```
from google.colab import files
uploaded = files.upload()
```

```
Choose Files deepseek_vs_chatgpt.csv
• deepseek_vs_chatgpt.csv(text/csv) - 2416971 bytes, last modified: 4/6/2025 - 100% done
Saving deepseek_vs_chatgpt.csv to deepseek_vs_chatgpt.csv
```

```
import pandas as pd

df = pd.read_csv('deepseek_vs_chatgpt.csv')  # Use the correct file name
df.head()  # Optional: preview the data
```

	Date	Month_Num	Weekday	AI_Platform	AI_Model_Version	Active_Users	New_Users	Churned_Users	Daily_Churn_Rate	Retention_Rate	.
0	2024-09-21	9	Saturday	ChatGPT	GPT-4-turbo	500000	25000	25000	0.05	0.95	
1	2024-09-21	9	Saturday	ChatGPT	GPT-4-turbo	500000	25000	25000	0.05	0.95	
2	2024-09-21	9	Saturday	ChatGPT	GPT-4-turbo	500000	25000	25000	0.05	0.95	
3	2024-09-21	9	Saturday	ChatGPT	GPT-4-turbo	500000	25000	25000	0.05	0.95	
4	2024-05-16	5	Thursday	DeepSeek	DeepSeek-Chat 1.5	1700000	170000	34000	0.02	0.95	

5 rows × 28 columns

```
# Data Indexing
print("First row using iloc:\n", df.iloc[0])
print("First row using loc:\n", df.loc[0])
```

```
Month_Num                                                          9
Weekday                                                     Saturday
AI_Platform                                                  ChatGPT
AI_Model_Version                                           GPT-4-turbo
Active_Users                                                  500000
New_Users                                                      25000
Churned_Users                                                  25000
Daily_Churn_Rate                                                0.05
Retention_Rate                                                  0.95
User_ID                                 c878a177-2da9-4224-8cf8-1d56a1c6a755
Query_Type                                                  General
Input_Text                 Draft a professional email about personal finance
Input_Text_Length                                                  7
Response_Tokens                                                  280
Topic_Category                                    Professional Writing
User_Rating                                                        4
User_Experience_Score                                           1.28
Session_Duration_sec                                              40
Device_Type                                                   Mobile
Language                                                          es
Response_Accuracy                                             0.7842
Response_Speed_sec                                               3.3
Response_Time_Category                                      Standard
Correction_Needed                                                  0
User_Return_Frequency                                              6
Customer_Support_Interactions                                      2
Region                          Antarctica (the territory South of 60 deg S)
Name: 0, dtype: object
First row using loc:
 Date                                                     2024-09-21
Month_Num                                                          9
Weekday                                                     Saturday
AI_Platform                                                  ChatGPT
AI_Model_Version                                           GPT-4-turbo
Active_Users                                                  500000
New_Users                                                      25000
Churned_Users                                                  25000
Daily_Churn_Rate                                                0.05
Retention_Rate                                                  0.95
```

```
Input_Text_Length                                                    7
Response_Tokens                                                    280
Topic_Category                                     Professional Writing
User_Rating                                                        4
User_Experience_Score                                           1.28
Session_Duration_sec                                              40
Device_Type                                                   Mobile
Language                                                          es
Response_Accuracy                                             0.7842
Response_Speed_sec                                               3.3
Response_Time_Category                                      Standard
Correction_Needed                                                  0
User_Return_Frequency                                              6
Customer_Support_Interactions                                      2
Region                          Antarctica (the territory South of 60 deg S)
Name: 0, dtype: object
```

```
#Series Object
platform_series = df["AI_Platform"]
print("Series (AI_Platform):\n", platform_series.head())
```

```
Series (AI_Platform):
 0     ChatGPT
1     ChatGPT
2     ChatGPT
3     ChatGPT
4    DeepSeek
Name: AI_Platform, dtype: object
```

```
#Indexing Series
print("First platform:", platform_series[0])
print("First 5 platforms:\n", platform_series[:5])
```

```
First platform: ChatGPT
First 5 platforms:
 0     ChatGPT
1     ChatGPT
2     ChatGPT
3     ChatGPT
4    DeepSeek
Name: AI_Platform, dtype: object
```

```
#Selecting Multiple Columns
df_subset = df[["AI_Platform", "User_Experience_Score", "Response_Accuracy"]]
print("Subset of columns:\n", df_subset.head())
```

```
Subset of columns:
   AI_Platform  User_Experience_Score  Response_Accuracy
0     ChatGPT                   1.28             0.7842
1     ChatGPT                   1.28             0.8194
2     ChatGPT                   0.88             0.8090
3     ChatGPT                   1.68             0.8233
4    DeepSeek                   2.04             0.9366
```

```
#Filtering Rows
perfect_users = df[df["User_Experience_Score"] == 10]
print("Users with score 10:\n", perfect_users)
```

```
Users with score 10:
 Empty DataFrame
Columns: [Date, Month_Num, Weekday, AI_Platform, AI_Model_Version, Active_Users, New_Users, Churned_Users, Daily_Churn_Rate, Retention_R
Index: []

[0 rows x 28 columns]
```

```
#Universal Function (round)
rounded_accuracy = df["Response_Accuracy"].round(2)
print("Rounded Response Accuracy:\n", rounded_accuracy.head())
```

```
Rounded Response Accuracy:
 0    0.78
1    0.82
2    0.81
3    0.82
4    0.94
Name: Response_Accuracy, dtype: float64
```

```
#Index Alignment in Arithmetic
combined_score = df["User_Experience_Score"] + df["Response_Accuracy"]
print("Combined Score (Experience + Accuracy):\n", combined_score.head())
```

```
Combined Score (Experience + Accuracy):
 0    2.0642
1    2.0994
2    1.6890
3    2.5033
4    2.9766
dtype: float64
```

```
#Add a New Column
df["ExperiencePlusAccuracy"] = combined_score
print("New column 'ExperiencePlusAccuracy' added:\n", df[["ExperiencePlusAccuracy"]].head())
```

```
New column 'ExperiencePlusAccuracy' added:
    ExperiencePlusAccuracy
0                  2.0642
1                  2.0994
2                  1.6890
3                  2.5033
4                  2.9766
```

```
#Drop Rows with Null Values
df_clean = df.dropna()
print("DataFrame after dropping nulls:\n", df_clean.head())
```

```
DataFrame after dropping nulls:
          Date  Month_Num   Weekday AI_Platform   AI_Model_Version  \
0  2024-09-21          9  Saturday     ChatGPT        GPT-4-turbo
1  2024-09-21          9  Saturday     ChatGPT        GPT-4-turbo
2  2024-09-21          9  Saturday     ChatGPT        GPT-4-turbo
3  2024-09-21          9  Saturday     ChatGPT        GPT-4-turbo
4  2024-05-16          5  Thursday    DeepSeek  DeepSeek-Chat 1.5

   Active_Users  New_Users  Churned_Users  Daily_Churn_Rate  Retention_Rate  \
0        500000      25000          25000              0.05            0.95
1        500000      25000          25000              0.05            0.95
2        500000      25000          25000              0.05            0.95
3        500000      25000          25000              0.05            0.95
4       1700000     170000          34000              0.02            0.95

   ...     Device_Type Language Response_Accuracy  Response_Speed_sec  \
0  ...          Mobile       es            0.7842                3.30
1  ...  Laptop/Desktop       zh            0.8194                3.28
2  ...          Mobile       en            0.8090                3.07
3  ...          Mobile       fr            0.8233                3.06
4  ...          Mobile       de            0.9366                1.48

   Response_Time_Category Correction_Needed  User_Return_Frequency  \
0                Standard                 0                      6
1                Standard                 1                      2
2                Standard                 0                      2
3                Standard                 0                      9
4                    Fast                 0                      9

   Customer_Support_Interactions  \
0                              2
1                              2
2                              0
3                              0
4                              3

                                      Region ExperiencePlusAccuracy
0  Antarctica (the territory South of 60 deg S)                2.0642
1                                    Ukraine                2.0994
2                                    Grenada                1.6890
3                                     Guyana                2.5033
4                                      India                2.9766

[5 rows x 29 columns]
```

```
#Handling Missing Data (Check Nulls)
print("Null values in each column:\n", df.isnull().sum())
```

```
Null values in each column:
 Date                          0
Month_Num                      0
Weekday                        0
```

```
AI_Platform                       0
AI_Model_Version                  0
Active_Users                      0
New_Users                         0
Churned_Users                     0
Daily_Churn_Rate                  0
Retention_Rate                    0
User_ID                           0
Query_Type                        0
Input_Text                        0
Input_Text_Length                 0
Response_Tokens                   0
Topic_Category                    0
User_Rating                       0
User_Experience_Score             0
Session_Duration_sec              0
Device_Type                       0
Language                          0
Response_Accuracy               379
Response_Speed_sec                0
Response_Time_Category            0
Correction_Needed                 0
User_Return_Frequency             0
Customer_Support_Interactions     0
Region                            0
ExperiencePlusAccuracy          379
dtype: int64
```

```
#Fill Missing Values
df_filled = df.fillna(0)
print("DataFrame after filling nulls with 0:\n", df_filled.head())
```

```
DataFrame after filling nulls with 0:
         Date  Month_Num   Weekday AI_Platform   AI_Model_Version  \
0  2024-09-21          9  Saturday     ChatGPT        GPT-4-turbo
1  2024-09-21          9  Saturday     ChatGPT        GPT-4-turbo
2  2024-09-21          9  Saturday     ChatGPT        GPT-4-turbo
3  2024-09-21          9  Saturday     ChatGPT        GPT-4-turbo
4  2024-05-16          5  Thursday    DeepSeek  DeepSeek-Chat 1.5

   Active_Users  New_Users  Churned_Users  Daily_Churn_Rate  Retention_Rate  \
0        500000      25000          25000              0.05            0.95
1        500000      25000          25000              0.05            0.95
2        500000      25000          25000              0.05            0.95
3        500000      25000          25000              0.05            0.95
4       1700000     170000          34000              0.02            0.95

   ...     Device_Type Language Response_Accuracy  Response_Speed_sec  \
0  ...          Mobile       es            0.7842                3.30
1  ...  Laptop/Desktop       zh            0.8194                3.28
2  ...          Mobile       en            0.8090                3.07
3  ...          Mobile       fr            0.8233                3.06
4  ...          Mobile       de            0.9366                1.48

   Response_Time_Category Correction_Needed  User_Return_Frequency  \
0                Standard                 0                      6
1                Standard                 1                      2
2                Standard                 0                      2
3                Standard                 0                      9
4                    Fast                 0                      9

   Customer_Support_Interactions  \
0                              2
1                              2
2                              0
3                              0
4                              3

                                      Region ExperiencePlusAccuracy
0  Antarctica (the territory South of 60 deg S)                2.0642
1                                    Ukraine                2.0994
2                                    Grenada                1.6890
3                                     Guyana                2.5033
4                                      India                2.9766

[5 rows x 29 columns]
```

```
#Replace Values
df["Language"] = df["Language"].replace("en", "English")
print("Updated Language column:\n", df["Language"].head())
```

```
Updated Language column:
 0         es
1         zh
2    English
3         fr
4         de
Name: Language, dtype: object
```

```
#GroupBy + Aggregation
avg_score_by_platform = df.groupby("AI_Platform")["User_Experience_Score"].mean()
print("Average experience score by platform:\n", avg_score_by_platform)
```

```
Average experience score by platform:
 AI_Platform
ChatGPT     1.230971
DeepSeek    2.034657
Name: User_Experience_Score, dtype: float64
```

```
#Sort Values
sorted_df = df.sort_values("User_Experience_Score", ascending=False)
print("Data sorted by User_Experience_Score:\n", sorted_df[["AI_Platform", "User_Experience_Score"]].head())
```

```
Data sorted by User_Experience_Score:
      AI_Platform  User_Experience_Score
5212    DeepSeek                   2.28
5211    DeepSeek                   2.28
5213    DeepSeek                   2.28
1530    DeepSeek                   2.27
7688    DeepSeek                   2.27
```

```
#Value Counts
platform_counts = df["AI_Platform"].value_counts()
print("Count of each AI Platform:\n", platform_counts)
```

```
Count of each AI Platform:
 AI_Platform
ChatGPT     5076
DeepSeek    4924
Name: count, dtype: int64
```

```
#Unique Values
languages = df["Language"].unique()
print("Unique Languages:\n", languages)
```

```
Unique Languages:
 ['es' 'zh' 'English' 'fr' 'de']
```

```
#Hierarchical Indexing
df_hier = df.set_index(["AI_Platform", "AI_Model_Version"])
print("Hierarchical Indexing:\n", df_hier.head())
```

```
ChatGPT     GPT-4-turbo                          6
            GPT-4-turbo                          2
            GPT-4-turbo                          2
            GPT-4-turbo                          9
DeepSeek    DeepSeek-Chat 1.5                    9

                                Customer_Support_Interactions  \
AI_Platform AI_Model_Version
ChatGPT     GPT-4-turbo                                     2
            GPT-4-turbo                                     2
            GPT-4-turbo                                     0
            GPT-4-turbo                                     0
DeepSeek    DeepSeek-Chat 1.5                               3

                                                                  Region  \
AI_Platform AI_Model_Version
ChatGPT     GPT-4-turbo       Antarctica (the territory South of 60 deg S)
            GPT-4-turbo                                            Ukraine
            GPT-4-turbo                                            Grenada
            GPT-4-turbo                                             Guyana
DeepSeek    DeepSeek-Chat 1.5                                        India

                                ExperiencePlusAccuracy
AI_Platform AI_Model_Version
ChatGPT     GPT-4-turbo                         2.0642
            GPT-4-turbo                         2.0994
            GPT-4-turbo                         1.6890
            GPT-4-turbo                         2.5033
DeepSeek    DeepSeek-Chat 1.5                   2.9766

[5 rows x 27 columns]
```

```
1 #Reset Index
2 df_reset = df_hier.reset_index()
3 print("Index reset:\n", df_reset.head())
```

```
Index reset:
   AI_Platform   AI_Model_Version        Date  Month_Num   Weekday  \
0     ChatGPT        GPT-4-turbo  2024-09-21          9  Saturday
1     ChatGPT        GPT-4-turbo  2024-09-21          9  Saturday
2     ChatGPT        GPT-4-turbo  2024-09-21          9  Saturday
3     ChatGPT        GPT-4-turbo  2024-09-21          9  Saturday
4    DeepSeek  DeepSeek-Chat 1.5  2024-05-16          5  Thursday

   Active_Users  New_Users  Churned_Users  Daily_Churn_Rate  Retention_Rate  \
0        500000      25000          25000              0.05            0.95
1        500000      25000          25000              0.05            0.95
2        500000      25000          25000              0.05            0.95
3        500000      25000          25000              0.05            0.95
4       1700000     170000          34000              0.02            0.95

   ...     Device_Type Language Response_Accuracy  Response_Speed_sec  \
0  ...          Mobile       es            0.7842                3.30
1  ...  Laptop/Desktop       zh            0.8194                3.28
2  ...          Mobile  English            0.8090                3.07
3  ...          Mobile       fr            0.8233                3.06
4  ...          Mobile       de            0.9366                1.48

   Response_Time_Category Correction_Needed  User_Return_Frequency  \
0                Standard                 0                      6
1                Standard                 1                      2
2                Standard                 0                      2
3                Standard                 0                      9
4                    Fast                 0                      9

   Customer_Support_Interactions  \
0                              2
1                              2
2                              0
3                              0
4                              3

                                     Region ExperiencePlusAccuracy
0  Antarctica (the territory South of 60 deg S)                2.0642
1                                    Ukraine                2.0994
2                                    Grenada                1.6890
3                                     Guyana                2.5033
4                                      India                2.9766

[5 rows x 29 columns]
```

```
1 #Apply Custom Function
2 df["Session_Category"] = df["Session_Duration_sec"].apply(lambda x: "Long" if x > 30 else "Short")
```

```
print("New Session_Category column:\n", df[["Session_Duration_sec", "Session_Category"]].head())
```

```
New Session_Category column:
    Session_Duration_sec Session_Category
0                    40             Long
1                    24            Short
2                    34             Long
3                    18            Short
4                    10            Short
```

```
#merge()
feedback_data = {
    "User_ID": df["User_ID"].head(10),  # take first 10 user IDs
    "User_Feedback": [
        "Excellent", "Good", "Average", "Poor", "Excellent",
        "Good", "Poor", "Average", "Excellent", "Good"
    ]
}

feedback_df = pd.DataFrame(feedback_data)

# Merge the original DataFrame with feedback_df on User_ID
merged_df = pd.merge(df, feedback_df, on="User_ID", how="left")

# Print merged result (selected columns)
print(merged_df[["User_ID", "AI_Platform", "User_Experience_Score", "User_Feedback"]].head(10))

```

```
                                User_ID AI_Platform  User_Experience_Score  \
0  c878a177-2da9-4224-8cf8-1d56a1c6a755     ChatGPT                   1.28
1  7096d0f1-d0dc-4333-a5d0-de9dfd1b99fa     ChatGPT                   1.28
2  e690c254-582f-49c1-89c3-0b4dd8ee59be     ChatGPT                   0.88
3  0b6a010d-9d03-44c4-bf7f-7f8d2cc461e2     ChatGPT                   1.68
4  ffa90616-1fa9-48ff-842d-e84e193c64f4    DeepSeek                   2.04
5  eabd6678-8383-40e9-98d8-0d4814d6daf9    DeepSeek                   2.04
6  914e7cef-0512-4a9b-a144-b4da35d11f4c    DeepSeek                   2.04
7  57a6cc5b-8884-4553-bdcf-e4579f92581b    DeepSeek                   2.04
8  c6bb6063-5271-43d3-a16e-b4ca2149f246    DeepSeek                   2.04
9  25f9fff3-0edf-4144-a435-08f1a8f50799    DeepSeek                   1.64

  User_Feedback
0     Excellent
1          Good
2       Average
3          Poor
4     Excellent
5          Good
6          Poor
7       Average
8     Excellent
9          Good
```