Build an OLS model in Excel/Python on select features of the Retail dataset without using in-built OLS functions, compare the output to the various other Python models in 2,3,5 above and identify/explain any differences

- OLS(Ordinary least square) We have equation: y = mx + b
- We can also write this equation in the form y = b2x+b1
- B = y-mx
- m = sumation (xi-mean(x))*(yi-mean(y))/summation(xi-mean(x)**2)
- R^2 TSS(Total sum square)
- Tss = summation((y-mean_y)**2)
- RSS(residual sum of square)
- RSS = summation((y-y_pred)**2)
- R^2 = 1-(rss/tss)

```
import pandas as pd
import numpy as np
```

```
def ordinary_least_squares(X, Y):
    # Step 1: Calculate the means of X and Y
    mean_X = np.mean(X)
    mean_Y = np.mean(Y)

    # Step 2: Calculate the deviations from the means
    dev_X = X - mean_X
    dev_Y = Y - mean_Y

    # Step 3: Calculate the slope (m)
    m = np.sum(dev_X * dev_Y) / np.sum(dev_X ** 2)

    # Step 4: Calculate the intercept (b)
    b = mean_Y - m * mean_X

    # Step 5: Return the coefficients
    return m, b
```

```python
def calculate_r_squared(X, Y, slope, intercept):
    # Step 1: Calculate the predicted values (Y_pred) using the linear regression model
    Y_pred = slope * X + intercept

    # Step 2: Calculate the total sum of squares (TSS)
    mean_Y = np.mean(Y)
    tss = np.sum((Y - mean_Y) ** 2)

    # Step 3: Calculate the residual sum of squares (RSS)
    rss = np.sum((Y - Y_pred) ** 2)

    # Step 4: Calculate R-squared (R²)
    r_squared = 1 - (rss / tss)

    return r_squared
```

```python
x = merged_final['Orders'].values
y = merged_final['Customers'].values

# Step 6: Use the function to get coefficients for your data

slope, intercept = ordinary_least_squares(x, y)
r_squared = calculate_r_squared(x, y, slope, intercept)

# Step 7: Output the results
print("Slope (m):", slope)
print("Intercept (b):", intercept)
print("R-squared (R²):", r_squared)
```

```
Slope (m): 0.107955007903139
Intercept (b): 9.8951915907096
R-squared (R²): 0.799892145855195
```