Introduction

- Pandas is a Python library used for working with **data sets.**
- It has functions for **analyzing, cleaning, exploring, and manipulating data**.
- Pandas are also able to delete rows that are not relevant, or contains wrong values, like empty or NULL values. This is called cleaning the data.
- The name "Pandas" has a reference to both **"Panel Data", and "Python Data Analysis"** and was created by Wes McKinney in 2008.

Installation of Pandas

pip install pandas

```
!pip install pandas
```

```
Requirement already satisfied: pandas in /usr/local/lib/python3.7/dist-packages (1.3.5)
Requirement already satisfied: python-dateutil>=2.7.3 in /usr/local/lib/python3.7/dist-p
Requirement already satisfied: numpy>=1.17.3 in /usr/local/lib/python3.7/dist-packages (
Requirement already satisfied: pytz>=2017.3 in /usr/local/lib/python3.7/dist-packages (f
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/dist-packages (from
```

Import pandas and know its version

```
import pandas as pd
print(pd.__version__)
```

```
1.3.5
```

DataFrame

DataFrame is a 2 dimensional data structure, like a table with rows and columns.

```
df1 = pd.DataFrame([[1,2,3],[7,4,9],[3,9,2],[1,8,4],[2,6,5],[5,8,3]],columns=["A","B","Num"])
print(df1)
print(df1.shape)
```

```
   A  B  Num
0  1  2    3
1  7  4    9
2  3  9    2
3  1  8    4
4  2  6    5
5  5  8    3
(6, 3)
```

```
df1.head()
```

	A	B	Num
0	1	2	3
1	7	4	9
2	3	9	2
3	1	8	4
4	2	6	5

```
print(df1.head(3))
```

```
   A  B  Num
0  1  2    3
1  7  4    9
2  3  9    2
```

```
print(df1.tail())
print("Last Three records")
print(df1.tail(3))
```

```
   A  B  Num
1  7  4    9
2  3  9    2
3  1  8    4
4  2  6    5
5  5  8    3
Last Three records
   A  B  Num
3  1  8    4
4  2  6    5
5  5  8    3
```

```
print(df1["Num"])
```

```
0    3
1    9
2    2
3    4
4    5
5    3
Name: Num, dtype: int64
```

```
df1
```

	A	B	Num
0	1	2	3
1	7	4	9
2	3	9	2
3	1	8	4
4	2	6	5

```
print (df1.iloc[1,2])
print(df1.iloc[2,:])
print(df1.iloc[:,0])
print (df1.iloc[1:4,1:3])
```

```
9
A      3
B      9
Num    2
Name: 2, dtype: int64
0    1
1    7
2    3
3    1
4    2
5    5
Name: A, dtype: int64
   B  Num
1  4    9
2  9    2
3  8    4
```

```
print(df1.iloc[:,2])
```

```
0    3
1    9
2    2
3    4
4    5
5    3
Name: Num, dtype: int64
```

Mount Google Drive

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mour
```

Read CSV file

```
mydata=pd.read_csv("/content/drive/MyDrive/AIT322-ML/income-expense.csv")
print(mydata.shape)
print(mydata.head(8))
```

```
(14, 3)
   Age   Income  Expense
0   25  40000.0    20000
1   26  35000.0    18000
2   27  90000.0    60000
3   32  70000.0    28000
4   31  75000.0    32000
5   30  71000.0    30000
6   47      NaN    25000
7  125  76000.0    30000
```

```
print(mydata.isnull().sum())
```

```
Age        0
Income     1
Expense    0
dtype: int64
```

```
print(mydata.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 14 entries, 0 to 13
Data columns (total 3 columns):
 #   Column   Non-Null Count  Dtype
---  ------   --------------  -----
 0   Age      14 non-null     int64
 1   Income   13 non-null     float64
 2   Expense  14 non-null     int64
dtypes: float64(1), int64(2)
memory usage: 464.0 bytes
None
```

```
print(mydata.mean())
print(mydata.median())
print(mydata.quantile(0.75))
```

```
Age           48.500000
Income     65000.000000
Expense    25285.714286
dtype: float64
Age           48.0
Income     70000.0
Expense    24500.0
dtype: float64
Age           55.75
```

```
Income     76000.00
Expense    29500.00
Name: 0.75, dtype: float64
```

```
mydata["Income"].fillna(mydata["Income"].median(),inplace=True)
mydata.isnull().sum()
```

```
Age        0
Income     0
Expense    0
dtype: int64
```

```
print(mydata)
```

```
    Age   Income  Expense
0    25  40000.0    20000
1    26  35000.0    18000
2    27  90000.0    60000
3    32  70000.0    28000
4    31  75000.0    32000
5    30  71000.0    30000
6    47  70000.0    25000
7   125  76000.0    30000
8    49  56000.0    16000
9    55  67000.0    20000
10   54  80000.0    24000
11   56  86000.0    25000
12   60  56000.0    15000
13   62  43000.0    11000
```

```
print(mydata["Age"].mean())
print(mydata["Income"].median())
print(mydata["Expense"].quantile(0.75))
```

```
48.5
70000.0
29500.0
```

```
mydata.describe()
```

	Age	Income	Expense
count	14.000000	14.000000	14.000000
mean	48.500000	65357.142857	25285.714286

```
mydata.corr()
```

	Age	Income	Expense
Age	1.000000	0.180893	-0.177566
Income	0.180893	1.000000	0.701646
Expense	-0.177566	0.701646	1.000000