

# Python Pour L'Analyse de Données




Dr. Amine Boukhaled - ESIEE IT



# Python



# Jupyter



# Cycle de vie d'une analyse de données

# Le cycle de vie d'une analyse de données

1

Importation des données, vérification du typage des colonnes.

2

Étude de la qualité des données, duplicatas, imputation, suppression d'indicateurs.

3

Analyse statistique, création de nouveaux indicateurs, filtrage, jointure de fichiers, visualisations.

4

Interprétation des résultats, conclusion, perspectives d'améliorations, exportation des données.

# Étape 1

- `df = pd.read_csv()` # importation
- `df.shape` # taille
- `df.info()` # taille, type, mémoire
- `df.dtypes` # type
- `df.columns` # nom colonnes
- `pd.to_numeric()` # cast
- `pd.to_datetime()` # cast date
- `df['col'].astype('int')` # cast int

## Étape 2

- `df.isna().mean()` # taux valeurs manquantes
- `df.count()` # taux remplissage
- `df.dropna()` # supprimer na
- `df.duplicates()` # détecter duplicatas
- `df.drop_duplicates()` # supprimer duplicatas
- `df.fillna()` # remplacer nan
- `df.drop()` # supprimer lignes ou colonnes

## Étape 3

- `df.loc[]` # selection ligne / colonne
- `df.iloc[]` # selection ligne / colonne
- `df['col1']` # selection colonne
- `df[['col1', 'col2']]` # selection colonnes
- `df.plot(kind='bar')` # graphiques
- `df.describe()` # statistiques
- `df.mean()` # moyenne
- `df.apply(np.max)` # appliquer fonction
- `df.groupby('col1')['col2'].max()` # grouper et agréger
- `df[df['col1'] == 10]` # filtrage
- `df['newcol'] = 0` # création colonne
- `df['sumcol'] = df['col1'] + df['col2']`





# Labs

