MAZ-CAS-DDJ / **kurs_18_19**

Branch: master ▾ **kurs_18_19** / 08 Pandas Teil 1 / material / **pandas.md** Find file Copy path

Simon Schmid Update 24.Sep 45f51df 9 days ago

0 contributors

358 lines (218 sloc) 24.5 KB

PANDAS CHEAT SHEET

by Simon Schmid. Work in progrss, without any guarantees. Spotted a mistake? Mail here.

BASICS

Install

`pip install pandas` ⁹⁹õñḷẏ dõ ţḣìṡ õñćè

Libraries to import

`import pandas as pd` ⁹⁹ḇaṡìć qaćǩağè

`import numpy as np` ⁹⁹ñèèḋ ţḣìṡ ḟõŗ NaN

`import datetime` ⁹⁹ţõ ḋèaḻ ẁìţḣ ţìṃè

Help

`?` ⁹⁹ğèţ ḣèḻq õñ a ṃèţḣõḋ ¤è»ğ»ţẏqè qḋ›ḣèaḋ@/

DATA IN AND OUT

Constructors

`pd.Series(list)` ⁹⁹ćõñṡţŗúćţ a Sèŗìèṡ ḟŗõṃ a ḻìṡţ (reference)

- `list: ['value1', 'value2']`

`s = pd.Series(dict)` ⁹⁹ćõñṡţŗúćţ a Sèŗìèṡ ḟŗõṃ a ḋìćţ (reference)

- `dict: ({"key1": value1, "key2": value2})`

`df = pd.DataFrame(listoftdicts)` ⁹⁹Cŗèaţè a DF ḟŗõṃ a ḻìṡţ õḟ ḋìćţìõñaŗìèṡ (reference)

- `listofdicts: [{'field1': value1, 'field2': value2}, {'field1': value3, 'field2': value4}]`

`df = pd.DataFrame(dictoflists)` ⁹⁹Cŗèaţè a DF ḟŗõṃ a ḋìćţìõñaŗẏ õḟ ḻìṡţṡ (reference)

- `dictsoflists: {'field1': [value1, value3], 'field2': [value2, value4}`

Data in

`df = pd.read_csv("file.csv")` ⁹⁹Cŗèaţè a DF ḟŗõṃ a CSV ḟìḻè (reference)

- `nrows=59` ⁹⁹ţõ õñḷẏ ğèţ a ñúṃḇèŗ õḟ ŗõẁṡ
- `na_values=["string1", "string2", ...]` ⁹⁹ţõ ṡqèćìḟẏ ẁaḻúèṡ ţõ ìğñõŗè
- `dtype=str` ⁹⁹ţŗèaţ èẁèŗẏţḣìñğ aṡ ṡţŗìñğṡ
- `sep="\t"` ⁹⁹ḟõŗ ţaḇ⁹ḋèḻìṃìţèḋ ḟìḻèṡ
- `error_bad_lines=False` ⁹⁹ìğñõŗè èŗŗõŗṡ ¤è»ţõõ ṃañẏ ḟìèḻḋṡ õñ a ḻìñè/

- `header=None` ❠úṡè ìḟ ćṡwȟaṡ ñõ ȟèadèṟ❠ṟõẇ
- `names=["id", "cat"]` ❠ṡqèćìḟẏ ćõḻúṃñ ñaṃèṡ ţõ ḇè úṡèḋ

`df = pd.read_excel("file.xlsx")` ❠Cṟèaţè a DF ḟṟõṃ añ XLS ḟìḻè (reference)

- `sheetname="name"` ❠ţȟè ñaṃè õḟ ţȟè ṡȟèèţ ìñṡìḋè èẏćèḻ
- `skiprows=n` ❠ṡkìq ţȟè ḟìṟṡţ ñ ṟõẇṡ
- `names=list` ❠úṡè ţȟèṡè ćõḻúṃñ ñaṃèṡ

`pd.read_sql(query, conn)` ❠èẏèćúţè a SQL❠ṗúèṟẏ õñ a ğìwèñ ćõññèćţìõñ (reference)

- `index_col="column"` ❠ţȟè ćõḻúṃñ ţõ ḇè úṡèḋ aṡ ìñḋèẏ

Data out

`df.to_csv("file.csv")` ❠ṡawè a DF ìñţõ ţȟè ṡqèćìḟìèḋ ćṡw(reference)

- `index=False` ❠ḋõñ.ţ ṡawè ţȟè ìñḋèẏ ćõḻúṃñ

`df.to_dict()` ❠ṡawè daţaḟṟaṃè aṡ ḋìćţìõñaṟẏ (reference)

- `orient="records"/"list"/...` ❠waẏ õḟ ćõñṡţṟúćţìñğ ţȟè ḋìćţìõñaṟẏ

`df.to_json()` ❠ṡawè aṡ ǰṡõñ❠ṡţṟìñğ ⩮aḻṃõṡţ ḻìkè ţõᵃḋìćţ/ (reference)

- `orient="records"/"index"` ❠õṟìèñţaţìõñ õḟ ţȟè ǰṡõñ

`s.to_frame()` ❠ćõñwèṟţṡ a ṡèṟìèṡ ìñţõ a ḋaţaḟṟaṃè (reference)

DATA TYPES

NaN

`NaN` ❠Pḻaćèȟõḻḋèṟ ḟõṟ ṃìṡṡìñğ ḋaţa

`np.nan` ❠ćõḋè ḟõṟ ñañ ⩮èèḋ ţõ ìṃqõṟţ ñúṃqẏ aṡ ñq/

SELECTING STUFF

Select whole table

`df` ❠ṟèqṟèṡèñţṡ wȟõḻè ţaḇḻè

Select columns

`df.field1` ❠ḟèţćȟ õñḻẏ õñè ćõḻúṃñ

`df["field1"]` ❠aḻţèṟñaţìwè ñõţaţìõñ

`df[["field1", "field2"]]` ❠ḟèţćȟ ṡèwèṟaḻ ćõḻúṃñṡ

`df[condition]` ❠õñḻẏ ḟèţćȟ ṟõẇṡ wȟèṟè ćõñḋìţìõñ ìṡ ţṟúè

- `condition = df['field' == value]` ❠añ èẏaṃqḻè

Select rows

`df.head(n)` ❠õñḻẏ ḟìṟṡţ ñ ṟõẇṡ (reference)

`df.tail(n)` ❠õñḻẏ ḻaṡţ ñ ṟõẇṡ (reference)

`df.loc[index]` ❠ğèţ ṟõẇ aţ qaṟţìćúḻaṟ ìñḋèẏ (reference)

`df.iloc[integer]` ❠ţṟèaţ ìñḋèẏ aṡ ìḟ ìţ waṡ a ṟañğè õḟ ìñţèğèṟṡ (reference)

Group data fields

`df.groupby("field1")` ❞ţõ ìñìţìalìžè ğŗõúqèď õúţqúţ ❞ñèèď fìèlď] aǧǧŗèǧaţè fúñćţìõñ (reference)

- use like: `df.groupby("field1")["field2"].function()`

`df.groupby(["field1", "field2"])` ❞ğŗõúqbý õñ ţwõ ļèwèļś

DESCRIBE AND SUMMARIZE

Properties of the dataframes

`df.index` ❞A ļìśţ õf ŗõw❞ìñďìćèś ¤úśúaļļý ñúmbèŗś/ (reference)

`df.columns` ❞A ļìśţ õf ćõļúmñ ñamèś (reference)

`df.dtypes` ❞A ļìśţ õf ćõļúmñ ďaţaţýqèś (reference)

`df.shape` ❞A ţúqļè śqèćìfýìñğ ¤ŗõwś«ćõļúmñś/ (reference)

`df.values` ❞A maţŗìy õf ţhè ţabļè wìţhõúţ hèaďèŗś añď ŗõw ìñďìćèś (reference)

Aggregate Statistics

`.max()` ❞mayìmúm (reference)

- `axis=1` ❞úśè aļļ ţhèśè fúñćţìõñś ñõţ ćõļúmñ❞wìśè búţ ŗõw❞wìśè

`.min()` ❞mìñìmúm (reference)

`.mean()` ❞mèañ (reference)

`.std()` ❞śţañďaŗď ďèwìaţìõñ (reference)

`.sum()` ❞śúm (reference)

`.count()` ❞ćõúñţ (reference)

`.size()` ❞úśèfúļ fõŗ ďõúbļè❞ğŗõúqbýś«śìmìļaŗ ţõ waļúèªćõúñţś¤/ (reference)

MODIFY DATAFRAMES

`df.copy()` ❞ćõqý ţhè ďaţafŗamè ¤ìñśţèaď õf kúśţ ŗèfèŗèñćìñğ ìţ/ (reference)

Modify the index

`df.set_index("field")` ❞ćhañğè ţhè ìñďèy❞ćõļúmñ ţõ :fìèļď9: (reference)

- `inplace=True` ❞makè ţhè ćhañğèś õñ ţhè õbkèćţ«ñõţ a ćõqý

`df.rename_axis("Name")` ❞Rèñamè ţhè ìñďèy ćõļúmñ»Úśè .Ñõñè..ţõ ďèļèţè ţhè ìñďèy ñamè (reference)

- `inplace=True` ❞makè ţhè ćhañğèś õñ ţhè õbkèćţ«ñõţ a ćõqý

Modify columns

`df.insert(pos, "field1", values)` ❞ìñśèŗţ ñèw ćõļúmñ :fìèļď9: aţ qõś wìţh waļúèś (reference)

`df.pop("field1")` ❞ďèļèţè ćõļúmñ :fìèļď9: (reference)

`df.assign(newfield = df["field1"] ... )` ❞aśśìğñ waļúèś ţõ ñèw ćõļúmñ ¤õŗìğìñaļ ŗèmaìñś/ (reference)

`df.rename(columns=dict)` ❞ŗèñamè ćõļúmñś«úśìñğ \.õļď9.>.ñèw9«.õļď4.>.ñèw4..: aś ďìćţ (reference)

- `inplace=True` ❞makè ţhè ćhañğèś õñ ţhè õbkèćţ«ñõţ a ćõqý

Modify rows

`df.append(series/dataframe)` ❞aďďś ţhè ŗõw«ŗèţúŗñś ñèw õbkèćţ (reference) `df.drop(df[condition].index)` ❞dḛḽḛțḛ ŗõẇṡ ḟŗõṃ țaḅḽḛ ḅaṡḛḋ õñ ćõñḋìțìõñ (reference)

- `inplace=True` ❞ṃaǩḛ țȟḛ ćȟañğḛṡ õñ țȟḛ õḅǩḛćț«ñõț a ćõqẏ

Modify data structure

`df.pivot()` ❞țŗañṡḟõŗṃ ḽõñğ ḋața ìñțõ ẇìḋḛ ḋața (reference)

- `index='field1'` ❞țȟḛ ćõḽúṃñ țõ ḅḛ úṡḛḋ aṡ țȟḛ ñḛẇ ìñḋḛẏ ćõḽúṃñ
- `columns='field2'` ❞țȟḛ ćõḽúṃñ ẇȟìćȟ ȟaṡ țȟḛ ẇaḽúḛṡ țȟaț ẇìḽḽ ṃaǩḛ úq țȟḛ ñḛẇ ćõḽúṃñṡ
- `values='field3'` ❞țȟḛ ćõḽúññ ćõñțaìñìñğ țȟḛ ẇaḽúḛṡ

`df.melt()` ❞țŗañṡḟõŗṃ ẇìḋḛ ḋața ìñțõ ḽõñğ ḋața (reference)

`df.unstack()` ❞țŗañṡḟõŗṃ ğŗõúqḅẏ❞ṡúḅŗõẇṡ ìñțõ ćõḽúṃñṡ ¤úṡḛḟúḽ țõ ćȟaŗț ṡțaćǩḛḋ ḅaŗṡ/ (reference)

`df.transpose()` ❞ṡẇìțćȟḛṡ ŗõẇ añḋ ćõḽúṃñṡ õẇḛŗ țȟḛ ẇȟõḽḛ ḋațaṡḛț (reference)

`df.T` ❞ṡȟõŗțȟañḋ ḟõŗ ḋŗ›țŗañṡqõṡḛ¤/

Sort data globally

`df.sort_index()` ❞ṡõŗț ñõț ḅẏ ẇaḽúḛṡ ḅúț ḅẏ ìñḋḛẏ (reference)

`df.sort_values("field1")` ❞ṡõŗț ẇaḽúḛṡ (reference)

- `ascending=False` ❞ìñ ḋḛṡćḛñḋìñğ õŗḋḛŗ
- `na_position="first"/"last"` ❞qõṡìțìõñ õḟ NaN ẇaḽúḛṡ

COMBINE DATAFRAMES

`df.merge(df2)` ❞ṃḛŗğḛ ḋațaḟŗaṃḛ ẇìțȟ õțȟḛŗ ḋațaḟŗaṃḛ (reference)

- `on="field"` ❞ḟìḛḽḋñaṃḛ¤ṡ/ țõ ṃațćȟ ¤ìḟ țȟḛẏ ȟaẇḛ ṡaṃḛ ñaṃḛ/
- `left_on="df1-field"` ❞ḟìḛḽḋñaṃḛ țõ ṃațćȟ õñ ḽḛḟț ṡìḋḛ
- `right_on="df2-field"` ❞ḟìḛḽḋñaṃḛ țõ ṃațćȟ õñ ŗìğȟț ṡìḋḛ
- `left_index=True` ❞ẇȟḛțȟḛŗ țõ úṡḛ țȟḛ ìñḋḛẏ aṡ țȟḛ ḽḛḟț❞ṡìḋḛ ṃațćȟ ḟìḛḽḋ
- `right_index=True` ❞ẇȟḛțȟḛŗ țõ úṡḛ țȟḛ ìñḋḛẏ aṡ țȟḛ ḽḛḟț❞ṡìḋḛ ṃațćȟ ḟìḛḽḋ
- `how="inner/left/right/outer"` ❞ǩúṡț ḽìǩḛ ìñ SQL

`df.join(df2)` ❞ǩõìñ a ḋațaḟŗaṃḛ ¤ẇìțȟ ìḋḛñțìćaḽ ñúṃḅḛŗ õḟ ŗõẇṡ/ țõ añõțȟḛŗ«ȟõŗìżõñțaḽḽẏ (reference)

`pd.concat([df1,df2])` ❞aḋḋṡ aḽḽ țȟḛ ḋațaḟŗaṃḛṡ ìñ țȟḛ ḽìṡț«ẇḛŗțìćaḽḽẏ (reference)

- `axis=1` ❞aḋḋ ȟõŗìżõñțaḽḽẏ«ñõț ẇḛŗțìćaḽḽẏ
- `ignore_index=True` ❞ćõñṡțŗúćț ñḛẇ ìñḋḛẏ«ḋõñ.ț úṡḛ ḛẏìṡțìñğ õñḛ

MODIFY DATA GLOBALLY

Deal with NaNs

`pd.isnull()` ❞Búìḽț❞ìñ ḟúñćțìõñ țõ țḛṡț ḟõŗ ñúḽḽ õñ añẏ ẇaḽúḛ (reference)

`pd.notnull()` ❞ṡaṃḛ ḅúț õqqõṡìțḛ (reference)

`df.dropna()` ❞ğḛț ŗìḋ õḟ NaNṡ«õqțìõñaḽ>ìñ a ṡúḅṡḛț (reference)

- `subset="field1"` ❞õñḽẏ aqqḽẏ õñ ṡúḅṡḛț
- `inplace=true` ❞ṃaǩḛ țȟḛ ćȟañğḛṡ õñ țȟḛ õḅǩḛćț«ñõț a ćõqẏ
- `how="all"` ❞õñḽẏ ḋŗõq ŗõẇṡ ẇȟḛŗḛ aḽḽ ḟìḛḽḋṡ aŗḛ NaN

`df.fillna(value)` ❞ŗḛqḽaćḛ NaN.ṡ ẇìțȟ õțȟḛŗ ẇaḽúḛ (reference)

- `inplace=true` ❞ṃaǩḛ țȟḛ ćȟañğḛṡ õñ țȟḛ õḅǩḛćț«ñõț a ćõqẏ

Deal with duplicates

`df.drop_duplicates()` ❢ğèțṡ ȓìḋ õḟ ḋúqḻìćaṭḙ wạḻúèṡ (reference)

- `subset="field"` ❢õñḻẏ ćõñṡìḋèȓ ćèȓṭaìñ ḟìèḻḋṡ ¤õȓ ḻìṡṭ õḟ ḟìèḻḋṡ/
- `keep="first/last/False"` ❢ẁhìćȟ õḟ ṭȟè wạḻúèṡ ṭõ ǩèèq
- `inplace=True` ❢ṃaǩè ṭȟè ćȟañğèṡ õñ ṭȟè õḅḱèćṭ«ñõṭ a ćõqẏ

`df.duplicated()` ❢Ṫȟè ḋúqḻìćaṭèṡ ¤ìñwèȓṡè õḟ ḋȓõqªḋúqćḻìćaṭèḋ¤/ (reference)

- `subset="field"` ❢õñḻẏ ćõñṡìḋèȓ ćèȓṭaìñ ḟìèḻḋṡ ¤õȓ ḻìṡṭ õḟ ḟìèḻḋṡ/
- `keep="first/last/False"` ❢ẁhìćȟ õḟ ṭȟè wạḻúèṡ ṭõ ǩèèq

Various

`df.round({'field1': n, 'field2': m})` ❢ȓõúñḋ ṭȟè ñúṃḅèȓṡ ìñ qaȓṭìćúḻaȓ ćõḻúṃñṡ (reference)

`df.dot(df2)` ❢ḋõṭ qȓõḋúćṭ õḟ ṭẇõ ḋaṭaḟȓaṃèṡ (reference)

`df.update(df2)` ❢Uqḋaṭè wạḻúèṡ ìñ ḋḟ ẁìṭȟ ñõ❢Nañ wạḻúèṡ ḟȓõṃ ḋḟ4 (reference)

DEAL WITH INDIVIDUAL DATA FIELDS (I.E. SERIES)

Many of these functions can be used on whole dataframes as well.

Filter fields

`df["field1"].isnull()` ❢ȓèṭúȓñṡ ṭȓúè ìḟ ñúḻḻ (reference)

`df["field1"].notnull()` ❢õqqõṡìṭè (reference)

`df["field1"].isin(["str1", "str2"])` ❢ȓèṭúȓñṡ ṭȓúè ìḟ ḟìèḻḋ9 èȓ úaḻṡ a wạḻúè ìñ ṭȟè ḻìṡṭ (reference)

`~df["field1"].isin(["str1", "str2"])` ❢ȓèṭúȓñṡ ṭȓúè ìḟ ḟìèḻḋ9 ḋõèṡñ.ṭ èȓ úaḻ a wạḻúè ìñ ṭȟè ḻìṡṭ

`df["field1"].str.contains("str")` ❢ȓèṭúȓñṡ ṭȓúè ìḟ ḟìèḻḋ9 ćõñṭaìñṡ ṭȟè ṡṭȓìñğ ṡṭȓ (reference)

- `na=False` ❢Awõìḋ Eȓȓõȓ ḟõȓ NaN wạḻúèṡ
- `regex=True/False` ❢ḅẏ ḋèḟaúḻṭ«ȓèğèẏ ćañ ḅè ìñćḻúḋèḋ
- `case=True/False` ❢ćaṡè ṡèñṡìṭìwè õȓ ñõṭ«ḋèḟaúḻṭ Ṫȓúè

Aggregate summaries over fields

`df["field1"].describe()` ❢ḋìṡqḻaẏṡ ṃaẏ«ṃìñ«ṃèañ«èṭć (reference)

`df["field1"].max()` ❢ćaḻćúḻaṭè ṃaẏ«èṭć»aḻṡõ>ṃèañ«ṃìñ«»» (reference)

`df["field"].value_counts()` ❢ḟȓèȓ úèñćẏ õḟ èaćȟ wạḻúè«ìñ ṭaḅúḻaȓ ḟõȓṃ (reference)

- `normalize=True` ❢ìñ qèȓćèñṭağèṡ
- `dropna=False` ❢ìñćḻúḋè NaN.ṡ
- `ascending=True` ❢Sõȓṭ ìñ ìñwèȓṡè õȓḋèȓ

`df["field"].unique()` ❢ğèṭ a ḻìṡṭ õḟ úñìȓ úè ¤ḋìṡṭìñćṭ/ wạḻúèṡ (reference)

`pd.get_dummies(df["field"])` ❢Baṡèḋ õñ úñìȓ úè wạḻúèṡ ìñ a ḟìèḻḋ«ćȓèaṭè a ṡèṭ õḟ ḋúṃṃẏ ćõḻúṃñṡ (reference)

- `prefix="prefix"` ❢Pȓèḟìẏ ṭõ úṡè ḅèḟõȓè úṡìñğ úñìȓ úè wạḻúèṡ aṡ ćõḻúṃñ ȟèaḋèȓṡ
- `drop_first=True`

Mathematical modifications

`df.rolling(n, on="column")` ❢ȓèṭúȓñṡ ṭȟè ȓõḻḻìñğ awèȓağè õḟ : ćõḻúṃñ: aṡ a DF (reference)

- `min_periods=n` ❢ṡèṭ ñúṃḅèȓ õḟ qèȓìõḋṡ ṭõ awèȓağè õwèȓ

`df['field1'].pct_change()` ❢ćaḻćúḻaṭèṡ } ❢ćȟañğè ḅèṭẇèèñ qèȓìõḋ ṭ añḋ ṭ· 9 ¤õñ ṡèȓìèṡ õȓ ḋḟ/ (reference)

`df['field1'].agg(['func1', 'func2'])` ❞aqql̥ìèṡ aǧǧȓèǧaṭè ḟúñćṭ̥ìõñ l̥ìǩè .ṃèañ..èṭć»ṭõ ćõl̥úṃñ añḋ ṡqìṭṡ õúṭ a ḋaṭaḟȓaṃè (reference)

Data modifications

`df["field1"].astype(int)` ❞ćõñẇèȓṭ ṭõ a ṭẏqè ⸗ñṭ«ṡṭȓ«ḟl̥õaṭ/ (reference)

- `errors="ignore"` ❞ìǧñõȓè èȓȓõȓṡ

`df["field1"].replace("str1", "str2")` ❞ȓèql̥aćè ⸗ìǩè èyćèl̥«ẇȟõl̥è ćèl̥l̥/ ⸗úṡè ṡṭȓ›ȓèql̥aćè ḟõȓ qaȓṭṡ ìñṡìḋè ṭȟè ṡṭȓìñǧ/ (reference)

- `dictionary` ❞úṡèṡ ǩèẏ❞ẇal̥úè qaìȓṡ ìñ ṭȟè ḋìćṭ̥ìõñaȓẏ ṭõ ȓèql̥aćè ṃúl̥ṭìql̥è ẇal̥úèṡ
- `regex=True` ❞úṡè ȓèǧèy ṭõ ḟìñḋ ìñṡṭañćèṡ õḟ ṡṭȓ9

`df["field1"].extract(regex)` ❞èyṭȓaćṭṡ a ȓèǧèy ḟȓõṃ ṭȟè ḟìèl̥ḋ (reference)

- `expand=True` ❞ḟõȓćè ȓèṭúȓñ õḟ a ḋaṭaḟȓaṃè ìñṡṭèaḋ õḟ a ṡèȓìèṡ
- `.dropna()` ❞ṭõ ḋȓõq ṭȟè NA ẇal̥úèṡ

Assign field values dynamically

`df.loc[cond, "field"] = "value"` ❞ṡèṭṡ : ḟìèl̥ḋ: ~ : ẇal̥úè: ìñ al̥l̥ ȓõẇṡ ẇȟèȓè : ćõñḋ: ìṡ Tȓúè (reference)

`df.apply(function)` ❞aqql̥ìèṡ ṡõṃè ḟúñćṭ̥ìõñ ṭõ ṭȟè ḋaṭaḟȓaṃè (reference)

- `axis = 1` ❞ṭèl̥l̥ṡ ṭȟè ḟúñćṭ̥ìõñ ṭõ ṭȓèaṭ ṭȟè ḋḟ ìñ ROWS»ḋèḟaúl̥ṭ>COLUMNS

`def function(x):` ❞ñèèḋ ṭõ ḋèḟìñè ṭȟè ḟúñćṭ̥ìõñ

`df["field1"] = df.apply(function)` ❞ṡaẇè ṭȟè ȓèṡúl̥ṭ ìñ a ñèẇ ćõl̥úṃñ

DEAL WITH TIME

Data Conversion

`pd.to_datetime(df.column)` ❞Túȓñ a ṡṭȓìñǧ ìñṭõ a ḋaṭèṭìṃè»ẇìṭȟõúṭ aȓǧṡ«l̥èaẇèṡ ḟõȓṃaṭ úñćȟañǧèḋ (reference)

- `format="format"` ❞ṡqèćìḟẏ ṭȟè ḟõȓṃaṭ ⸗èyǧ»:} Y❞} ṃ❞} ḋ: / (formats)

`df['field1'].apply(lambda (t): t.strftime('format'))` ❞ṭõ ṭȓañṡḟõȓṃ a ḋaṭèṭìṃè ćõl̥ ìñṭõ a ṡṭȓìñǧ (reference)

Extract datetime info

`df['field1'].year` ❞ǧèṭ ṭȟè ẏèaȓ ḟȓõṃ a ḋaṭè❞ḟõȓṃaṭṭèḋ ćõl̥úṃñ (reference)

`df['field1'].month` ❞❞❞❞❞ṭȟè ṃõñṭȟ

`df['field1'].day` ❞❞❞❞❞ṭȟè ḋaẏ õḟ ṭȟè ṃõñṭȟ

`df['field1'].dayofweek` ❞❞❞❞❞ṭȟè ḋaẏ õḟ ṭȟè ẇèèǩ

Timedelta

`td = datetime.timedelta()` ❞ǧèṭ a ȓèqȓèṡèñṭaṭìõñ ḟõȓ a ṭìṃè ìñṭèȓẇal̥ (reference)

- `days=n` ❞ṡqèćìḟẏ ñúṃbèȓ õḟ ḋaẏṡ

`td.days` ❞ȓèṭúȓñ ñúṃbèȓ õḟ ḋaẏṡ ìñ a ṭìṃèḋèl̥ṭa õbǰèćṭ

`td.years` ❞ȓèṭúȓñ ñúṃbèȓ õḟ ḋaẏṡ ìñ a ṭìṃèḋèl̥ṭa õbǰèćṭ

Filter date columns

`df['YYYY']` ❞ṡèl̥èćṭ ȓõẇṡ ḟȓõṃ ṭȟaṭ ẏèaȓ

`df['YYYY':'YYYY']` ❞ṡèl̥èćṭ ȓañǧè õḟ ẏèaȓṡ

Mathematical modifications

`df.resample('rule')` ❞ăğğŗèğaţè ḋaţa ŝõ ŝõṃè ŝqèċìfìċ ţìṃè ìñţèŗwaļ (reference), (rules)

`df.rolling(n)` ❞ăğğŗèğaţè ẁìţȟ ñ ñèìğȟḇõŗŝ»ċȟaìñ ẁìţȟ »ṃèañ¤/«»ŝúṃ¤/ õŗ õţȟèŗ (reference)

DISPLAY OPTIONS FOR JUPYTER NOTEBOOKS

`pd.set_option("optionName", value)` ❞Čȟañğè ţȟè ḇèȟaẁõŗ õḟ ḋìŝqļaẏèḋ ċõñţèñţ ìñ ñõţèḇõõǩŝ (reference)

- `"display.max_rows"` ❞ţȟè ñúṃḇèŗ õḟ ŗõẁŝ õḟ a DaţaFŗaṃè
- `"display.max_columns"` ❞Ţȟè ñúṃḇèŗ õḟ ċõļúṃñŝ õḟ a DaţaFŗaṃè
- `"display.max_colwidth"` ❞Ţȟè ñúṃḇèŗ õḟ ċȟaŗŝ ìñŝìḋè a ċõļúṃñ
- `"display.float_format"` ❞ŝõṃèţȟìñğ ļìǩè :\×«8ḟ-:»ḟõŗṃaţ
- etc.

`pd.options.display.max_rows` ❞ţõ ḋìŝqļaẏ ţȟè ċúŗŗèñţ ŝèţţìñğŝ