Filter rows based on a condition:

In [ ]:
```
filtered_data = df[df['column_name'] == condition]
```

Filter rows based on multiple conditions:

In [ ]:
```
filtered_data = df[(df['column1'] == condition1) & (df['column2'] == condition2)]
```

Filter rows based on partial string match:

In [ ]:
```
filtered_data = df[df['column_name'].str.contains('partial_string')]
```

Filter rows based on exact string match:

In [ ]:
```
filtered_data = df[df['column_name'] == 'exact_string']
```

Filter rows based on values in a list:

In [ ]:
```
filtered_data = df[df['column_name'].isin(['value1', 'value2', 'value3'])]
```

Filter rows based on missing values (NaN):

In [ ]:
```
filtered_data = df[df['column_name'].isnull()]
```

Filter rows based on non-missing values:

In [ ]:
```
filtered_data = df[df['column_name'].notnull()]
```

Filter rows based on numeric conditions:

In [ ]:
```
filtered_data = df[df['column_name'] > 10]
```

Filter rows based on dates:

In [ ]:
```
filtered_data = df[df['date_column'] > '2022-01-01']
```

Filter rows based on multiple conditions with OR logic:

In [ ]:
```
filtered_data = d[(df['column1'] == condition1) | (df['column2'] == condition2)]
```

Filter rows based on the absence of a value:

In [ ]:
```
filtered_data = df[df['column_name'].isna()]
```

Filter rows based on the presence of a value:

In [ ]:
```
filtered_data = df[df['column_name'].notna()]
```

Filter rows based on the start of a string:

In [ ]:
```
filtered_data = df[df['column_name'].str.startswith('start_string')]
```

Filter rows based on the end of a string:

In [ ]:
```
filtered_data = df[df['column_name'].str.endswith('end_string')]
```

Filter rows based on a case-insensitive condition:

In [ ]:
```
filtered_data = df[df['column_name'].str.contains('partial_string', case=False)]
```

Filter rows based on multiple conditions with OR and AND logic:

In [ ]:
```
filtered_data = df[((df['column1'] == condition1) | (df['column2'] == condition2)) & (
```

Filter rows based on a condition with a wildcard character:

In [ ]:
```
filtered_data = df[df['column_name'].str.contains('part_of_string.*')]
```

Filter rows based on a condition with regular expressions:

In [ ]:
```
import re
filtered_data = df[df['column_name'].str.contains(r'regex_pattern')]
```

Filter rows based on a condition with a custom function:

In [ ]:
```
filtered_data = df[df['column_name'].apply(custom_function)]
```

Filter rows based on the presence of any non-zero values:

In [ ]:
```
filtered_data = df[df.any(axis=1)]
```

Filter rows based on the absence of any non-zero values:

In [ ]:
```
filtered_data = df[~df.any(axis=1)]
```

Filter rows based on the presence of any duplicate values:

In [ ]:
```
filtered_data = df[df.duplicated()]
```

Filter rows based on the absence of any duplicate values:

In [ ]:
```
filtered_data = df[~df.duplicated()]
```

Filter rows based on the rank of values:

In [ ]:
```
filtered_data = df[df['column_name'].rank(method='dense') <= rank_threshold]
```

Filter rows based on the index:

In [ ]:
```
filtered_data = df.iloc[index_start:index_end]
```

In [ ]:
```

```

In [ ]:
```

```