In [1]:
```
import pandas as pd
```

In [3]:
```
df = pd.read_csv('Downloads/synthetic_handwritten_data.csv')
```

In [5]:
```
print(df.head())
```

```
   pixel_1  pixel_2  pixel_3  pixel_4  pixel_5  pixel_6  pixel_7  pixel_8  \
0      255      198      245      111       78      162       71      198
1       89       12      102      124       47      122      223       73
2      116      211      170      138      127       35      219      130
3      117      137       85       16      206      163      163       92
4      118      114      192       40      110       98       75       70

   pixel_9  pixel_10  ...  pixel_776  pixel_777  pixel_778  pixel_779  \
0      108       129  ...        166        254         53         29
1        8       196  ...          6        145         77        164
2      225       151  ...         75        184        169         19
3       57       214  ...        228        141        187         79
4       21        57  ...         11          1        200        248

   pixel_780  pixel_781  pixel_782  pixel_783  pixel_784  label
0        254          8         32         92         17      E
1         68         35         29        166        210      I
2        164          9        203        149        238      C
3         36        236         92         12         44      E
4        190        141        128         90         78      G

[5 rows x 785 columns]
```

In [7]:
```
print(df.describe())
```

```
           pixel_1      pixel_2      pixel_3      pixel_4      pixel_5  \
count  1000.000000  1000.000000  1000.000000  1000.000000  1000.000000
mean    126.685000   126.017000   126.953000   126.574000   125.086000
std      73.322206    73.848726    74.194188    73.679245    75.378356
min       0.000000     0.000000     0.000000     0.000000     0.000000
25%      65.750000    60.000000    61.000000    61.000000    57.000000
50%     126.000000   129.000000   129.000000   126.500000   126.000000
75%     189.000000   189.000000   191.000000   191.250000   189.000000
max     255.000000   255.000000   255.000000   255.000000   255.000000

           pixel_6      pixel_7      pixel_8      pixel_9     pixel_10  ...  \
count  1000.000000  1000.000000  1000.000000  1000.000000  1000.000000  ...
mean    126.992000   134.137000   125.383000   128.759000   125.146000  ...
std      74.427397    74.032297    74.575075    72.529093    74.080785  ...
min       0.000000     0.000000     0.000000     0.000000     0.000000  ...
25%      61.000000    73.000000    61.000000    67.000000    60.000000  ...
50%     127.000000   137.000000   124.500000   127.000000   124.000000  ...
75%     193.000000   198.000000   188.000000   192.250000   190.000000  ...
max     255.000000   255.000000   255.000000   255.000000   255.000000  ...

          pixel_775    pixel_776    pixel_777    pixel_778   pixel_779  \
count  1000.000000  1000.000000  1000.000000  1000.000000  1000.00000
mean    126.689000   129.011000   122.397000   128.564000   124.88500
std      74.070539    74.959828    73.353897    75.163272    71.18104
min       0.000000     0.000000     0.000000     0.000000     0.00000
25%      62.000000    66.000000    59.000000    65.000000    63.00000
50%     127.000000   128.000000   120.000000   130.000000   128.00000
75%     192.000000   195.000000   185.000000   194.000000   183.25000
max     255.000000   255.000000   255.000000   255.000000   255.00000

          pixel_780    pixel_781    pixel_782    pixel_783    pixel_784
count  1000.000000  1000.000000  1000.000000  1000.000000  1000.000000
mean    129.571000   128.688000   126.872000   126.328000   122.857000
std      74.774449    74.660665    73.892989    73.351652    73.787907
min       0.000000     0.000000     0.000000     0.000000     0.000000
25%      63.750000    62.000000    64.750000    65.000000    56.000000
50%     131.000000   132.000000   127.500000   124.000000   123.000000
75%     196.000000   190.000000   191.000000   192.250000   186.000000
max     255.000000   255.000000   255.000000   255.000000   255.000000

[8 rows x 784 columns]
```

In [9]:
```
print(df.isnull().sum())
```

```
pixel_1      0
pixel_2      0
pixel_3      0
pixel_4      0
pixel_5      0
            ..
pixel_781    0
pixel_782    0
pixel_783    0
pixel_784    0
label        0
Length: 785, dtype: int64
```

In [11]:
```
import matplotlib.pyplot as plt
import numpy as np
```

In [17]:
```
def display_image(row):
    pixels = row[:-1].values  # Exclude label (last column)
    image = pixels.reshape(28, 28)  # Reshape to 28x28
    plt.imshow(image, cmap='gray')
    plt.title(f"Label: {row['label']}")
    plt.show()
```

In [19]:
```
label_counts = df['label'].value_counts()
print(label_counts)
```

```
label
B    121
G    114
E    111
H    103
D    101
C     96
A     93
J     93
I     84
F     84
Name: count, dtype: int64
```

In [21]:
```
X = df.drop('label', axis=1)  # Pixel values
y = df['label']  # Labels
```

In [23]:
```
print(X.shape, y.shape)
```

```
(1000, 784) (1000,)
```

In [25]:
```
X_normalized = X / 255.0
```

In [27]:
```
print(X_normalized.iloc[0])
```

```
pixel_1      1.000000
pixel_2      0.776471
pixel_3      0.960784
pixel_4      0.435294
pixel_5      0.305882
               ...
pixel_780    0.996078
pixel_781    0.031373
pixel_782    0.125490
pixel_783    0.360784
pixel_784    0.066667
Name: 0, Length: 784, dtype: float64
```

In [29]:
```
df_shuffled = df.sample(frac=1).reset_index(drop=True)
```

In [31]:
```
print(df_shuffled.head())
```

```
   pixel_1  pixel_2  pixel_3  pixel_4  pixel_5  pixel_6  pixel_7  pixel_8  \
0      230      213       25       77      146       78      200      176
1       74      124        8      107      133      112      217       47
2       71       34      173      177      127      216       40      137
3       96      114       10      132      117      205      106       28
4       80       40       80       68      115      117      163       60

   pixel_9  pixel_10  ...  pixel_776  pixel_777  pixel_778  pixel_779  \
0      216        19  ...        204        195          9        203
1        0       130  ...        125         39        144        151
2       33        97  ...        112        224        175        133
3      169       167  ...         46         51         55        170
4       50       153  ...        114        159         55         70

   pixel_780  pixel_781  pixel_782  pixel_783  pixel_784  label
0        178        221         67        228        241      D
1         29         94        219        121         50      B
2         81        143         76         86        219      B
3         73        185        115        227        131      H
4        199         55        226         95        177      H

[5 rows x 785 columns]
```

In [33]:
```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X_normalized, y, test_size=0.2,

print(X_train.shape, X_test.shape, y_train.shape, y_test.shape)
```

```
(800, 784) (200, 784) (800,) (200,)
```

In [35]:
```
df_normalized = pd.concat([X_normalized, y], axis=1)
df_normalized.to_csv('processed_handwritten_data.csv', index=False)
```

In [ ]:
```
```