In [1]:
```python
# import required libraries
import pandas as pd
import statsmodels.api as sm
from statsmodels.formula.api import ols
from scipy import stats
```

In [2]:
```python
import pandas as pd
import numpy as np

np.random.seed(123)

n = 150

gender = np.random.choice(['Male', 'Female'], size=n)
age_group = np.random.choice(['Young', 'Middle-aged', 'Old'], size=n)

beauty = np.random.normal(
    loc=[6.5 if a == 'Young' else 5.5 if a == 'Middle-aged' else 4.5 for a in ag
    scale=1.2,
    size=n
)
beauty = np.clip(beauty, 1, 10)

eval_score = (
    0.6 * beauty +
    np.where(gender == 'Female', 0.4, -0.2) +
    np.random.normal(0, 1.2, size=n)
)
eval_score = np.clip(eval_score, 1, 10)

data = pd.DataFrame({
    'gender': gender,
    'age_group': age_group,
    'beauty': beauty.round(2),
    'eval': eval_score.round(2)
})

data.to_csv("teacher_evaluations.csv", index=False)

data
```

Out[2]:

	gender	age_group	beauty	eval
0	Male	Young	5.90	2.60
1	Female	Middle-aged	5.78	1.90
2	Male	Young	6.27	5.29
3	Male	Middle-aged	6.77	3.33
4	Male	Old	1.53	1.00
...	...	...	...	...
145	Male	Middle-aged	4.35	1.08
146	Female	Young	4.89	4.27
147	Female	Middle-aged	6.68	3.46
148	Female	Young	6.44	3.21
149	Female	Middle-aged	6.68	6.92

150 rows × 4 columns

In [3]:
```
# Q1. Regression with T-test: Does gender affect teaching evaluation rates?

# assuming columns are named 'eval' (teaching evaluation) and 'gender' (male/fem
model_ttest = ols('eval ~ gender', data=data).fit()
ttest_result = model_ttest.t_test([0, 1])  # tests coefficient of gender variabl

print("\nQ1. Regression with T-test: Does gender affect teaching evaluation rate
print(model_ttest.summary())
```

```
Q1. Regression with T-test: Does gender affect teaching evaluation rates?
                            OLS Regression Results
==============================================================================
Dep. Variable:                   eval   R-squared:                       0.076
Model:                            OLS   Adj. R-squared:                  0.070
Method:                 Least Squares   F-statistic:                     12.18
Date:                Tue, 28 Oct 2025   Prob (F-statistic):           0.000637
Time:                        12:08:36   Log-Likelihood:                -260.25
No. Observations:                 150   AIC:                             524.5
Df Residuals:                     148   BIC:                             530.5
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      3.8953      0.158     24.590      0.000       3.582       4.208
gender[T.Male]    -0.7870      0.226     -3.490      0.001      -1.233      -0.341
==============================================================================
Omnibus:                        1.670   Durbin-Watson:                   1.871
Prob(Omnibus):                  0.434   Jarque-Bera (JB):                1.467
Skew:                           0.102   Prob(JB):                        0.480
Kurtosis:                       2.561   Cond. No.                         2.60
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly spec
```

In [4]:
```
# Q2. Regression with ANOVA: Does beauty score differ by age group?

model_anova = ols('beauty ~ C(age_group)', data=data).fit()
```

```
anova_table = sm.stats.anova_lm(model_anova, typ=2)

print("\nQ2. Regression with ANOVA: Does beauty score for instructors differ by
print(anova_table)
```

```
Q2. Regression with ANOVA: Does beauty score for instructors differ by age?
                  sum_sq     df          F        PR(>F)
C(age_group)   90.781678    2.0  27.900131  5.349196e-11
Residual      239.154911  147.0        NaN           NaN
```

In [5]:
```
# Q3 . Correlation: Using the teachers' rating dataset, Is teaching evaluation s

import statsmodels.api as sm
from statsmodels.formula.api import ols

# perform OLS regression: eval vs beauty
model = ols('eval ~ beauty', data=data).fit()

# display summary
print(model.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                   eval   R-squared:                       0.240
Model:                            OLS   Adj. R-squared:                  0.235
Method:                 Least Squares   F-statistic:                     46.86
Date:                Tue, 28 Oct 2025   Prob (F-statistic):           1.89e-10
Time:                        12:08:55   Log-Likelihood:                -245.55
No. Observations:                 150   AIC:                             495.1
Df Residuals:                     148   BIC:                             501.1
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      0.9022      0.394      2.290      0.023       0.124       1.681
beauty         0.4719      0.069      6.846      0.000       0.336       0.608
==============================================================================
Omnibus:                        0.729   Durbin-Watson:                   1.938
Prob(Omnibus):                  0.695   Jarque-Bera (JB):                0.854
Skew:                          -0.142   Prob(JB):                        0.652
Kurtosis:                       2.764   Cond. No.                         22.7
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly spe
```

In [ ]:
```

```