

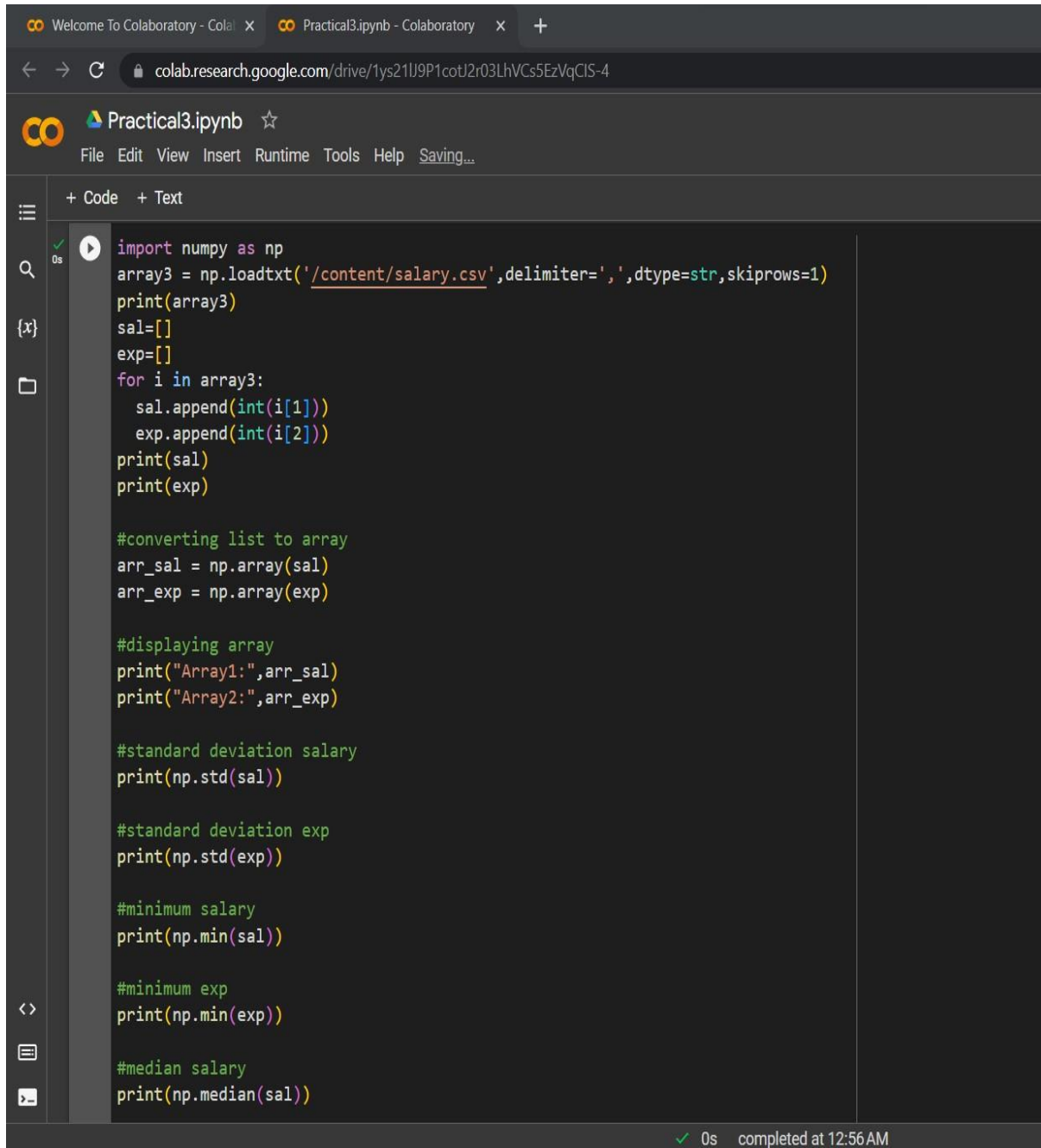
NAME : JANARDHAN BALAJI LAVHALE

ROLL NO. : 637

PRN : 202201050027

DIV : F-2

CODE :



The screenshot shows a Google Colaboratory notebook titled 'Practical3.ipynb'. The code is written in Python and performs the following operations:

- Imports numpy as np.
- Loads a CSV file located at '/content/salary.csv' using np.loadtxt, with a delimiter of ',' and dtype of str, skipping the first row.
- Prints the loaded array.
- Creates two empty lists, sal and exp.
- Iterates over each row in the array, appending the salary and experience values to the respective lists.
- Prints the resulting lists.
- Converts the lists to numpy arrays.
- Displays the arrays.
- Calculates and prints the standard deviation for salary and experience.
- Calculates and prints the minimum salary and experience.
- Calculates and prints the median salary.

The code is executed successfully, as indicated by the green checkmark and '0s' completion time at the bottom right of the notebook interface.

```
import numpy as np
array3 = np.loadtxt('/content/salary.csv', delimiter=',', dtype=str, skiprows=1)
print(array3)
sal=[]
exp=[]
for i in array3:
    sal.append(int(i[1]))
    exp.append(int(i[2]))
print(sal)
print(exp)

#converting list to array
arr_sal = np.array(sal)
arr_exp = np.array(exp)

#displaying array
print("Array1:",arr_sal)
print("Array2:",arr_exp)

#standard deviation salary
print(np.std(sal))

#standard deviation exp
print(np.std(exp))

#minimum salary
print(np.min(sal))

#minimum exp
print(np.min(exp))

#median salary
print(np.median(sal))
```

Welcome To Colaboratory - Colab Practical3.ipynb - Colaboratory +

colab.research.google.com/drive/1ys21U9P1cotJ2r03LhVCs5EzVqCIS-4

Practical3.ipynb ☆

File Edit View Insert Runtime Tools Help Saving...

+ Code + Text

0s

✓

▶

```
#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 :

Practical3.ipynb - Colaboratory

colab.research.google.com/drive/1ys21U9P1cotU2r03LhVCs5EzVqCIS-4

Practical3.ipynb

File Edit View Insert Runtime Tools Help

+ 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]]
```

completed at 1:01 AM

