pandas

Data Types and Missing Values

5

PD

By Abdul Rauf Jatoi

Datatypes

pandas

Pandas supports various data types: int64, float64, object (string), bool, datetime64, timedelta64, category, and complex. These types enable efficient data manipulation and analysis.

```python
import pandas as pd

# Creating a DataFrame with different data types
data = {
    'Integer': [1, 2, 3],
    'Float': [1.1, 2.2, 3.3],
    'String': ['a', 'b', 'c'],
    'Boolean': [True, False, True],
    'Date': pd.to_datetime(['2024-01-01', '2024-02-01', '2024-03-01'])
}

df = pd.DataFrame(data)

print(df.dtypes)
```

```
Integer                int64
Float                  float64
String                 object
Boolean                bool
Date           datetime64[ns]
dtype: object
```

Convertin

pandas

```python
import pandas as pd
# Creating a DataFrame with integer data
df = pd.DataFrame({'Integers': [1, 2, 3, 4]})
# Converting integer to float
df['Floats'] = df['Integers'].astype(float)
print(df)

```

	Integers	Floats
0	1	1.0
1	2	2.0
2	3	3.0
3	4	4.0

```python
import pandas as pd
# Creating a DataFrame with float data
df = pd.DataFrame({'Floats': [1.1, 2.2, 3.3, 4.4]})
# Converting float to integer (truncates decimal part)
df['Integers'] = df['Floats'].astype(int)
print(df)

```

	Floats	Integers
0	1.1	1
1	2.2	2
2	3.3	3
3	4.4	4

Missing values

pandas

Pandas handles missing values with NaN. Use methods like dropna() to remove, fillna() to replace, and isna() to detect missing values for effective data cleaning and handling.

```python
import pandas as pd
import numpy as np
# Creating a DataFrame with missing values
data = {
    'A': [1, 2, np.nan, 4],
    'B': [5, np.nan, np.nan, 8],
    'C': ['x', 'y', 'z', np.nan]
}
df = pd.DataFrame(data)
print("Original DataFrame:")
print(df)
```

```
Original DataFrame:
     A    B    C
0  1.0  5.0    x
1  2.0  NaN    y
2  NaN  NaN    z
3  4.0  8.0  NaN
```

```python
import numpy as np
# Creating a DataFrame with missing values
data = {
    'A': [1, 2, np.nan, 4],
    'B': [5, np.nan, np.nan, 8],
    'C': ['x', 'y', 'z', np.nan]
}
df = pd.DataFrame(data)

# Filling missing values with a specific value
df_filled_value = df.fillna(value={'A': 0, 'B': 0, 'C': 'unknown'})

print("\nDataFrame after filling missing values with specific values:")
print(df_filled_value)

# Forward filling missing values
df_filled_ffill = df.fillna(method='ffill')

print("\nDataFrame after forward filling missing values:")
print(df_filled_ffill)

# Backward filling missing values
df_filled_bfill = df.fillna(method='bfill')

print("\nDataFrame after backward filling missing values:")
print(df_filled_bfill)
```

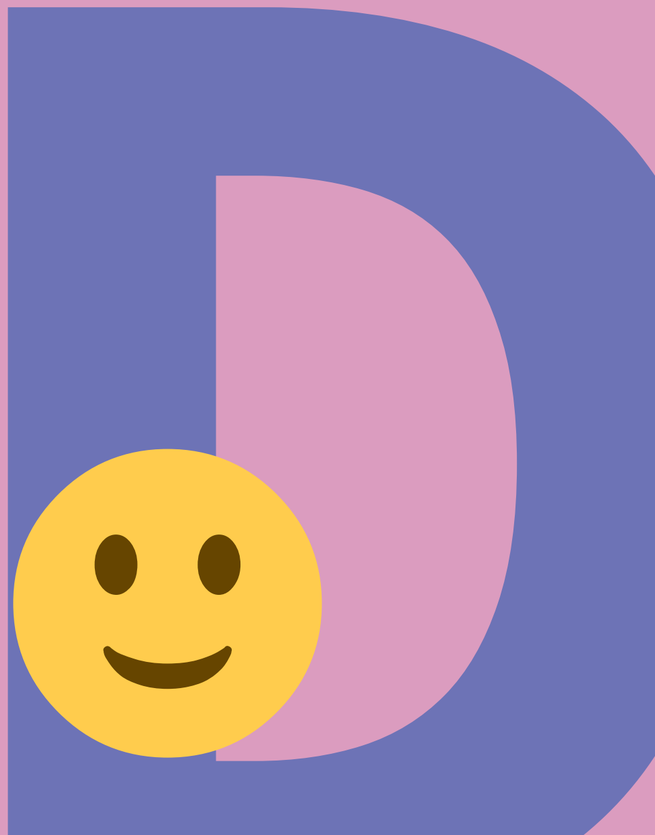
PROBLEMS 13 DEBUG CONSOLE TERMINAL PORTS

TERMINAL

```
PS D:\Documents\GitHub\Pandas\vid5> python ex9.py

DataFrame after filling missing values with specific values:
     A    B        C
0  1.0  5.0        x
1  2.0  0.0        y
2  0.0  0.0        z
3  4.0  8.0  unknown
D:\Documents\GitHub\Pandas\vid5\ex9.py:18: FutureWarning: DataFrame.fillna with 'method' is deprecated and will raise in a future version. Use obj.ffill() or obj.bfill() instead.
  df_filled_ffill = df.fillna(method='ffill')

DataFrame after forward filling missing values:
     A    B  C
0  1.0  5.0  x
1  2.0  5.0  y
2  2.0  5.0  z
3  4.0  8.0  z
D:\Documents\GitHub\Pandas\vid5\ex9.py:24: FutureWarning: DataFrame.fillna with 'method' is deprecated and will raise in a future version. Use obj.ffill() or obj.bfill() instead.
  df_filled_bfill = df.fillna(method='bfill')

DataFrame after backward filling missing values:
     A    B    C
0  1.0  5.0    x
1  2.0  8.0    y
2  4.0  8.0    z
3  4.0  8.0  NaN
```

pandas

By fillin vals



pandas

By Interpolating

```python
import pandas as pd
import numpy as np
# Creating a DataFrame with missing values
data = {
    'A': [1, 2, np.nan, 4],
    'B': [5, np.nan, np.nan, 8],
    'C': ['x', 'y', 'z', np.nan]
}
df = pd.DataFrame(data)
# Interpolating missing values
df_interpolated = df.interpolate()

print("\nDataFrame after interpolating missing values:")
print(df_interpolated)
```

OBLEMS 15 DEBUG CONSOLE TERMINAL PORTS

TERMINAL

```
PS D:\Documents\GitHub\Pandas\vid5> python ex10.py
D:\Documents\GitHub\Pandas\vid5\ex10.py:11: FutureWarning: DataFrame.interpolate with
ects(copy=False) before interpolating instead.
  df_interpolated = df.interpolate()

DataFrame after interpolating missing values:
     A    B    C
0  1.0  5.0    x
1  2.0  6.0    y
2  3.0  7.0    z
3  4.0  8.0  NaN
PS D:\Documents\GitHub\Pandas\vid5>
```

pandas

Thank you

5

By Abdul Rauf Jatoi