

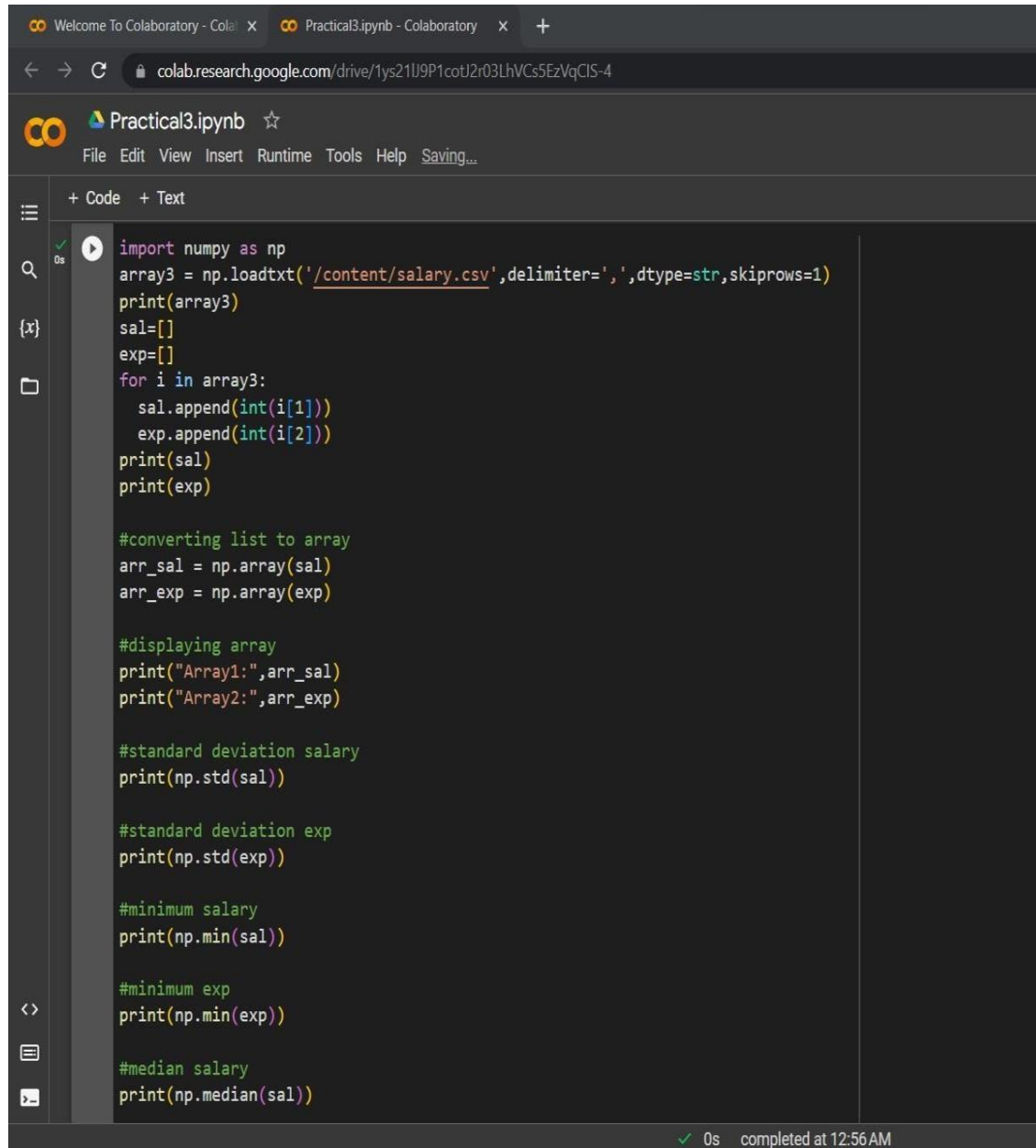
NAME : Krushna kulkarni

ROLL NO. : 737

PRN : 202201090101

DIV : G-2

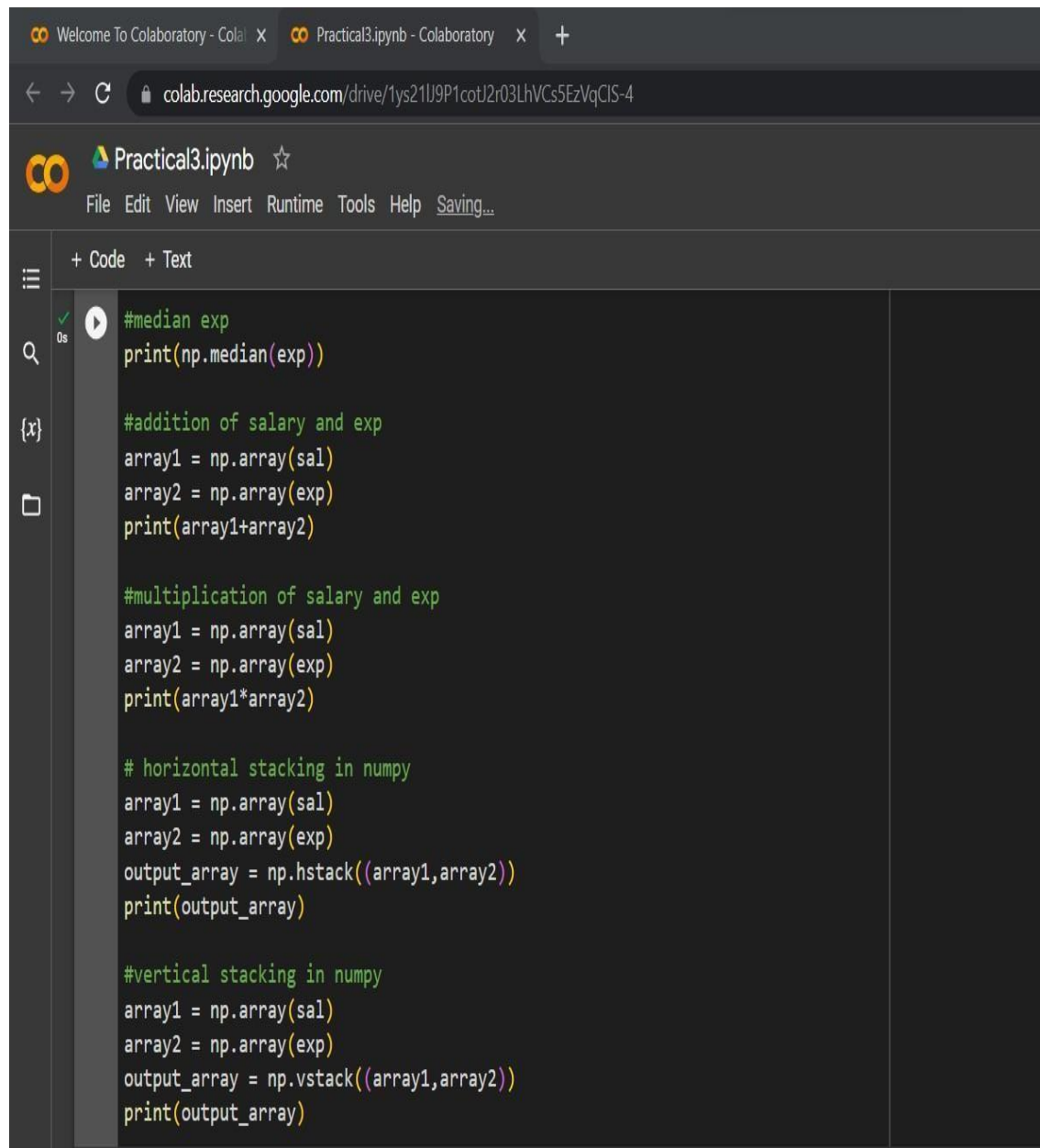
CODE :



The screenshot shows a Google Colaboratory notebook interface. The browser address bar displays the URL: `colab.research.google.com/drive/1ys21U9P1cotJ2r03LhVCs5EzVqCIS-4`. The notebook title is "Practical3.ipynb". The code is written in Python and performs the following operations:

- Imports `numpy` as `np`.
- Loads a CSV file `/content/salary.csv` using `np.loadtxt()` with `delimiter=','` and `skiprows=1`.
- Prints the loaded data (`array3`).
- Creates two empty lists, `sal` and `exp`.
- Iterates over each row in `array3` and appends the salary and experience values to `sal` and `exp` respectively.
- Prints the lists `sal` and `exp`.
- Converts the lists `sal` and `exp` into NumPy arrays (`arr_sal` and `arr_exp`).
- Displays the arrays using `print("Array1:", arr_sal)` and `print("Array2:", arr_exp)`.
- Calculates and prints the standard deviation for salary (`np.std(sal)`).
- Calculates and prints the standard deviation for experience (`np.std(exp)`).
- Calculates and prints the minimum salary (`np.min(sal)`).
- Calculates and prints the minimum experience (`np.min(exp)`).
- Calculates and prints the median salary (`np.median(sal)`).

The status bar at the bottom indicates that the code was executed successfully in 0 seconds and completed at 12:56 AM.



```
#median exp
print(np.median(exp))

#addition of salary and exp
array1 = np.array(sal)
array2 = np.array(exp)
print(array1+array2)

#multiplication of salary and exp
array1 = np.array(sal)
array2 = np.array(exp)
print(array1*array2)

# horizontal stacking in numpy
array1 = np.array(sal)
array2 = np.array(exp)
output_array = np.hstack((array1,array2))
print(output_array)

#vertical stacking in numpy
array1 = np.array(sal)
array2 = np.array(exp)
output_array = np.vstack((array1,array2))
print(output_array)
```

OUTPUT :



+ Code + Text



```
✓ 0s
[[['raj' '25000' '12']
  ['vijay' '20000' '8']
  ['kishor' '15000' '7']
  ['kiran' '18000' '8']
  ['sahil' '21000' '10']
  ['priyank' '30000' '5']
  ['ramesh' '28000' '6']]
[25000, 20000, 15000, 18000, 21000, 30000, 28000]
[12, 8, 7, 8, 10, 5, 6]
Array1: [25000 20000 15000 18000 21000 30000 28000]
Array2: [12 8 7 8 10 5 6]
5038.626311013828
2.2038926600773587
15000
5
21000.0
8.0
[25012 20008 15007 18008 21010 30005 28006]
[300000 160000 105000 144000 210000 150000 168000]
[25000 20000 15000 18000 21000 30000 28000    12    8    7    8    10
 5    6]
[[25000 20000 15000 18000 21000 30000 28000]
 [ 12    8    7    8    10    5    6]]
```