Châu Gia Kiệt – 2274802010449 – 0103

LAB 2: BÀI THỰC HÀNH TRÌNH BÀY DỮ LIỆU

Phần 1: Thống kê dữ liệu

1. Hãy sắp xếp dữ liệu điểm DH1 theo thứ tự tăng dần

```python
import pandas as pd

# Đọc dữ liệu từ file CSV
file_path = "processed_dulieuxettuyendaihoc.csv"
df = pd.read_csv(file_path)

# Câu 1: Sắp xếp dữ liệu điểm DH1 theo thứ tự tăng dần
df_sorted_dh1 = df.sort_values(by="DH1", ascending=True)
print("Dữ liệu sắp xếp theo DH1 tăng dần:")
print(df_sorted_dh1[["DH1"]])
```

```
Dữ liệu sắp xếp theo DH1 tăng dần:
     DH1
53  1.00
43  1.25
72  1.50
5   1.50
87  1.75
..   ...
84  6.50
11  6.75
23  6.75
24  7.00
97  7.00

[100 rows x 1 columns]
```

2. Hãy sắp xếp dữ liệu điểm DH2 tăng dần theo nhóm giới tính

```python
# Câu 2: Sắp xếp dữ liệu điểm DH2 tăng dần theo nhóm giới tính (GT)
df_sorted_dh2_gt = df.sort_values(by=["GT", "DH2"], ascending=[True, True])
print("\nDữ liệu sắp xếp theo DH2 và GT:")
print(df_sorted_dh2_gt[["GT", "DH2"]])
```

```
Dữ liệu sắp xếp theo DH2 và GT:
   GT   DH2
95  F  1.50
25  F  2.00
24  F  2.25
46  F  2.25
13  F  2.75
.. ..   ...
68  M  6.00
11  M  6.50
47  M  6.50
2   M  6.75
97  M  8.00

[100 rows x 2 columns]
```

3. Hãy tạo pivot-table để thống kê các giá trị count, sum, mean, median, min, max, std, Q1, Q2 và Q3 của DH1 theo KT

```
# Câu 3: Tạo pivot-table thống kê DH1 theo KT
pivot_dh1_kt = df.pivot_table(
    values="DH1",
    index="KT",
    aggfunc=["count", "sum", "mean", "median", "min", "max", "std",
             lambda x: x.quantile(0.25), lambda x: x.quantile(0.50), lambda x: x.quantile(0.75)]
)
print("Tổng điểm DH1:", df["DH1"].sum())
pivot_dh1_kt.columns = ["count", "sum", "mean", "median", "min", "max", "std", "Q1", "Q2", "Q3"]
print("\nPivot-table DH1 theo KT:")
print(pivot_dh1_kt)
```

```
Tổng điểm DH1: 374.0

Pivot-table DH1 theo KT:
    count     sum      mean  median   min   max       std      Q1    Q2     Q3
KT
A      49  163.50  3.336735    3.25  1.00  6.50  1.274296  2.2500  3.25  4.250
A1      6   17.50  2.916667    3.00  2.00  4.25  0.861201  2.1875  3.00  3.250
B       9   29.50  3.277778    3.50  1.75  4.50  0.823905  3.2500  3.50  3.750
C      14   79.25  5.660714    5.25  4.75  7.00  0.812209  5.0625  5.25  6.375
D1     22   84.25  3.829545    3.75  1.25  6.75  1.370146  3.0000  3.75  4.875
```

4. Hãy tạo pivot-table để thống kê các giá trị count, sum, mean, median, min, max, std, Q1, Q2 và Q3 của DH1 theo KT và KV

```
# Câu 4: Tạo pivot-table thống kê DH1 theo KT và KV
pivot_dh1_kt_kv = df.pivot_table(
    values="DH1",
    index=["KT", "KV"],
    aggfunc=["count", "sum", "mean", "median", "min", "max", "std",
             lambda x: x.quantile(0.25), lambda x: x.quantile(0.50), lambda x: x.quantile(0.75)]
)
pivot_dh1_kt_kv.columns = ["count", "sum", "mean", "median", "min", "max", "std", "Q1", "Q2", "Q3"]
print("\nPivot-table DH1 theo KT và KV:")
print(pivot_dh1_kt_kv)
```

```
Pivot-table DH1 theo KT và KV:
       count    sum      mean  median   min   max       std      Q1      Q2      Q3
KT KV
A  1      29  93.75  3.232759   3.250  1.00  6.50  1.369194  2.0000   3.250  4.2500
   2       9  31.75  3.527778   3.500  2.00  6.25  1.427653  2.2500   3.500  4.2500
   2NT    11  38.00  3.454545   3.500  2.00  4.75  0.913908  2.8750   3.500  4.1250
A1 1       2   4.75  2.375000   2.375  2.00  2.75  0.530330  2.1875   2.375  2.5625
   2NT     4  12.75  3.187500   3.250  2.00  4.25  0.921389  2.9375   3.250  3.5000
B  1       8  25.75  3.218750   3.375  1.75  4.50  0.860207  3.0000   3.375  3.5625
   2NT     1   3.75  3.750000   3.750  3.75  3.75       NaN  3.7500   3.750  3.7500
C  1       8  43.50  5.437500   5.250  4.75  7.00  0.728869  5.0000   5.250  5.4375
   2       2  11.50  5.750000   5.750  5.00  6.50  1.060660  5.3750   5.750  6.1250
   2NT     4  24.25  6.062500   6.000  5.25  7.00  0.943729  5.2500   6.000  6.8125
D1 1      13  46.75  3.596154   3.500  1.50  5.75  1.264658  2.7500   3.500  4.2500
   2       8  33.75  4.218750   4.500  1.25  6.75  1.617082  3.5625   4.500  5.0000
   2NT     1   3.75  3.750000   3.750  3.75  3.75       NaN  3.7500   3.750  3.7500
```

5. Hãy tạo pivot-table để thống kê các giá trị count, sum, mean, median, min, max, std, Q1, Q2 và Q3 của DH1 theo KT, KV và DT

```python
# Câu 5: Tạo pivot-table thống kê DH1 theo KT, KV và DT
pivot_dh1_kt_kv_dt = df.pivot_table(
    values="DH1",
    index=["KT", "KV", "DT"],
    aggfunc=["count", "sum", "mean", "median", "min", "max", "std",
             lambda x: x.quantile(0.25), lambda x: x.quantile(0.50), lambda x: x.quantile(0.75)]
)
pivot_dh1_kt_kv_dt.columns = ["count", "sum", "mean", "median", "min", "max", "std", "Q1", "Q2", "Q3"]
print("\nPivot-table DH1 theo KT, KV và DT:")
print(pivot_dh1_kt_kv_dt)
```

```
Pivot-table DH1 theo KT, KV và DT:
             count    sum      mean  median   min   max       std      Q1     Q2      Q3
KT KV  DT
A  1   0.0     29  93.75  3.232759   3.250  1.00  6.50  1.369194  2.0000  3.250  4.2500
   2   0.0      8  27.50  3.437500   3.000  2.00  6.25  1.498511  2.2500  3.000  4.1875
       6.0      1   4.25  4.250000   4.250  4.25  4.25       NaN  4.2500  4.250  4.2500
   2NT 0.0     11  38.00  3.454545   3.500  2.00  4.75  0.913908  2.8750  3.500  4.1250
A1 1   0.0      2   4.75  2.375000   2.375  2.00  2.75  0.530330  2.1875  2.375  2.5625
   2NT 0.0      4  12.75  3.187500   3.250  2.00  4.25  0.921389  2.9375  3.250  3.5000
B  1   0.0      8  25.75  3.218750   3.375  1.75  4.50  0.860207  3.0000  3.375  3.5625
   2NT 0.0      1   3.75  3.750000   3.750  3.75  3.75       NaN  3.7500  3.750  3.7500
C  1   0.0      8  43.50  5.437500   5.250  4.75  7.00  0.728869  5.0000  5.250  5.4375
   2   0.0      2  11.50  5.750000   5.750  5.00  6.50  1.060660  5.3750  5.750  6.1250
   2NT 0.0      4  24.25  6.062500   6.000  5.25  7.00  0.943729  5.2500  6.000  6.8125
D1 1   0.0     12  43.75  3.645833   3.625  1.50  5.75  1.307576  2.6875  3.625  4.4375
       1.0      1   3.00  3.000000   3.000  3.00  3.00       NaN  3.0000  3.000  3.0000
   2   0.0      7  32.50  4.642857   4.500  3.00  6.75  1.171334  4.1250  4.500  5.0000
       1.0      1   1.25  1.250000   1.250  1.25  1.25       NaN  1.2500  1.250  1.2500
   2NT 0.0      1   3.75  3.750000   3.750  3.75  3.75       NaN  3.7500  3.750  3.7500
```

Phần 2: Trình bày dữ liệu

```python
import pandas as pd
import matplotlib.pyplot as plt

# Đọc dữ liệu từ file CSV
file_path = "processed_dulieuxettuyendaihoc.csv"
df = pd.read_csv(file_path)
```

1. Hãy trình bày dữ liệu biến: GT

Gợi ý:

- Lập bảng tần số và tần suất
- Vẽ biểu đồ tần số (cột), biểu đồ tần suất (tròn).

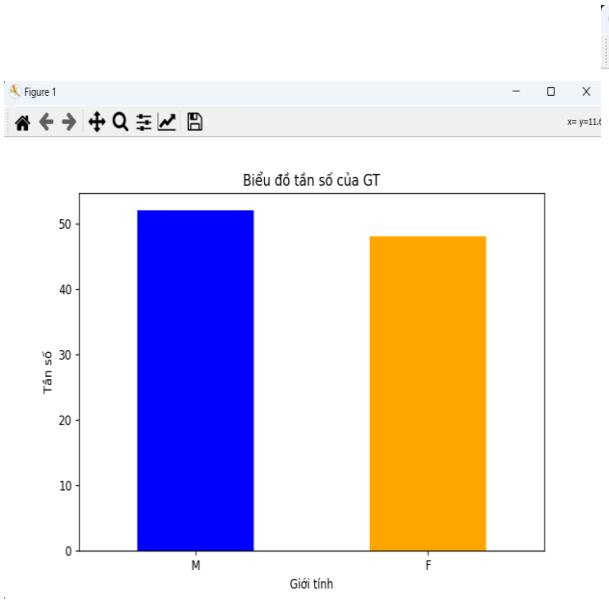
GT	Tần số	Tần suất (%)
M	52	52.0
F	48	48.0

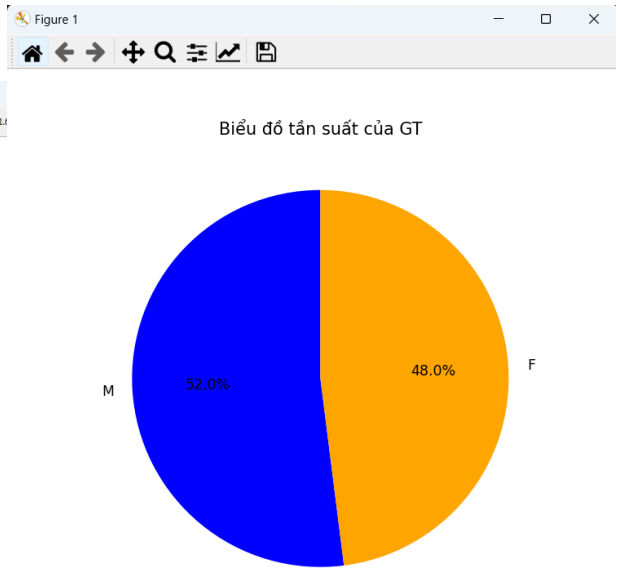
```python
# Câu 1: Trình bày dữ liệu biến GT
# Lập bảng tần số và tần suất
gt_counts = df["GT"].value_counts()
gt_frequencies = df["GT"].value_counts(normalize=True) * 100

gt_table = pd.DataFrame({"Tần số": gt_counts, "Tần suất (%)": gt_frequencies})
print("Bảng tần số và tần suất của GT:")
print(gt_table)

# Vẽ biểu đồ tần số (cột)
plt.figure(figsize=(8, 5))
gt_counts.plot(kind="bar", color=["blue", "orange"])
plt.title("Biểu đồ tần số của GT")
plt.xlabel("Giới tính")
plt.ylabel("Tần số")
plt.xticks(rotation=0)
plt.show()

# Vẽ biểu đồ tần suất (tròn)
plt.figure(figsize=(6, 6))
gt_counts.plot(kind="pie", autopct="%.1f%%", colors=["blue", "orange"], startangle=90)
plt.title("Biểu đồ tần suất của GT")
plt.ylabel("")  # Ẩn nhãn trục y
plt.show()
```





2. Hãy trình bày dữ liệu lần lượt các biến: US_TBM1, US_TBM2 và US_TBM3

```python
# Câu 2: Trình bày dữ liệu lần lượt các biến US_TBM1, US_TBM2 và US_TBM3
us_tbm_columns = ["US_TBM1", "US_TBM2", "US_TBM3"]
print("\nThống kê mô tả cho US_TBM1, US_TBM2, US_TBM3:")
print(df[us_tbm_columns].describe())
```

```
Thống kê mô tả cho US_TBM1, US_TBM2, US_TBM3:
          US_TBM1     US_TBM2    US_TBM3
count  100.000000  100.000000  100.00000
mean     2.434480    2.559600    2.67176
std      0.391203    0.377656    0.29981
min      1.616000    1.768000    1.91600
25%      2.107000    2.223000    2.47300
50%      2.480000    2.588000    2.70600
75%      2.724000    2.893000    2.87400
max      3.232000    3.416000    3.31600
```

3. Hãy trình bày dữ liệu biến DT với các học sinh là nam

```python
# Câu 3: Trình bày dữ liệu biến DT với các học sinh là nam
dt_nam = df[df["GT"] == "M"]
dt_grouped = dt_nam.groupby("DT")["DT"].agg("count")
print("\nTần số của DT cho học sinh nam:")
print(dt_grouped)
```

```
Tần số của DT cho học sinh nam:
DT
0.0    50
1.0     1
6.0     1
Name: DT, dtype: int64
```

4. Hãy trình bày dữ liệu biến KV với các học sinh là nam thuộc dân tộc Kinh, có điểm thỏa mãn điều kiện (DH1 >= 5.0 và DH2 >= 4.0 và DH3 >= 4.0)

```python
# Câu 4: Trình bày dữ liệu biến KV với các học sinh là nam thuộc dân tộc Kinh, có điểm thỏa mãn điều kiện
df_filtered = df[(df["GT"] == "M") & (df["DT"] == 1) & (df["DH1"] >= 5.0) & (df["DH2"] >= 4.0) & (df["DH3"] >= 4.0)]
kv_grouped = df_filtered.groupby("KV")["KV"].agg("count")
print("\nTần số của KV với học sinh nam thuộc dân tộc Kinh và đủ điều kiện điểm:")
print(kv_grouped)
```

```
Tần số của KV với học sinh nam thuộc dân tộc Kinh và đủ điều kiện điểm:
Series([], Name: KV, dtype: int64)
```

5. Hãy trình bày dữ liệu lần lượt các biến DH1, DH2, DH3 lớn hơn bằng 5.0 và thuộc khu vực 2NT

```python
# Câu 5: Trình bày dữ liệu lần lượt các biến DH1, DH2, DH3 lớn hơn bằng 5.0 và thuộc khu vực 2NT
df_filtered_dh = df[(df["DH1"] >= 5.0) & (df["DH2"] >= 5.0) & (df["DH3"] >= 5.0) & (df["KV"] == "2NT")]
print("\nDữ liệu học sinh có DH1, DH2, DH3 >= 5.0 và thuộc KV 2NT:")
print(df_filtered_dh[["DH1", "DH2", "DH3", "KV"]])
```

```
Dữ liệu học sinh có DH1, DH2, DH3 >= 5.0 và thuộc KV 2NT:
     DH1   DH2  DH3   KV
23  6.75  5.25  5.0  2NT
```

Châu Gia Kiệt – 2274802010449 – 0103

Phần 3: Trực quan hóa dữ liệu theo nhóm phân loại

1. Trực quan dữ liệu học sinh nữ trên các nhóm XL1, XL2, XL3 dạng unstacked

Gợi ý:

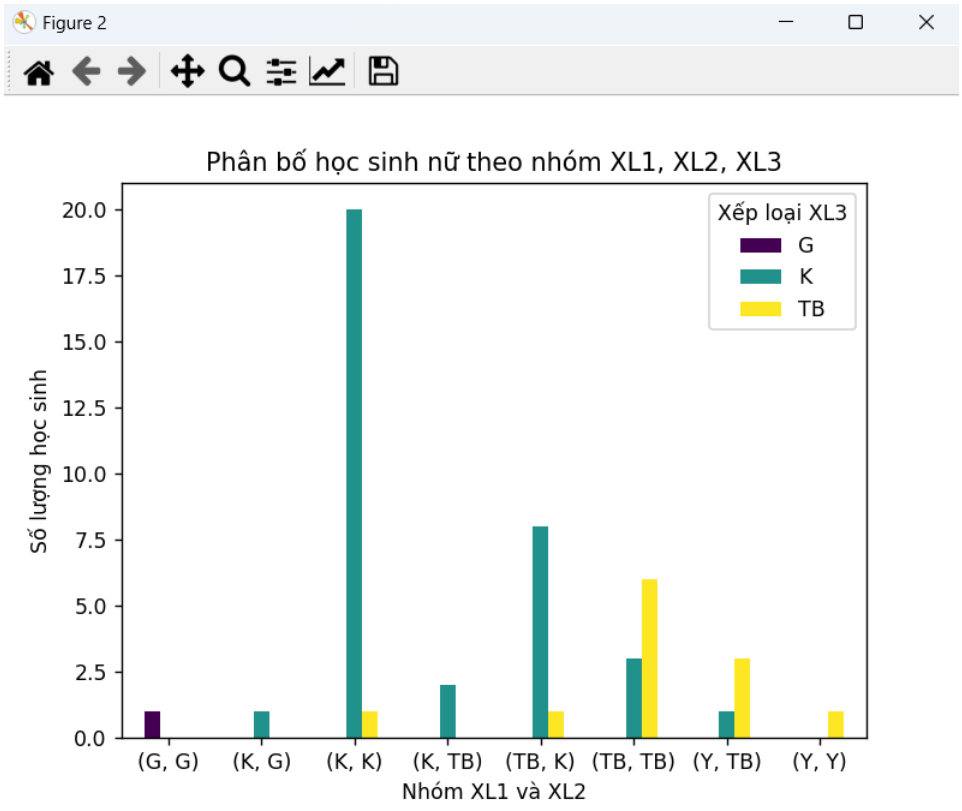
- Lọc dữ liệu giới tính là nữ
- Oy: Chiều cao biểu đồ cột thể hiển số lượng học sinh theo xếp loại
- Màu sắc thể hiện giá trị xếp loại: [Y, TB, K, G, XS]
- Ox: thể hiện nhóm XL1, XL2 và XL3

```python
# Câu 1: Trực quan dữ liệu học sinh nữ trên các nhóm XL1, XL2, XL3 dạng unstacked
# Lọc dữ liệu học sinh nữ
df_female = df[df["GT"] == "F"]

# Nhóm dữ liệu theo XL1, XL2, XL3
grouped_xl = df_female.groupby(["XL1", "XL2", "XL3"]).size().unstack()

# Hiển thị bảng dữ liệu
df_female_summary = grouped_xl
print("\nBảng tần số học sinh nữ theo XL1, XL2, XL3:")
print(df_female_summary)

# Vẽ biểu đồ cột unstacked để hiển thị số lượng học sinh theo xếp loại
plt.figure(figsize=(10, 6))
df_female_summary.plot(kind="bar", stacked=False, colormap="viridis")
plt.title("Phân bố học sinh nữ theo nhóm XL1, XL2, XL3")
plt.xlabel("Nhóm XL1 và XL2")
plt.ylabel("Số lượng học sinh")
plt.legend(title="Xếp loại XL3")
plt.xticks(rotation=0)
plt.show()
```



Châu Gia Kiệt – 2274802010449 – 0103

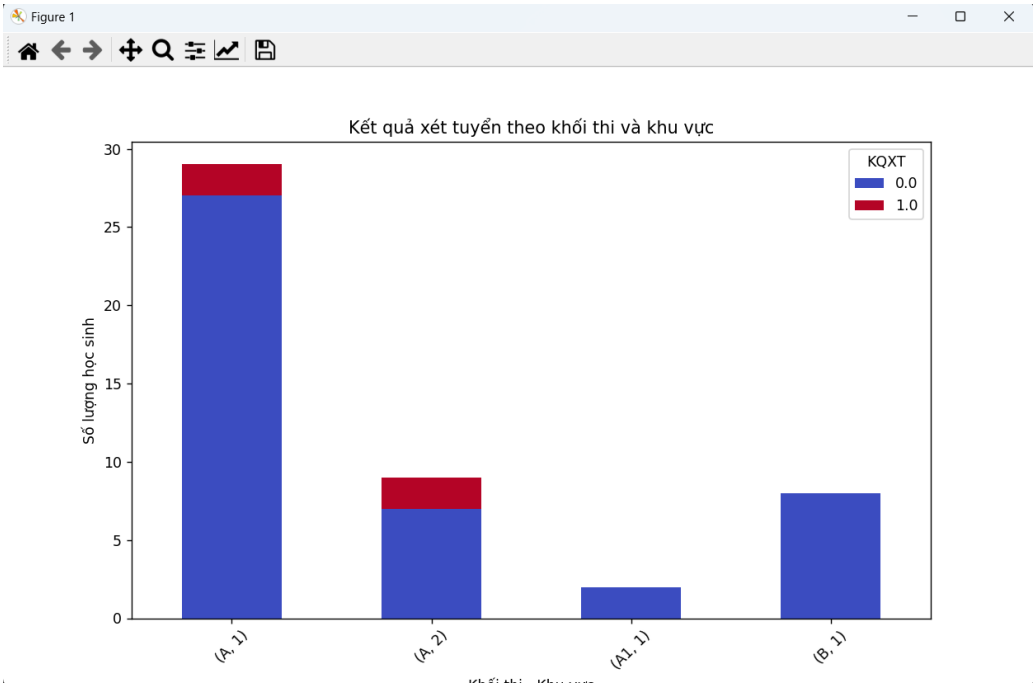
2. Trực quan dữ liệu KQXT trên nhóm học sinh có khối thi A, A1, B thuộc khu vực 1, 2

```python
# Câu 2: Trực quan dữ liệu KQXT trên nhóm học sinh có khối thi A, A1, B thuộc khu vực 1, 2
df_kqxt = df[(df["KT"].isin(["A", "A1", "B"])) & (df["KV"].isin(["1", "2"]))]
kqxt_grouped = df_kqxt.groupby(["KT", "KV"])["KQXT"].value_counts().unstack()

# Hiển thị bảng dữ liệu
print("\nBảng KQXT trên nhóm học sinh có khối thi A, A1, B thuộc khu vực 1, 2:")
print(kqxt_grouped)

# Vẽ biểu đồ cột
kqxt_grouped.plot(kind="bar", stacked=True, figsize=(10, 6), colormap="coolwarm")
plt.title("Kết quả xét tuyển theo khối thi và khu vực")
plt.xlabel("Khối thi - Khu vực")
plt.ylabel("Số lượng học sinh")
plt.legend(title="KQXT")
plt.xticks(rotation=45)
plt.show()
```

```
Bảng KQXT trên nhóm học sinh có khối thi A, A1, B thuộc khu vực 1, 2:
KQXT    0.0  1.0
KT KV
A  1   27.0  2.0
   2    7.0  2.0
A1 1    2.0  NaN
B  1    8.0  NaN
```

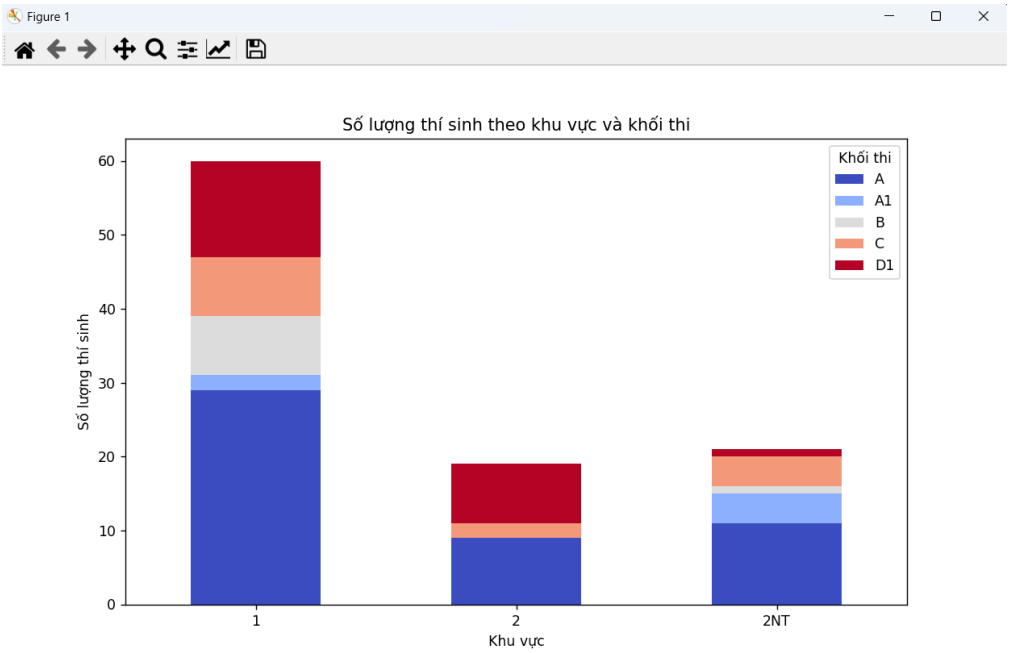


3. Trực quan dữ liệu số lượng thí sinh từng khu vực dựa trên từng nhóm khối thi

```
Bảng số lượng thí sinh theo khu vực và khối thi:
KT      A   A1    B    C    D1
KV
1    29.0  2.0  8.0  8.0  13.0
2     9.0  NaN  NaN  2.0   8.0
2NT  11.0  4.0  1.0  4.0   1.0
```

```python
# Câu 3: Trực quan dữ liệu số lượng thí sinh từng khu vực dựa trên từng nhóm khối thi
df_khuvuc = df.groupby(["KV", "KT"]).size().unstack()
print("\nBảng số lượng thí sinh theo khu vực và khối thi:")
print(df_khuvuc)

df_khuvuc.plot(kind="bar", stacked=True, figsize=(10, 6), colormap="coolwarm")
plt.title("Số lượng thí sinh theo khu vực và khối thi")
plt.xlabel("Khu vực")
plt.ylabel("Số lượng thí sinh")
plt.legend(title="Khối thi")
plt.xticks(rotation=0)
plt.show()
```

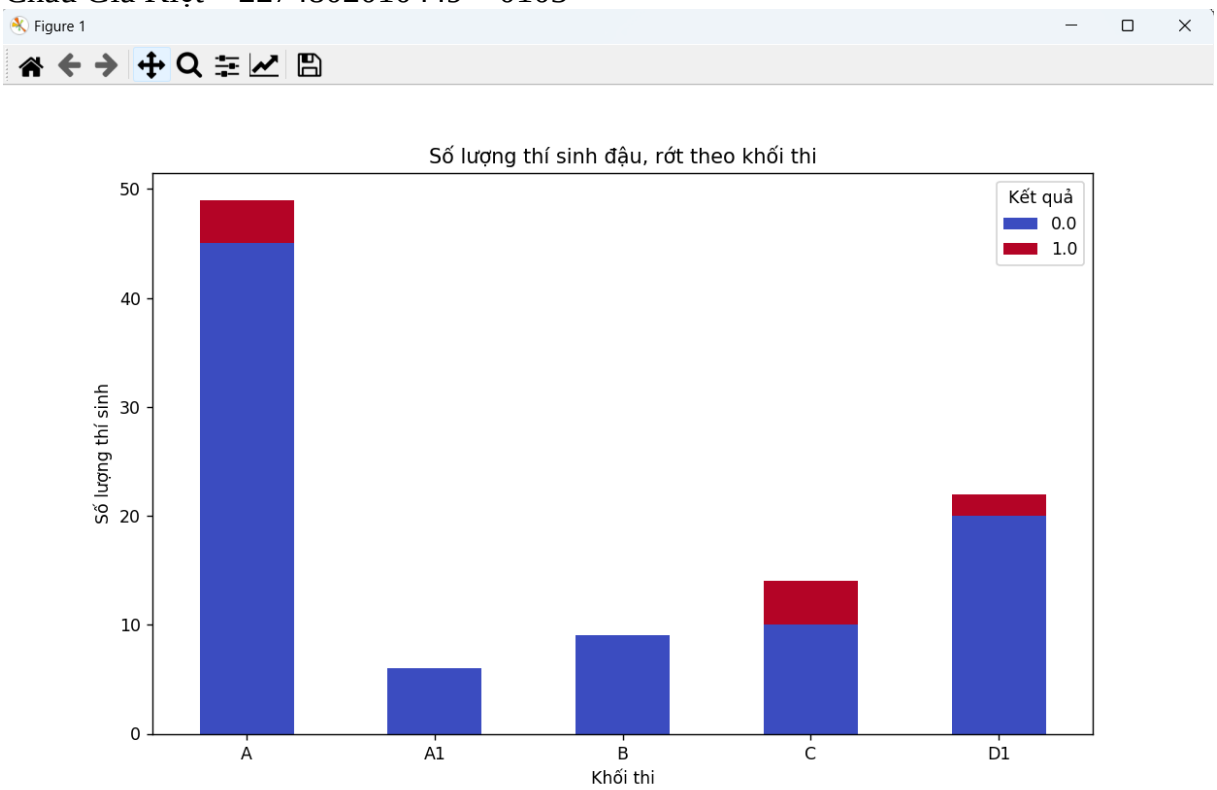


4. Trực quan dữ liệu số lượng thí sinh đậu, rớt trên từng nhóm khối thi

```python
# Câu 4: Trực quan dữ liệu số lượng thí sinh đậu, rớt trên từng nhóm khối thi
df_kq_kt = df.groupby(["KT", "KQXT"]).size().unstack()
print("\nBảng số lượng thí sinh đậu, rớt theo từng nhóm khối thi:")
print(df_kq_kt)

df_kq_kt.plot(kind="bar", stacked=True, figsize=(10, 6), colormap="coolwarm")
plt.title("Số lượng thí sinh đậu, rớt theo khối thi")
plt.xlabel("Khối thi")
plt.ylabel("Số lượng thí sinh")
plt.legend(title="Kết quả")
plt.xticks(rotation=0)
plt.show()
```

```
Bảng số lượng thí sinh đậu, rớt theo từng nhóm khối thi:
KQXT   0.0  1.0
KT
A     45.0  4.0
A1     6.0  NaN
B      9.0  NaN
C     10.0  4.0
D1    20.0  2.0
```

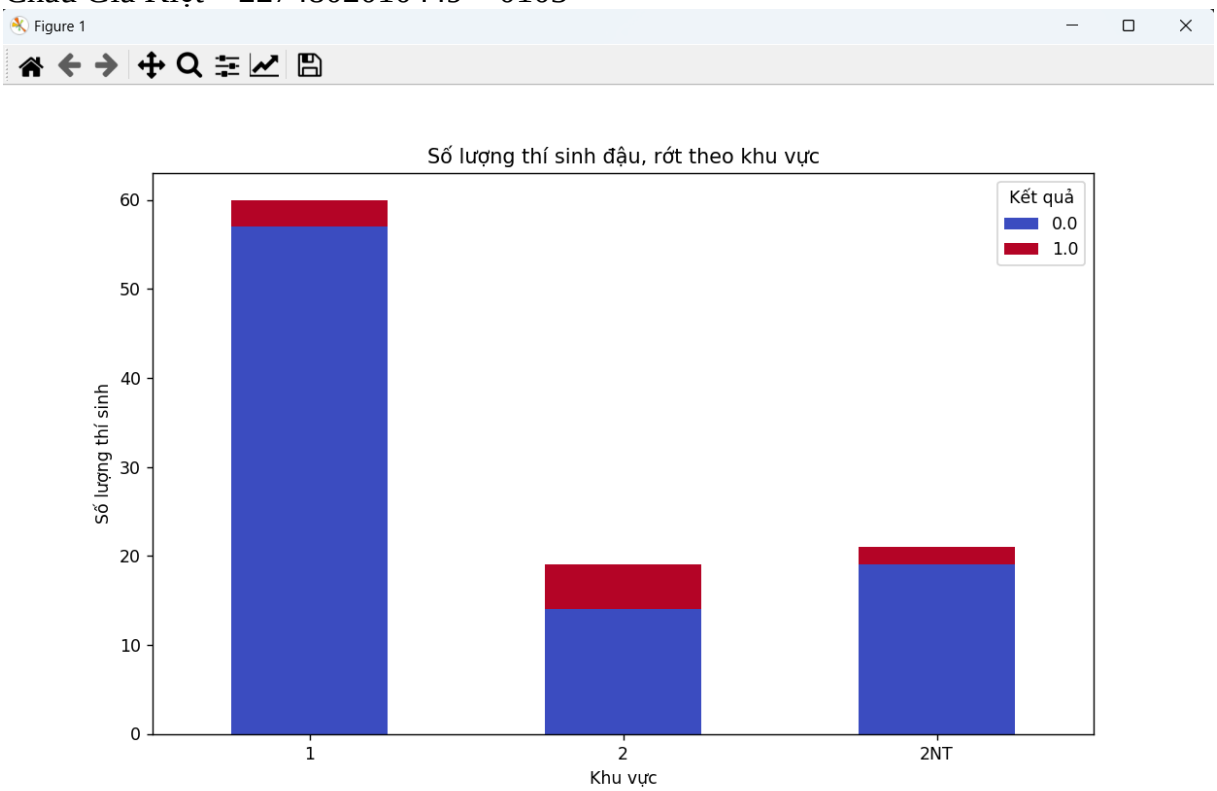


5. Trực quan dữ liệu số lượng thí sinh đậu rớt trên từng nhóm khu vực.

```python
# Câu 5: Trực quan dữ liệu số lượng thí sinh đậu rớt trên từng nhóm khu vực
df_kq_kv = df.groupby(["KV", "KQXT"]).size().unstack()
print("\nBảng số lượng thí sinh đậu, rớt theo khu vực:")
print(df_kq_kv)

df_kq_kv.plot(kind="bar", stacked=True, figsize=(10, 6), colormap="coolwarm")
plt.title("Số lượng thí sinh đậu, rớt theo khu vực")
plt.xlabel("Khu vực")
plt.ylabel("Số lượng thí sinh")
plt.legend(title="Kết quả")
plt.xticks(rotation=0)
plt.show()
```

```
Bảng số lượng thí sinh đậu, rớt theo khu vực:
KQXT  0.0  1.0
KV
1      57    3
2      14    5
2NT    19    2
```

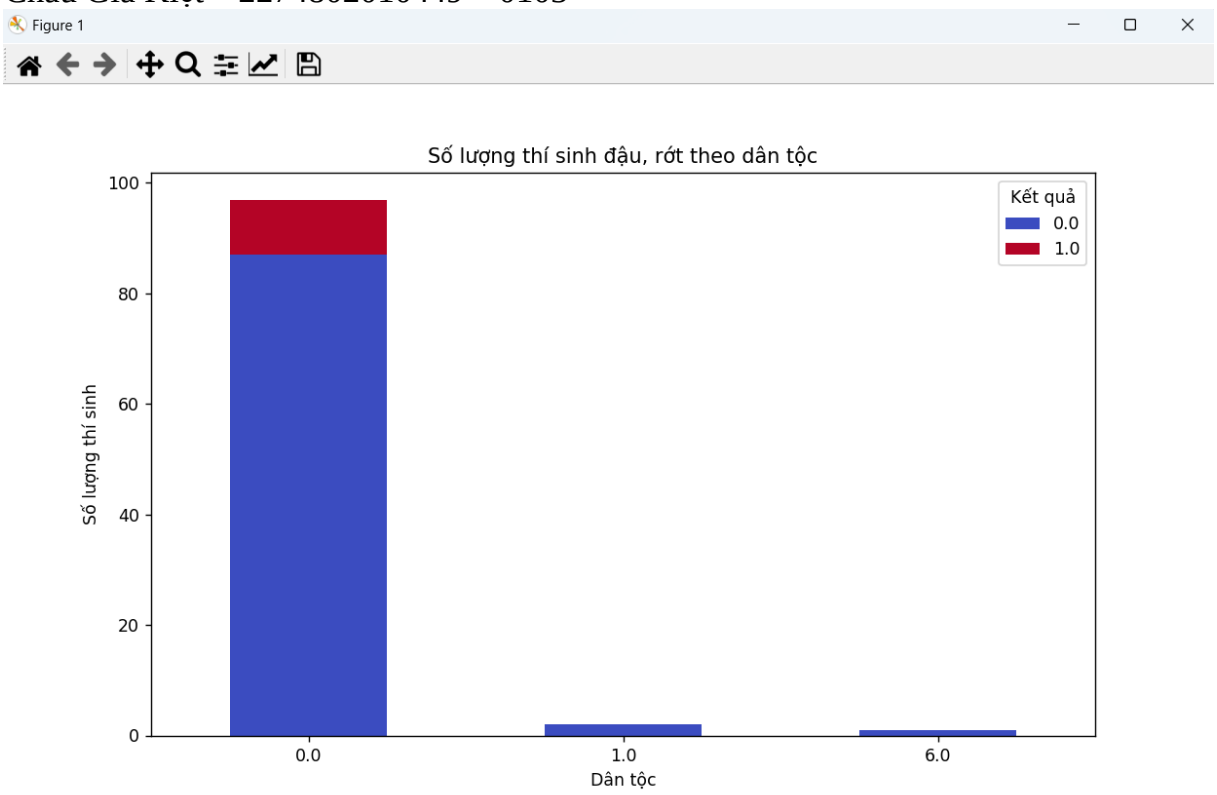


6. Trực quan dữ liệu số lượng thí sinh đậu rớt dựa trên từng nhóm dân tộc

```python
# Câu 6: Trực quan dữ liệu số lượng thí sinh đậu rớt dựa trên từng nhóm dân tộc
df_kq_dt = df.groupby(["DT", "KQXT"]).size().unstack()
print("\nBảng số lượng thí sinh đậu, rớt theo dân tộc:")
print(df_kq_dt)

df_kq_dt.plot(kind="bar", stacked=True, figsize=(10, 6), colormap="coolwarm")
plt.title("Số lượng thí sinh đậu, rớt theo dân tộc")
plt.xlabel("Dân tộc")
plt.ylabel("Số lượng thí sinh")
plt.legend(title="Kết quả")
plt.xticks(rotation=0)
plt.show()
```

```
Bảng số lượng thí sinh đậu, rớt theo dân tộc:
KQXT   0.0   1.0
DT
0.0   87.0  10.0
1.0    2.0   NaN
6.0    1.0   NaN
```

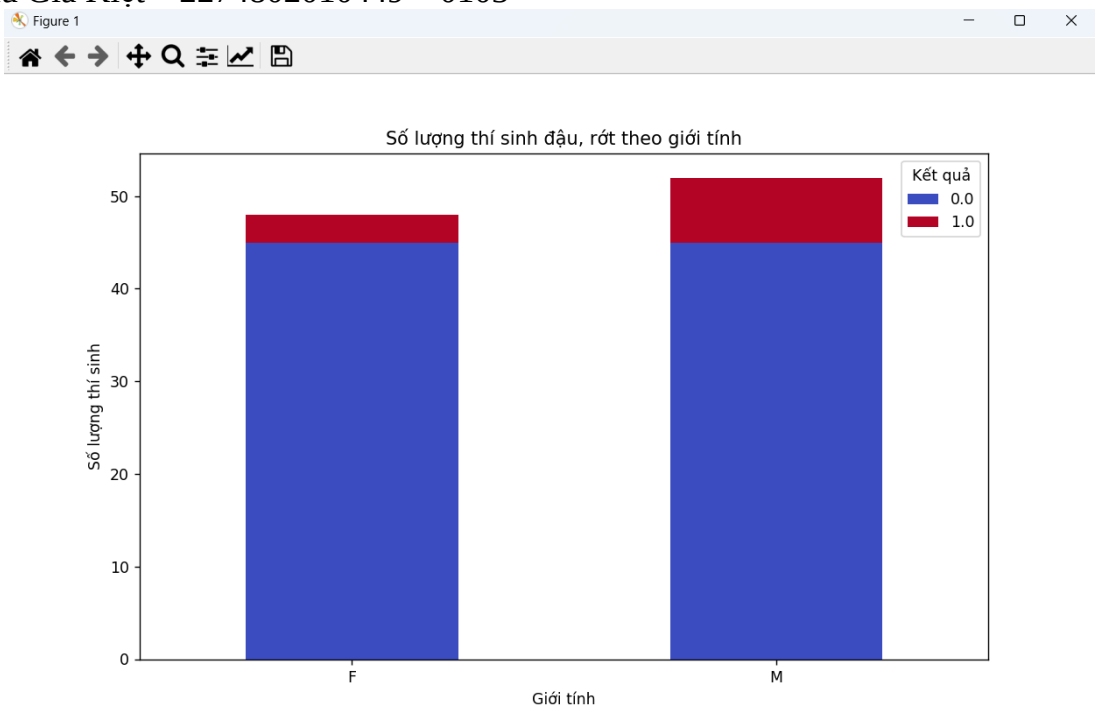


7. Trực quan dữ liệu số lượng thí sinh đậu rớt dựa trên từng nhóm giới tính.

```python
# Câu 7: Trực quan dữ liệu số lượng thí sinh đậu rớt dựa trên từng nhóm giới tính
df_kq_gt = df.groupby(["GT", "KQXT"]).size().unstack()
print("\nBảng số lượng thí sinh đậu, rớt theo giới tính:")
print(df_kq_gt)

df_kq_gt.plot(kind="bar", stacked=True, figsize=(10, 6), colormap="coolwarm")
plt.title("Số lượng thí sinh đậu, rớt theo giới tính")
plt.xlabel("Giới tính")
plt.ylabel("Số lượng thí sinh")
plt.legend(title="Kết quả")
plt.xticks(rotation=0)
plt.show()
```

```
Bảng số lượng thí sinh đậu, rớt theo giới tính:
KQXT  0.0  1.0
GT
F      45    3
M      45    7
```



Phần 4: Trực quan hóa dữ liệu nâng cao

1. Vẽ biểu đồ đường Simple cho biến T1

2. Hãy tạo biến phân loại (phanlopt1) cho môn toán (T1) như sau:

a. Từ 0 đến dưới 5 = kém (ký hiệu “k”)

b. Từ 5 đến dưới 7 = trung bình (ký hiệu “tb”)

c. Từ 7 đến dưới 8 = khá (ký hiệu “k”)

d. Từ 8 trở lên = giỏi (ký hiệu “g”)

3. Lập bảng tần số cho biến phanloait1

4. Vẽ biểu đồ đường Multiple Line cho biến T1 được phân loại bởi biến phanlopt1

5. Vẽ biểu đồ Drop-line cho biến T1 được phân loại bởi biến phanlopt1

Châu Gia Kiệt – 2274802010449 – 0103

Phần 5: Mô tả dữ liệu và khảo sát dạng phân phối

1. Hãy mô tả và khảo sát phân phối cho biến T1

Gợi ý:

- Mô tả độ tập trung và phân tán của dữ liệu T1
- Vẽ biểu đồ Box-Plot và xác định các 10 đại lượng trong biểu đồ đó
- Mô tả hình dáng lệch của phân phối T1 dựa vào các đại lượng hướng tâm
- Vẽ biểu đồ Histogram biểu thị hình dáng phân phối
- Mô tả các đặc trưng của phân phối, mức độ lệch và mức độ nhọn
- Kiểm chứng phân phối chuẩn QQ-Plot
- Nhận xét và đánh giá về phân phối của T1

2. Hãy mô tả và khảo sát phân phối cho biến T1 trên từng nhóm phân lớp (phanlopT1)

Gợi ý:

- Trực quan hóa biểu đồ Box-plot, histogram và QQ-plot theo phân nhóm là giá trị của ‘phanlopT1’.

3. Hãy khảo sát tương quan giữa biến DH1 theo biến T1

Gợi ý:

- Nhận xét giá trị Covariance hoặc Correlation
- Vẽ biểu đồ Scatter thể hiện liên hệ của biến phụ thuộc DH1 theo biến độc lập T1

4. Hãy khảo sát tương quan giữa biến DH1 theo biến T1 trên từng nhóm khu vực

5. Hãy khảo sát tương quan giữa các biến DH1, DH2, DH3

Gợi ý:

- Nhận xét ma trận hiệp phương sai hoặc ma trận tương quan
- Vẽ biểu đồ Scatter giữa các biến