

# How can I select columns in Pandas based on their data type?

Authored by  
**stats writer**

June 26, 2024

## RECOMMENDED CITATION

stats writer (2024). *How can I select columns in Pandas based on their data type?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=154025>

Selecting columns in Pandas based on their data type is a useful functionality for data analysis and manipulation. This feature allows the user to filter and extract specific columns from a dataset based on their data type, such as numerical or categorical. To do so, the user can use the "select\_dtypes" method in Pandas, which enables them to specify the desired data types and retrieve only those columns. This can be particularly helpful in large datasets, where it is necessary to focus on specific types of data for analysis. By utilizing this feature, the user can efficiently and accurately select and work with the desired columns in their dataset.

## Pandas: Select Columns by Data Type

You can use the following methods to select columns in a pandas DataFrame that are equal to a specific data type:

### Method 1: Select Columns Equal to Specific Data Type

#select all columns that have an int or float data type  
`df.select_dtypes(include=)`

### Method 2: Select Columns Not Equal to Specific Data Type

#select all columns that don't have a bool or object data type  
`df.select_dtypes(exclude=)`

The following examples show how to use each method with the following pandas DataFrame:

```
import pandas as pd

#create DataFrame
df = pd.DataFrame({'team': ,
'points': ,
'assists': ,
'minutes': ,
'all_star': })

#view DataFrame
print(df)

team points assists minutes all_star
0 A 18 5 10.1 True
1 B 22 7 12.0 False
2 C 19 7 9.0 False
3 D 14 9 8.0 True
4 E 14 12 8.4 True
5 F 11 9 7.5 True
```

### Example 1: Select Columns Equal to Specific Data Type

We can use the following code to select all columns in the DataFrame that have a data type equal to either int or float:

```
#select all columns that have an int or float data type  
df.select_dtypes(include=)
```

```
points assists minutes
```

```
0 18 5 10.1
```

```
1 22 7 12.0
```

```
2 19 7 9.0
```

```
3 14 9 8.0
```

```
4 14 12 8.4
```

```
5 11 9 7.5
```

Notice that only the columns with a data type equal to int or float are selected.

Example 2: Select Columns Not Equal to Specific Data Type

We can use the following code to select all columns in the DataFrame that do not have a data type equal to either bool or object:

```
#select all columns that don't have a bool or object data type  
df.select_dtypes(exclude=)
```

```
points assists minutes
```

```
0 18 5 10.1
```

```
1 22 7 12.0
```

**2 19 7 9.0**

**3 14 9 8.0**

**4 14 12 8.4**

**5 11 9 7.5**

**Notice that only the columns that don't have a data type equal to bool or object are selected.**

**Also note that you can use the following syntax to display the data type of each column in the DataFrame:**

```
#display data type of all columnsdf.dtypes
```

**team object**

**points int64**

**assists int64**

**minutes float64**

**all\_star bool**

**dtype: object**

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