Handling Missing Data

Missing data is very common in many data analysis applications. pandas has a great ability to deal with the missing data.
Let's learn some convenient methods to deal with missing data in pandas:

Method/Function	Description	Example
isnull() / isna()	Checks for missing values, returns True for NaN values and False for non-NaN values.	df.isnull()
notnull() / notna()	Opposite of isnull(), returns True for non-NaN values and False for NaN values.	df.notnull()
sum()	Counts the number of missing or non-missing values in a DataFrame or Series.	df.isnull().sum()
info()	Provides a summary of the DataFrame, including the count of non-null values.	df.info()
dropna()	Drops rows or columns with missing values.	df.dropna(axis=0) (drop rows) df.dropna(axis=1) (drop columns)
fillna()	Fills missing values with a specific constant or calculated value (mean, median, etc.).	df.fillna(value=0) df.fillna(method='ffill')
SimpleImputer (Sklearn)	A Scikit-learn class for advanced strategies of imputing missing data (mean, median, most frequent, constant).	imputer = SimpleImputer(strategy='mean') df['A'] = imputer.fit_transform(df[['A']])
thresh parameter in dropna()	Allows you to set a threshold of non-null values required to retain a row or column.	df.dropna(axis=0, thresh=3) (retain rows with at least 3 non-null values)

Method/Function	Description	Example
method='ffill' in fillna()	Forward fills missing values, filling each missing value with the previous valid value.	df.fillna(method='ffill')
method='bfill' in fillna()	Backward fills missing values, filling each missing value with the next valid value.	df.fillna(method='bfill')
Filling with Mean	Fills missing values with the mean of the column.	df['A'].fillna(value=df['A'].mean())
Filling with Median	Fills missing values with the median of the column.	df['A'].fillna(value=df['A'].median())