.0	146000.0	STEM
21	Film	102100.0	37900.0	68500.0	33900.0	136000.0	HASS
9	Chemistry	102700.0	42600.0	79900.0	45300.0	148000.0	STEM
3	Anthropology	104200.0	36800.0	61500.0	33800.0	138000.0	HASS
11	Communications	105500.0	38100.0	70000.0	37500.0	143000.0	HASS
34	Journalism	106600.0	35600.0	66700.0	38400.0	145000.0	HASS
39	Music	107300.0	35900.0	55000.0	26700.0	134000.0	HASS
7	Business Management	108200.0	43000.0	72100.0	38800.0	147000.0	Business
0	Accounting	109800.0	46000.0	77100.0	42200.0	152000.0	Business
25	Geology	111000.0	43500.0	79500.0	45000.0	156000.0	STEM
28	History	112000.0	39200.0	71000.0	37000.0	149000.0	HASS
2	Agriculture	113700.0	42600.0	71900.0	36300.0	150000.0	Business
14	Construction	114700.0	53700.0	88900.0	56300.0	171000.0	Business
30	Industrial Engineering	115900.0	57700.0	94700.0	57100.0	173000.0	STEM
16	Drama	116300.0	35900.0	56900.0	36700.0	153000.0	HASS
33	International Relations	118800.0	40900.0	80900.0	38200.0	157000.0	HASS
44	Physics	122000.0	50300.0	97300.0	56000.0	178000.0	STEM
8	Chemical Engineering	122100.0	63200.0	107000.0	71900.0	194000.0	STEM
45	Political Science	126800.0	40800.0	78200.0	41200.0	168000.0	HASS
42	Philosophy	132500.0	39900.0	81200.0	35500.0	168000.0	HASS
36	Marketing	132900.0	40800.0	79600.0	42100.0	175000.0	Business
37	Math	137800.0	45400.0	92400.0	45200.0	183000.0	STEM
22	Finance	147800.0	47900.0	88300.0	47200.0	195000.0	Business
17	Economics	159400.0	50100.0	98600.0	50600.0	210000.0	Business

In [17]:
```
#=====================================================Challenge Questions======

    # Question 1: Find the degrees with the highest potential ? This means top 5
```

```
dataset.sort_values(by ="Mid-Career 90th Percentile Salary", ascending = False)
dataset.head(5)


    # Challenge Question 2: Find the degrees with the greatest spread as well
dataset.sort_values(by= "Spread", ascending= False)
dataset.head(5)
```

Out[17]:

	u_major	Spread	Starting Median Salary	Mid-Career Median Salary	Mid-Career 10th Percentile Salary	Mid-Career 90th Percentile Salary	Group
0	Accounting	109800.0	46000.0	77100.0	42200.0	152000.0	Business
1	Aerospace Engineering	96700.0	57700.0	101000.0	64300.0	161000.0	STEM
2	Agriculture	113700.0	42600.0	71900.0	36300.0	150000.0	Business
3	Anthropology	104200.0	36800.0	61500.0	33800.0	138000.0	HASS
4	Architecture	85400.0	41600.0	76800.0	50600.0	136000.0	Business

In [18]:

```
#======================================================= The use of group by fun

    # Lets count all the data by the categories( count adds how many fall into e
dataset.groupby("Group").count()
pd.options.display.float_format = '{:,.2f}'.format
    # Use of mean to find the average salary epr group
dataset.groupby("Group").mean(numeric_only=True)
# dataset.groupby("Group")["Starting Median Salary"].mean()
# dataset.groupby("Group").agg({"Starting Median Salary" : "mean",
#                               "Mid-Career Median Salary": "mean"})
```

Out[18]:

Group	Spread	Starting Median Salary	Mid-Career Median Salary	Mid-Career 10th Percentile Salary	Mid-Career 90th Percentile Salary
Business	103,958.33	44,633.33	75,083.33	43,566.67	147,525.00
HASS	95,218.18	37,186.36	62,968.18	34,145.45	129,363.64
STEM	101,600.00	53,862.50	90,812.50	56,025.00	157,625.00