Pandas is an easier way to work with datasets in python.
It has lots of useful functions in its documentation.
Release date - 11 Jan, 2008.

Dataframe

A Pandas object that is used to store the dataset.
Information is organized in rows and columns.
Dataframes simplify common operations, like sorting or wrangling data.
Can be created from a dictionary of lists. Keys become column headers.

Series

A Pandas object used to create dataframes.
Seen as a one-dimensional list of data.

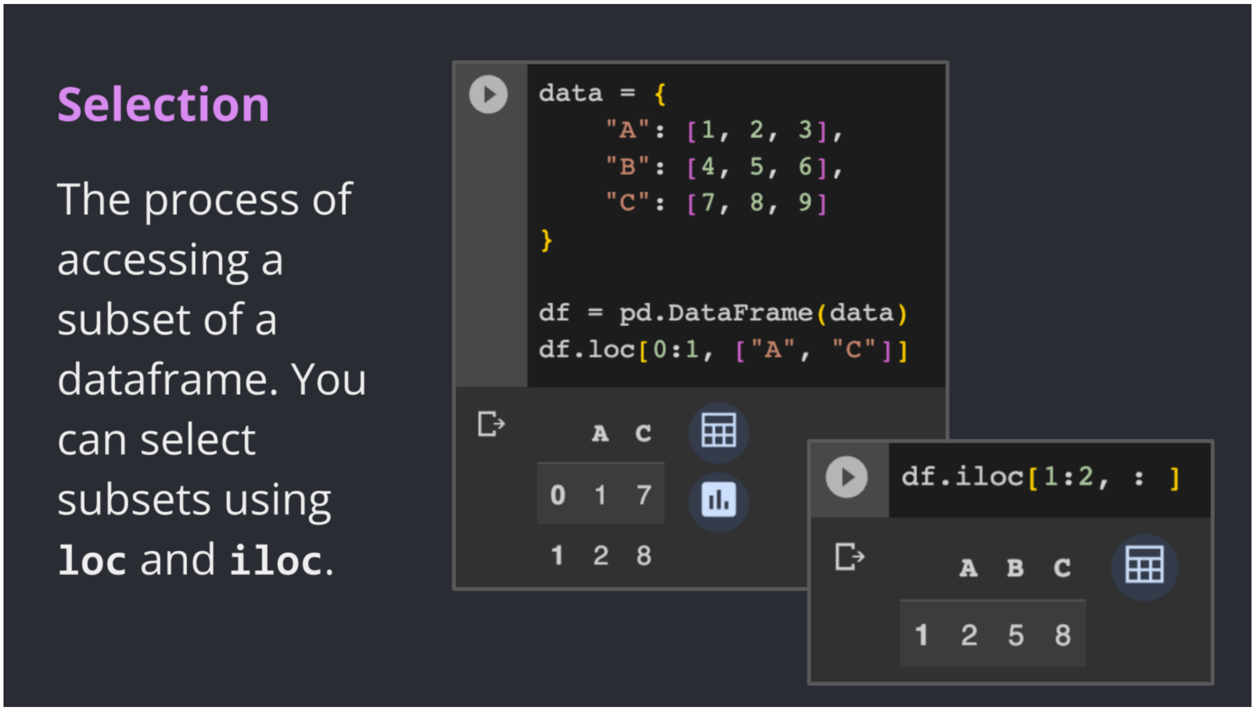
> Think of it as a single column in a dataframe.

Indexing into dataframes
Techniques:

df.loc[]	df.loc[row_label, col_label]
df.iloc[]	df.iloc[row_index, col_index]

Selection

The process of accessing a subset of a dataframe. You can select subsets using **loc** and **iloc**.



Selection

The process of accessing a subset of a dataframe. You can select subsets using **loc** and **iloc**.

```python
data = {
    "A": [1, 2, 3],
    "B": [4, 5, 6],
    "C": [7, 8, 9]
}

df = pd.DataFrame(data)
df.loc[0:1, ["A", "C"]]
```

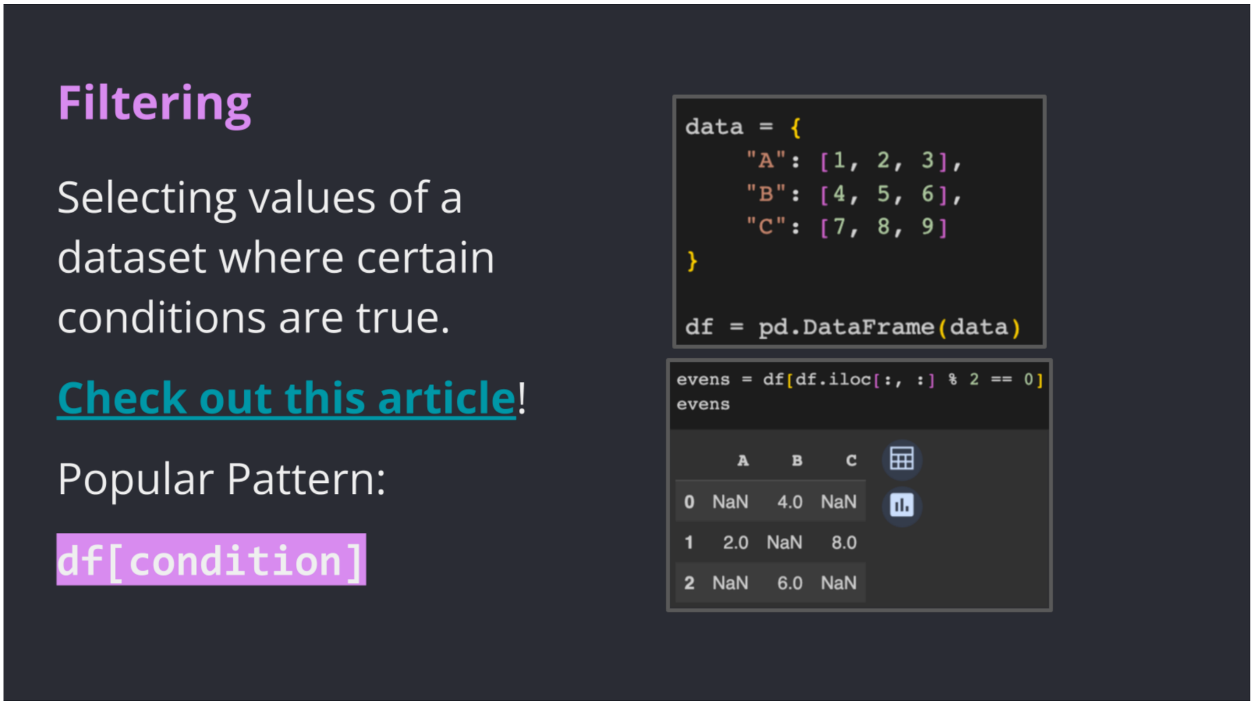
```
   A  C
0  1  7
1  2  8
```

```python
df.iloc[1:2, : ]
```

```
   A  B  C
1  2  5  8
```

Filtering

Selecting values of a dataset where certain conditions are true.
df[condition]



Filtering

Selecting values of a dataset where certain conditions are true.

Check out this article!

Popular Pattern:

df[condition]

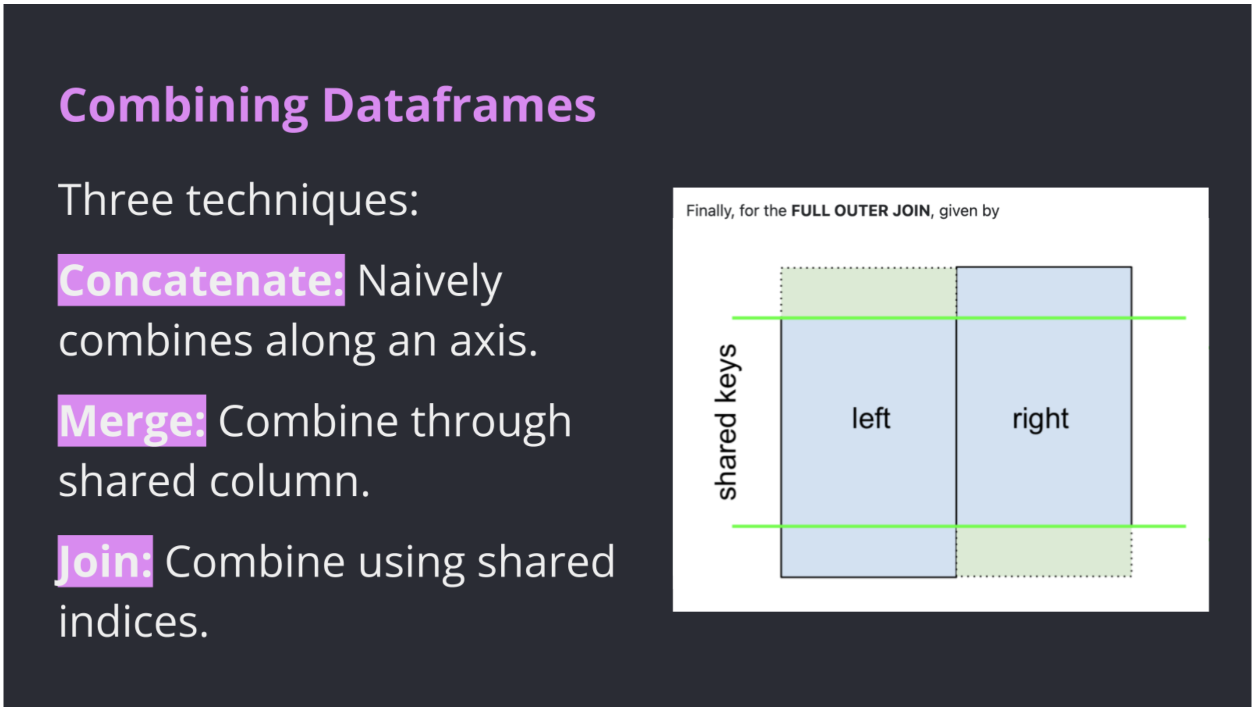
```
data = {
    "A": [1, 2, 3],
    "B": [4, 5, 6],
    "C": [7, 8, 9]
}

df = pd.DataFrame(data)
```

```
evens = df[df.iloc[:, :] % 2 == 0]
evens
```

	A	B	C
0	NaN	4.0	NaN
1	2.0	NaN	8.0
2	NaN	6.0	NaN

Combining Data Frames

Concatenate: Naively combines along an axis
Merge: Combine through shared column
Join: combine using shared indices (Inner, Left, Right, Outer)



Combining Dataframes

Three techniques:

Concatenate: Naively combines along an axis.

Merge: Combine through shared column.

Join: Combine using shared indices.

Finally, for the **FULL OUTER JOIN**, given by