```
import pandas as pd
import numpy as np

# Step 1: Create sample data with missing values
data = {
    "Student_ID": list(range(101, 111)),
    "Name": ["Alice", "Bob", "Charlie", "David", "Eva", "Frank", "Grace", "Helen", "Ian", "Jane"],
    "Math": [85, 90, 78, np.nan, 88, 76, np.nan, 95, 89, 91],
    "Science": [89, 94, np.nan, 90, 91, 83, 87, 96, np.nan, 93],
    "English": [87, 85, 80, 88, np.nan, 79, 85, 97, 86, 92]
}

# Step 2: Create DataFrame and save to CSV
df = pd.DataFrame(data)
df.to_csv("student_scores.csv", index=False)

# Step 3: Fill missing values with column-wise average
df_filled = df.copy()
df_filled[['Math', 'Science', 'English']] = df_filled[['Math', 'Science', 'English']].fillna(df_filled[['Math', 'Science', 'English']].me

# Step 4: Filter students scoring >75 in all subjects
high_achievers = df_filled[
    (df_filled['Math'] > 75) &
    (df_filled['Science'] > 75) &
    (df_filled['English'] > 75)
]

# Step 5: Save cleaned and filtered data to a new CSV
high_achievers.to_csv("cleaned_student_scores.csv", index=False)

print("CSV files created: student_scores.csv and cleaned_student_scores.csv")
```

```
CSV files created: student_scores.csv and cleaned_student_scores.csv
```

```
import pandas as pd

# Step 1: Read the CSV file
df = pd.read_csv("student_scores.csv")

# Step 2: Fill missing subject marks with column-wise averages
df[['Math', 'Science', 'English']] = df[['Math', 'Science', 'English']].fillna(
    df[['Math', 'Science', 'English']].mean()
)

# Step 3: Filter students who scored more than 75 in all subjects
high_achievers = df[
    (df['Math'] > 75) &
    (df['Science'] > 75) &
    (df['English'] > 75)
]

# Step 4: Display the result
print("High Achievers (scored >75 in all subjects):")
print(high_achievers)

# Step 5: Save the filtered data to a new CSV
high_achievers.to_csv("cleaned_student_scores.csv", index=False)
```

```
High Achievers (scored >75 in all subjects):
   Student_ID     Name  Math  Science    English
0         101    Alice  85.0   89.000  87.000000
1         102      Bob  90.0   94.000  85.000000
2         103  Charlie  78.0   90.375  80.000000
3         104    David  86.5   90.000  88.000000
4         105      Eva  88.0   91.000  86.555556
5         106    Frank  76.0   83.000  79.000000
6         107    Grace  86.5   87.000  85.000000
7         108    Helen  95.0   96.000  97.000000
8         109      Ian  89.0   90.375  86.000000
9         110     Jane  91.0   93.000  92.000000
```

```
import pandas as pd


data = {
    "Patient_ID": [1, 2, 3, 4, 5, 6, 7],
    "Name": ["Anita", "Ramesh", "Sita", "John", "Radha", "Kavita", "Asha"],
    "Age": [55, 40, 72, 65, None, 58, 62],
    "Gender": ["Female", "Male", "Female", "Male", "Female", "Female", "Female"],
    "Diagnosis": ["Diabetes", "Hypertension", "Diabetes", "Asthma", "Diabetes", "Heart Disease", "Diabetes"]
}

df = pd.DataFrame(data)

# Save as patients.csv
df.to_csv("patients.csv", index=False)

print("patients.csv file created successfully!")
```

```
patients.csv file created successfully!
```

```
import pandas as pd

# Load the dataset
df = pd.read_csv("patients.csv")

# Replace missing Age values with median age
median_age = df['Age'].median()
df['Age'] = df['Age'].fillna(median_age)

# Filter female patients aged above 50 diagnosed with Diabetes
filtered_patients = df[
    (df['Gender'].str.lower() == 'female') &
    (df['Age'] > 50) &
    (df['Diagnosis'].str.contains('diabetes', case=False, na=False))
]

# Display the filtered data
print("Filtered elderly female patients diagnosed with Diabetes:")
print(filtered_patients)

# Optional: Save filtered data to a new CSV
filtered_patients.to_csv("filtered_elderly_female_diabetes_patients.csv", index=False)
```

```
Filtered elderly female patients diagnosed with Diabetes:
   Patient_ID   Name   Age  Gender Diagnosis
0           1  Anita  55.0  Female  Diabetes
2           3   Sita  72.0  Female  Diabetes
4           5  Radha  60.0  Female  Diabetes
6           7   Asha  62.0  Female  Diabetes
```