Join in pandas

In pandas, the join() function is used to join two or more DataFrame objects based on common columns or indexes. It allows you to combine data from multiple DataFrames into a single DataFrame based on matching values in the specified columns or indexes.

Here's an example to demonstrate how the join() function works:

Let's say we have two DataFrames, df1 and df2, representing sales data for different regions:

```python
import pandas as pd

# Creating DataFrame 1
data1 = {'Region': ['North', 'South', 'East', 'Wes
t'],
          'Sales': [1000, 1500, 1200, 1800]}
df1 = pd.DataFrame(data1)

# Creating DataFrame 2
data2 = {'Region': ['North', 'South', 'East', 'Wes
t'],
          'Profit': [200, 300, 250, 400]}
df2 = pd.DataFrame(data2)
```

DataFrame df1:

```
   Region  Sales
0   North   1000
1   South   1500
2    East   1200
3    West   1800
```

DataFrame df2:

```
   Region  Profit
0   North     200
1   South     300
2    East     250
3    West     400
```

Now, we can use the join() function to join these two DataFrames based on the 'Region' column:

```
df3 = df1.join(df2.set_index('Region'), on='Regio
n')
```

DataFrame df3 after joining:

```
   Region  Sales  Profit
0   North   1000     200
1   South   1500     300
2    East   1200     250
3    West   1800     400
```

n the above example, we used the join() function to join df1 and df2 based on the 'Region' column. The resulting DataFrame df3 contains all the columns from both df1 and df2, with matching rows joined together. The on='Region' parameter specifies the column to be used for joining.

Note that by default, the join() function performs a left join, meaning all the rows from the left DataFrame (df1 in this case) are included in the result, and only the matching rows from the right DataFrame (df2 in this case) are added. Other join types like inner join, right join, and outer join can also be performed using the join() function by specifying the how parameter.