

# How can I split a column of lists into multiple columns in Pandas?

Authored by  
**stats writer**

June 27, 2024

## RECOMMENDED CITATION

stats writer (2024). *How can I split a column of lists into multiple columns in Pandas?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=154587>

Splitting a column of lists into multiple columns in Pandas is a process that involves breaking down a single column containing lists of data into separate columns. This can be done by using the "str.split" function in Pandas, which allows for the separation of the lists based on a defined delimiter. This allows for the data to be organized and analyzed more efficiently, making it easier to work with and manipulate. By splitting a column of lists into multiple columns, it can provide a clearer and more organized structure for the data, making it easier to extract insights and information.

## **Pandas: Split a Column of Lists into Multiple Columns**

**You can use the following basic syntax to split a column of lists into multiple columns in a pandas DataFrame:**

```
#split column of lists into two new columns  
split = pd.DataFrame(df.to_list(), columns = )
```

```
#join split columns back to original DataFrame  
df = pd.concat(, axis=1)
```

**The following example shows how to use this syntax in practice.**

**Example: Split Column of Lists into Multiple Columns in Pandas**

**Suppose we have the following pandas DataFrame in which the column called points contains lists of values:**

```
import pandas as pd
```

```
#create DataFrame  
df = pd.DataFrame({'team': ,  
'points': , , , ]})
```

```
#view DataFrame  
print(df)
```

```
team points  
0 Mavs  
1 Heat  
2 Kings  
3 Suns
```

We can use the following syntax to create a new DataFrame in which the points column is split into two new columns called game1 and game2:

```
#split column of lists into two new columns  
split = pd.DataFrame(df.to_list(), columns = )
```

```
#view DataFrame  
print(split)
```

```
game1 game2  
0 99 105
```

1 94 113

2 99 97

3 87 95

If we'd like, we can then join this split DataFrame back with the original DataFrame by using the `concat()` function:

```
#join split columns back to original DataFrame
```

```
df = pd.concat(, axis=1)
```

```
#view updated DataFrameprint(df)
```

```
team points game1 game2
```

```
0 Mavs 99 105
```

```
1 Heat 94 113
```

```
2 Kings 99 97
```

```
3 Suns 87 95
```

Lastly, we can drop the original points column from the DataFrame if we'd like:

```
#drop original points column
```

```
df = df.drop('points', axis=1)
```

```
#view updated DataFrame  
print(df)
```

```
team game1 game2  
0 Mavs 99 105  
1 Heat 94 113  
2 Kings 99 97  
3 Suns 87 95
```

The end result is a DataFrame in which the original points column of lists is now split into two new columns called game1 and game2.

**Note:** If your column of lists has an uneven number of values in each list, pandas will simply fill in missing values with NaN values when splitting the lists into columns.

The following tutorials explain how to perform other common operations in pandas: