Amr Abdelkarem

PANDAS CHEAT SHEET

ESSENTIAL COMMANDS FOR DATA ANALYSIS

Swipe to know

Amr Abdelkarem

DATA IMPORT

- pd.read_csv('file.csv') – CSV
- pd.read_excel('file.xlsx') – Excel
- pd.read_sql(query, conn) – SQL
- pd.read_json('file.json') – JSON
- pd.read_parquet('file.parquet') – Parquet

💡 Tip: Use pd.read_clipboard() for quick tests

Amr Abdelkarem

DATA SELECTION

- ['col'] – Select column
- df.loc[row, 'col'] – Label-based
- df.iloc[0:5, 0:2] – Integer-based
- df.query('col > 5') – SQL-style filter
- df[df['col'].isin(['A','B'])] – Multiple values

Amr Abdelkarem

DATA MANIPULATION

- df.groupby('col').agg({'col2':['mean','sum']})
- df.merge(df2, on='key')
- df.pivot_table(values='val', index='idx')
- df.sort_values(['col1','col2'])
- df.melt(id_vars=['id'], value_vars=['A','B'])
- df.apply(lambda x: x*2)

💡 Also try .assign() to create new columns in a chain

Amr Abdelkarem

DATA CLEANING

- df.dropna(subset=['col'])
- df.fillna(method='ffill')
- df.drop_duplicates(subset=['col'])
- df['col'].replace({'old':'new'})
- df['col'].astype('category')
- df.interpolate(method='linear')

💡 Check missing data → df.isnull().sum()

Amr Abdelkarem

STRING OPERATIONS

- df['col'].str.contains('pattern')
- df['col'].str.extract('(\d+)')
- df['col'].str.split(',', expand=True)
- df['col'].str.lower()
- df['col'].str.strip()
- df['col'].str.replace(r'\s+',' ')

Amr Abdelkarem

STATISTICS

- df.describe() – Summary stats
- df['col'].agg(['mean','median','std'])
- df['col'].value_counts(normalize=True)
- df.corr(method='pearson')
- df.cov()
- df.quantile([0.25,0.5,0.75])

💡 Add: .skew(), .kurt() for distribution insights

Amr Abdelkarem

TIME SERIES

- df.resample('M').mean() – Monthly avg
- df.rolling(window=7).mean() – Rolling mean
- df.shift(periods=1) – Shift values
- pd.date_range('2024', periods=12, freq='M') – Date range
- df['date'].dt.strftime('%Y-%m-%d') – Format dates

Amr Abdelkarem

ADVANCED FEATURES

- df.pipe(func) – Method chaining
- pd.eval('df1 + df2') – Expression eval
- df.memory_usage(deep=True) – Memory usage
- df.select_dtypes(include='number') – Select numeric
- df.nlargest(5, 'col') – Top N values
- df.explode('col') – Expand lists

💡 Try .sample(n=5) for random samples

Amr Abdelkarem

DATA EXPORT

- df.to_csv('output.csv')
- df.to_excel('output.xlsx')
- df.to_parquet('output.parquet')
- df.to_json('output.json')

Amr Abdelkarem

TIPS & BEST PRACTICES

- Use .copy() when creating DataFrames
- Chain operations with .pipe
- Use astype('category') for optimization
- Avoid inplace=True when possible