



# Dataframes

---

Data Engineering Workshops pt 3  
Ionian IEEE Student Branch

# Τι είναι ένα Dataframe

- Δομή δεδομένων:
  - Διδιάστατοι πίνακες
  - Μεταβλητό μέγεθος
  - Συλλογή δεδομένων
- Αποτελείται από:
  - Στήλες (Columns)
  - Γραμμές (Rows)
  - Δεδομένα (Data)

The diagram illustrates the structure of a Dataframe. It features a table with 7 rows and 6 columns. The columns are labeled 'Name', 'Team', 'Number', 'Position', and 'Age'. The rows are indexed from 0 to 6. Annotations include: 'Columns' with arrows pointing to the column headers; 'Rows' with arrows pointing to the row indices; and 'Data' with a box highlighting the data cells for rows 2, 3, and 4. The table data is as follows:

	<i>Name</i>	<i>Team</i>	<i>Number</i>	<i>Position</i>	<i>Age</i>
0	Avery Bradley	Boston Celtics	0.0	PG	25.0
1	John Holland	Boston Celtics	30.0	SG	27.0
2	Jonas Jerebko	Boston Celtics	8.0	PF	29.0
3	Jordan Mickey	Boston Celtics	NaN	PF	21.0
4	Terry Rozier	Boston Celtics	12.0	PG	22.0
5	Jared Sullinger	Boston Celtics	7.0	C	NaN
6	Evan Turner	Boston Celtics	11.0	SG	27.0

OG

# Δημιουργία Dataframe

- Δημιουργία από:
  - Βάσεις Δεδομένων
  - Αρχεία csv
  - Dictionaries
  - Αρχεία JSON
  - Lists

```
df = pd.read_json('sample.json')  
df
```

	name	age	car	gender
0	John	30	None	male
1	Marie	26	Volvo	female

```
df = pd.read_csv("sample.csv")  
df
```

	name	age	car	gender
0	John	30	NaN	male
1	Marie	26	Volvo	female

```
import pandas as pd
```

```
df = pd.DataFrame([[18, "male"],  
                  [35, "female"],  
                  [56, "male"],  
                  [24, "female"]],  
                  columns=['age', 'gender'])
```

```
df
```

	age	gender
0	18	male
1	35	female
2	56	male
3	24	female

# Επισκόπηση Dataframe

---

- Εμφάνιση πρώτων γραμμών:
  - `df.head()`
- Εμφάνιση διαστάσεων dataframe:
  - `df.shape`

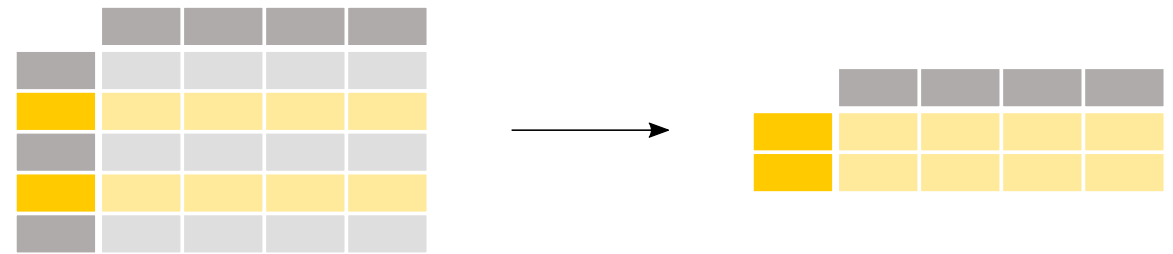
# “SELECT” από Dataframe

---

- Select 1 γραμμή:
  - `print(df.loc[0])`
- Select πολλές γραμμές:
  - `print(df.loc[[0, 1]])`
- Select στήλη:
  - `print(df["col_name"])`
  - `print(df.col_name)`
- Select πολλές γραμμές:
  - `print(df[["col_1", "col_2"]])`

# Filter σε γραμμές

- Επιλογή βάσει συνθήκης:
  - `df[df["age"] > 35]`
- Επιλογή με διάστημα τιμών:
  - `df[df["age"].isin([18, 60])]`
- Επιλογή βάσει πολλαπλών συνθηκών:
  - `df[(df["age"] > 35) | (df["gender"] == "male")]`



# Data Cleaning

---

- Διαγραφή σειρών με NULL τιμές
  - `df.dropna()`
- Αντικατάσταση των NULL τιμών
  - `df.fillna(value)`

# Summary Stats

---

- Άθροισμα τιμών στήλης
  - `df["age"].sum()`
- Μέγιστο\Ελάχιστο στήλης
  - `df["age"].max() \ df["age"].min()`
- Μέσος όρος τιμών στήλης
  - `df["age"].mean()`
- Διάμεσος τιμών
  - `df["age"].median()`



# Εφαρμογή Συνάρτησης στο Dataframe

- Εφαρμογή της συνάρτησης “func” σε όλες τις τιμές του Dataframe:
  - `df.apply(func)`
- Εφαρμογή της “func” σε όλες τις τιμές μιας στήλης:
  - `df[“age”].apply(func)`

```
import pandas as pd

def func(x):
    return x*2

df = pd.DataFrame([[4, 9]] * 3, columns=['A', 'B'])
display(df)
df.apply(func)
```

	A	B
0	4	9
1	4	9
2	4	9

	A	B
0	8	18
1	8	18
2	8	18

# Κανονικοποίηση Δεδομένων

---

- Αντικατάσταση των τιμών ώστε να βρίσκονται στο διάστημα  $[0,1]$
- Απαραίτητο για machine learning
  - `df.apply(lambda x: x/x.max(), axis=0)`

# Ομαδοποίηση Δεδομένων

- Δημιουργία νέου Dataframe με ομάδες βασισμένες σε τιμές του αρχικού Dataframe
- Χρησιμοποιείται μαζί με κάποιο Summary Stat
  - `df.groupby("age").count()`
  - `df.groupby("gender").median()`
- Η τελευταία συνάρτηση εφαρμόζεται σε όλες τις τιμές που δεν χρησιμοποιούνται για ομαδοποίηση

```
import pandas as pd
```

```
df = pd.DataFrame([[18, "male"],  
                  [35, "female"],  
                  [56, "male"],  
                  [24, "female"]], columns=['age', 'gender'])
```

```
display(df)
```

```
df_group = df.groupby("gender").mean()
```

```
display(df_group)
```

	age	gender
0	18	male
1	35	female
2	56	male
3	24	female

	age
gender	
female	29.5
male	37.0