# Aggregating Data

This lesson introduces us to aggregation of data and focuses on how we can use entire columns to aggregate using Pandas.

WE'LL COVER THE FOLLOWING

- Aggregation

## Aggregation

**Aggregation** is the procedure of converting a large number of values, or a dataset, into a single value or quantity aimed to summarize or describe the data. During data analysis, we always want to summarize data in one way or another. When we take a *sum*, *count* the number of items, or take the *average* of some values we aggregate data. Common aggregation methods are:

- *sum*
- *count*
- *maximum*
- *minimum*
- *average*

Aggregation is an essential step in analyzing data as it tells us the nature of the data in a single quantity. Let's look at examples on our California Census Housing Dataset to see how this is true.

housing.csv

```
import pandas as pd
df = pd.read_csv('housing.csv')
print('total household blocks: ', len(df))

# Sum function
summmation = df['households'].sum()
print('total households : ', summation)
print('total households : ', summation)

# Minimum Function
minimum = df['median_income'].min()
print('minimum median income (in tens of thousabds of $) of a household:', minimum)

# maximum Function
maximum = df['median_house_value'].max()
print('maximum median house value of a household block:',maximum)

# Average Function
avg = df['population'].mean()
print('average population of each block: ', avg)

# Using aggregated value to filter
condition = df['population'] > avg
new_df = df[condition]
print('number of household blocks that have population greater than average population: ', le
```

In **line 6**, we select the `households` column and find its sum. This means we have *aggregated* the whole column with just one value which tells us the total number of households in California. In the same way, we have found the mean, minimum, and maximum for different columns in the following lines.

In **line 23** we filter the data using the quantity `avg` we calculated above in **line 18**. This filtration tells us the number of household blocks with higher than the average population. Then we find the number of blocks that are extracted after this filter using the `len` function in **line 24**.

This gives us great insight that out of 20640, only 7512 household blocks have a population higher than the average population of a household block. This was a use case of how we can use aggregations from data to filter the data.

Now that we know how to aggregate data and can calculate statistics for an individual column, we may want to go a little deeper and gather information about specific types of items in a column. We will see that in the next lesson.