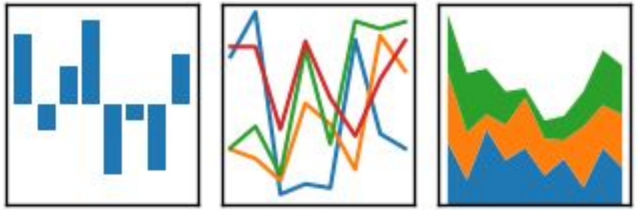
Sessions *a GenTech Initiative*

Coding x Analytics

Pandas

- Series
- Data Frames
- Mutations
  - concat
  - merge
  - join
  - dropna
  - fillna
- Utilities
  - unique
  - columns
  - index
  - describe
  - shape
  - isnull
- Data Loading

Pandas is a library that helps you to visualise and manipulate data for analysis.



Structure

1. Variables
2. Data Types
3. Conditional Statements
4. For Loops
5. While Loops
6. Functions
7. Using Libraries
8. **Pandas**

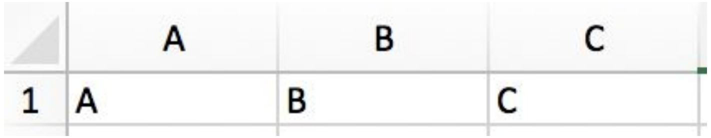
Learning Outcome

- Familiar with Python
- Able to manipulate common data types
- How to install other libraries
- Using Pandas

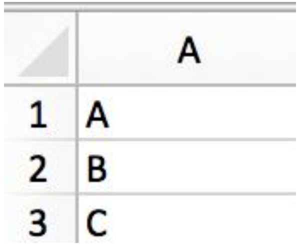
Pandas

- **Series**
- Data Frames
- Mutations
  - concat
  - merge
  - join
  - dropna
  - fillna
- Utilities
  - unique
  - columns
  - index
  - describe
  - shape
  - isnull
- Data Loading

Series is a set of data of 1 column and multiple rows (indexes), **or** multiple columns and 1 row (index).



	A	B	C
1	A	B	C



	A
1	A
2	B
3	C

Structure

1. Variables
2. Data Types
3. Conditional Statements
4. For Loops
5. While Loops
6. Functions
7. Using Libraries
8. **Pandas**

Learning Outcome

- Familiar with Python
- Able to manipulate common data types
- How to install other libraries
- Using Pandas

Pandas

- **Series**
- Data Frames
- Mutations
  - concat
  - merge
  - join
  - dropna
  - fillna
- Utilities
  - unique
  - columns
  - index
  - describe
  - shape
  - isnull
- Data Loading

Syntax

```
import pandas as pd

columns = [“A”, “B”, “C”]
data = [10, 20, 30]

pd.Series(data = data, index = columns)
```

Structure

1. Variables
2. Data Types
3. Conditional Statements
4. For Loops
5. While Loops
6. Functions
7. Using Libraries
8. **Pandas**

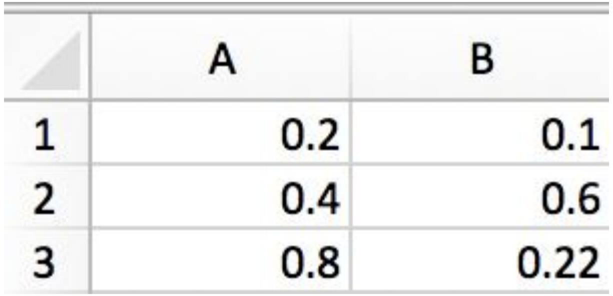
Learning Outcome

- Familiar with Python
- Able to manipulate common data types
- How to install other libraries
- Using Pandas

Pandas

- Series
- **Data Frames**
- Mutations
  - concat
  - merge
  - join
  - dropna
  - fillna
- Utilities
  - unique
  - columns
  - index
  - describe
  - shape
  - isnull
- Data Loading

Data Frames is a table of data containing multiple columns **and** multiple rows (indexes).



	A	B
1	0.2	0.1
2	0.4	0.6
3	0.8	0.22

Structure

1. Variables
2. Data Types
3. Conditional Statements
4. For Loops
5. While Loops
6. Functions
7. Using Libraries
8. **Pandas**

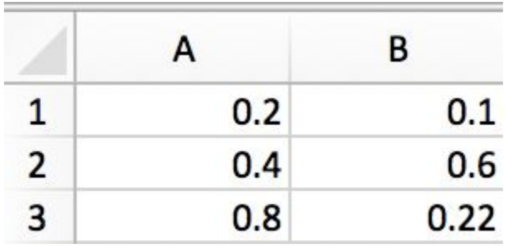
Learning Outcome

- Familiar with Python
- Able to manipulate common data types
- How to install other libraries
- Using Pandas

Pandas

- Series
- **Data Frames**
- Mutations
  - concat
  - merge
  - join
  - dropna
  - fillna
- Utilities
  - unique
  - columns
  - index
  - describe
  - shape
  - isnull
- Data Loading

Syntax



	A	B
1	0.2	0.1
2	0.4	0.6
3	0.8	0.22

```
import pandas as pd

columns = [“A”, “B”]
indexes = [“1”, “2”, “3”]
data = [[0.2, 0.1], [0.4, 0.6], [0.8, 0.22]]

pd.DataFrame(data = data, index = indexes, columns = columns)
```

Structure

1. Variables
2. Data Types
3. Conditional Statements
4. For Loops
5. While Loops
6. Functions
7. Using Libraries
8. **Pandas**

Learning Outcome

- Familiar with Python
- Able to manipulate common data types
- How to install other libraries
- Using Pandas

Pandas

- Series
- Data Frames
- **Mutations**
  - **concat**
  - merge
  - join
  - dropna
  - fillna
- Utilities
  - unique
  - columns
  - index
  - describe
  - shape
  - isnull
- Data Loading

Description

Adds the list of DataFrames into one DataFrame.

Rules

- Same columns, or
- Same rows

Syntax

```
pd.concat([df1, df2, df3], axis = 0)
```

Structure

1. Variables
2. Data Types
3. Conditional Statements
4. For Loops
5. While Loops
6. Functions
7. Using Libraries
8. **Pandas**

Learning Outcome

- Familiar with Python
- Able to manipulate common data types
- How to install other libraries
- Using Pandas

Pandas

- Series
- Data Frames
- **Mutations**
  - concat
  - **merge**
  - join
  - dropna
  - fillna
- Utilities
  - unique
  - columns
  - index
  - describe
  - shape
  - isnull
- Data Loading

Description

Merges two DataFrame into one.

Rules

- Both DataFrames should have one similar column.

Syntax

```
pd.merge(df1, df2, how = “inner”, on = “key”)
```

Structure

1. Variables
2. Data Types
3. Conditional Statements
4. For Loops
5. While Loops
6. Functions
7. Using Libraries
8. **Pandas**

Learning Outcome

- Familiar with Python
- Able to manipulate common data types
- How to install other libraries
- Using Pandas

Pandas

- Series
- Data Frames
- **Mutations**
    - concat
    - merge
    - **join**
    - dropna
    - fillna
- Utilities
    - unique
    - columns
    - index
    - describe
    - shape
    - isnull
- Data Loading

Description

Joins two DataFrame into one.

Syntax

```
df1.join(df2, how = “inner”)
```

Structure

1. Variables
2. Data Types
3. Conditional Statements
4. For Loops
5. While Loops
6. Functions
7. Using Libraries
8. **Pandas**

Learning Outcome

- Familiar with Python
- Able to manipulate common data types
- How to install other libraries
- Using Pandas

Pandas

- Series
- Data Frames
- **Mutations**
  - concat
  - merge
  - join
  - **dropna**
  - fillna
- Utilities
  - unique
  - columns
  - index
  - describe
  - shape
  - isnull
- Data Loading

Description

Hides all rows/columns that contains ‘**Null**’ or ‘**NaN**’ values.

Syntax

```
df.dropna(axis = 0, thresh = None)
```

Structure

1. Variables
2. Data Types
3. Conditional Statements
4. For Loops
5. While Loops
6. Functions
7. Using Libraries
8. **Pandas**

Learning Outcome

- Familiar with Python
- Able to manipulate common data types
- How to install other libraries
- Using Pandas

Pandas

- Series
- Data Frames
- **Mutations**
  - concat
  - merge
  - join
  - dropna
  - **fillna**
- Utilities
  - unique
  - columns
  - index
  - describe
  - shape
  - isnull
- Data Loading

Description

Sets a value to all rows/columns that contains **'Null'** or **'NaN'** values.

Syntax

```
df.fillna(value = value)
```

Structure

1. Variables
2. Data Types
3. Conditional Statements
4. For Loops
5. While Loops
6. Functions
7. Using Libraries
8. **Pandas**

Learning Outcome

- Familiar with Python
- Able to manipulate common data types
- How to install other libraries
- Using Pandas

Pandas

- Series
- Data Frames
- Mutations
  - concat
  - merge
  - join
  - dropna
  - fillna
- **Utilities**
  - **unique**
  - columns
  - index
  - describe
  - shape
  - isnull
- Data Loading

Description

Generates a **list** of unique values based on column or index.

Syntax

```
df[“column”].unique()
```

Structure

1. Variables
2. Data Types
3. Conditional Statements
4. For Loops
5. While Loops
6. Functions
7. Using Libraries
8. **Pandas**

Learning Outcome

- Familiar with Python
- Able to manipulate common data types
- How to install other libraries
- Using Pandas

Pandas

- Series
- Data Frames
- Mutations
  - concat
  - merge
  - join
  - dropna
  - fillna
- **Utilities**
  - unique
  - **columns**
  - index
  - describe
  - shape
  - isnull
- Data Loading

Description

Returns a **list** of columns in the DataFrame.

Syntax

```
df.columns
```

Structure

1. Variables
2. Data Types
3. Conditional Statements
4. For Loops
5. While Loops
6. Functions
7. Using Libraries
8. **Pandas**

Learning Outcome

- Familiar with Python
- Able to manipulate common data types
- How to install other libraries
- Using Pandas

Pandas

- Series
- Data Frames
- Mutations
  - concat
  - merge
  - join
  - dropna
  - fillna
- **Utilities**
  - unique
  - columns
  - **index**
  - describe
  - shape
  - isnull
- Data Loading

Description

Returns the number of rows (indexes) in the DataFrame.

Syntax

```
df.index
```

Structure

1. Variables
2. Data Types
3. Conditional Statements
4. For Loops
5. While Loops
6. Functions
7. Using Libraries
8. **Pandas**

Learning Outcome

- Familiar with Python
- Able to manipulate common data types
- How to install other libraries
- Using Pandas

Pandas

- Series
- Data Frames
- Mutations
  - concat
  - merge
  - join
  - dropna
  - fillna
- **Utilities**
  - unique
  - columns
  - index
  - **describe**
  - shape
  - isnull
- Data Loading

Description

Returns a set of mathematical (count, mean, std, min, 25%, 50%, 75%, max) values in the DataFrame.

Syntax

```
series.describe()
df.describe()
```

Structure

1. Variables
2. Data Types
3. Conditional Statements
4. For Loops
5. While Loops
6. Functions
7. Using Libraries
8. **Pandas**

Learning Outcome

- Familiar with Python
- Able to manipulate common data types
- How to install other libraries
- Using Pandas

Pandas

- Series
- Data Frames
- Mutations
  - concat
  - merge
  - join
  - dropna
  - fillna
- **Utilities**
  - unique
  - columns
  - index
  - describe
  - **shape**
  - isnull
- Data Loading

Description

Returns the size (indexes, columns) of the DataFrame.

Syntax

```
df.shape()
```

Structure

1. Variables
2. Data Types
3. Conditional Statements
4. For Loops
5. While Loops
6. Functions
7. Using Libraries
8. **Pandas**

Learning Outcome

- Familiar with Python
- Able to manipulate common data types
- How to install other libraries
- Using Pandas

Pandas

- Series
- Data Frames
- Mutations
  - concat
  - merge
  - join
  - dropna
  - fillna
- **Utilities**
  - unique
  - columns
  - index
  - describe
  - shape
  - **isnull**
- Data Loading

Description

Returns a same shape DataFrame but the values are ‘True’ if there are no values in that specific column and index.

Syntax

```
df.isnull()
```

Structure

1. Variables
2. Data Types
3. Conditional Statements
4. For Loops
5. While Loops
6. Functions
7. Using Libraries
8. **Pandas**

Learning Outcome

- Familiar with Python
- Able to manipulate common data types
- How to install other libraries
- Using Pandas

Pandas

- Series
- Data Frames
- Mutations
  - concat
  - merge
  - join
  - dropna
  - fillna
- Utilities
  - unique
  - columns
  - index
  - describe
  - shape
  - isnull
- **Data Loading**

Description

Loads a CSV file into a DataFrame

Syntax

```
pd.read_csv(“path to csv”)
```

Tips

There are many other types that pandas can read (sql, excel, html, json, etc).

Structure

1. Variables
2. Data Types
3. Conditional Statements
4. For Loops
5. While Loops
6. Functions
7. Using Libraries
8. **Pandas**

Learning Outcome

- Familiar with Python
- Able to manipulate common data types
- How to install other libraries
- Using Pandas