

DAY-1 LAB EXPERIMENTS

Name : LARAVIND

Reg No: 192424080

Date : 16/12/2025

EXP_1 Analysing Student Performance Data

The screenshot shows a Google Colab notebook titled "Experiment 1 - Colab". The code cell contains the following Python script:

```
import pandas as pd
import numpy as np

df=pd.read_csv("student_score.csv")
print(df.head())

...
Student Math Science English History
0 Arun 78 82 75 80
1 Karthik 85 88 90 84
2 Priya 92 91 89 93
3 Sanjay 67 70 72 68
4 Ananya 88 85 87 90

student_scores=df[["Math","Science","English","History"]].to_numpy()

average_scores = np.mean(student_scores, axis=0)

subjects = ["Math", "Science", "English", "History"]

print("Average Scores:")
for i in range(len(subjects)):
    print(subjects[i],"-",average_scores[i])

print("Highest Average subject:",subjects[np.argmax(average_scores)])

Average Scores:
Math : 81.96875
Science : 82.5625
English : 82.96875
History : 82.875

Highest Average subject: English
```

The output of the code is displayed below the code cell, showing the average scores for each subject and the highest average subject.

EXP_2 Analysing Sales Data

The screenshot shows a Google Colab notebook titled "FODS_Experiments - Colab". The code cell contains the following Python script:

```
import numpy as np
import pandas as pd

df=pd.read_csv("Sales.csv")

prices=df["Price"].to_numpy()

average_price=np.mean(prices)

print("Average price of all products sold:",average_price)
```

The output of the code is displayed below the cell, showing the calculated average price:

```
... Average price of all products sold: 2728.44012
```

The Colab interface includes a sidebar with "Files" containing "sample_data", "Sales.csv", and "student_score.csv". The bottom status bar shows "Python 3" and the date "16-12-2025".

EXP_3 Analysing Dataset Containing Information about Houses in a Neighbourhood.

The screenshot shows a Google Colab notebook titled "FODS_Experiments - Colab". The code cell contains the following Python script:

```
import numpy as np
import pandas as pd

df=pd.read_csv("House_Prediction.csv")
print(df.head())

      S.no   beds   baths   size  zip_code  price
0       0       3    2.5   2590     98144  795000
1       1       4    2.0   2240     98106  915000
2       2       4    3.0   2940     98107  950000
3       3       4    3.0   3800     98199  1950000
4       4       2    2.0   1842     98102  950000

house_data = df[["beds", "price"]].to_numpy()
filtered_houses=house_data[house_data[:,0]>4]
average_price=np.mean(filtered_houses[:,1])

print("Average sale price of houses with more than 4 bedrooms is :",average_price)
```

The output of the code is displayed below the cell, showing the calculated average price:

```
... Average sale price of houses with more than 4 bedrooms is : 1566041.6751592357
```

The Colab interface includes a sidebar with "Files" containing "sample_data", "House_Prediction.csv", "Sales.csv", and "student_score.csv". The bottom status bar shows "Python 3" and the date "16-12-2025".

EXP_4 Sales Performance of a Company over the Past Four Quarters.

The screenshot shows a Google Colab interface with several tabs at the top: Google Classroom - Login Gui, LAB EXPERIMENTS, House_Prediction - Google Sheet, Sales - Google Sheets, and FODS_Experiments - Colab. The main area displays Python code in a code editor:

```
import numpy as np
import pandas as pd

df=pd.read_csv("sales_performance.csv")
print(df.head())

... 0 200001 1517 12359.995 1726.2440 257.669148 246.5 2161.6
1 200002 1248 12592.530 1745.6900 290.437535 326.4 2273.7
2 200003 1677 12607.676 1769.2586 310.550979 322.5 2336.7
3 200004 1393 12679.338 1789.2533 352.695291 321.0 2198.7
4 2001Q1 1558 12643.283 1819.1398 290.974546 329.4 2170.1

year_data=df[df["date"].str.contains("2000")]
sales_data=year_data[["Revenue"]].to_numpy()
total_sales=np.sum(sales_data)
percentage_increase=[(sales_data[3]-sales_data[0])/sales_data[0]]*100

... print("Quarterly Sales Data:", sales_data)
print("Total Sales for the Year:", total_sales)
print("Percentage Increase from Q1 to Q4:", percentage_increase, "%")

... Quarterly Sales Data: [1517 1248 1677 1393]
Total Sales for the Year: 5835
Percentage Increase from Q1 to Q4: [-8.174027686222809 %]
```

The code reads a CSV file named "sales_performance.csv" and prints the first few rows. It then filters the data for the year 2000, extracts the revenue column, calculates the total sales, and computes the percentage increase from Q1 to Q4. The output shows quarterly sales data, total sales for the year, and the percentage increase.

EXP_5 Analysis of Fuel Efficiency of Different Car Models

The screenshot shows a Google Colab interface with several tabs at the top: Google Classroom - Login Gui, Day 1-Datasets - Score, Fuel_Efficiency datasets.xlsx, House_Prediction - Google Sheet, Sales - Google Sheets, and FODS_Experiments - Colab. The main area displays Python code in a code editor:

```
import pandas as pd
import numpy as np

df=pd.read_csv("fuel_Efficiency.csv")
print(df.head())

... 0 25 midsize car 29 4.0
1 26 midsize car 30 4.0
2 25 small sport utility vehicle 27 4.0
3 26 small sport utility vehicle 29 4.0
4 26 small sport utility vehicle 28 4.0

displacement drive_fuel_type highway_mpg(Fuel Efficiency) make \
0 2.5 fwd mazda 36 mazda
1 2.5 fwd gas 37 mazda
2 2.5 fwd gas 31 mazda
3 2.0 fwd gas 34 mazda
4 2.0 fwd gas 32 mazda

model transmission year
0 5 m 2014
1 6 a 2014
2 cx-5 2wd a 2014
3 cx-5 2wd m 2014
4 cx-5 2wd a 2014

fuel_efficiency=df["highway_mpg(Fuel Efficiency)"].to_numpy()
average_efficiency = np.mean(fuel_efficiency)
model1=fuel_efficiency[0]
model2=fuel_efficiency[1]
percentage_increase=[(model2-model1)/model1]*100

... fuel_efficiency=df["highway_mpg(Fuel Efficiency)"].to_numpy()
average_efficiency = np.mean(fuel_efficiency)
model1=fuel_efficiency[0]
model2=fuel_efficiency[1]
percentage_increase=[(model2-model1)/model1]*100
```

The code reads a CSV file named "fuel_Efficiency.csv" and prints the first few rows. It then extracts the "highway_mpg(Fuel Efficiency)" column, calculates the average fuel efficiency, and computes the percentage increase between two specific models. The output shows fuel efficiency data, average fuel efficiency, and percentage increase.

The screenshot shows a Jupyter Notebook interface with a dark theme. The top bar includes the title "FODS_Experiments", navigation links like File, Edit, View, Insert, Runtime, Tools, Help, and a Share button. A status bar at the top right indicates "RAM Disk".

The left sidebar is titled "Files" and lists several CSV files: sample_data, Fuel_Efficiency.csv, House_Prediction.csv, Sales.csv, sales_performance.csv, and student_score.csv.

The main area contains two code cells:

```
[228] fuel_efficiency=df["highway_mpg(Fuel Efficiency)"].to_numpy()
average_efficiency = np.mean(fuel_efficiency)
model1=fuel_efficiency[0]
model2=fuel_efficiency[1]
percentage_increase=(model2-model1)/model1*100

[229] print("Average Fuel Efficiency is:",average_efficiency)
print("Percentage Increase in Fuel Efficiency is:",percentage_increase)
```

The output of the second cell is:

```
Average Fuel Efficiency is: 28.609050909090909
Percentage Increase in Fuel Efficiency is: 2.7777777777777776
```