(df

.select\_dtype(int)

.describe()

)

(df

.query(‘`Lot Frontage’.isna()’)

)

Observe the coumn names are inside `` not ‘ ‘. But if the column names don’t have the space, no need to use this.

Query kind of code

(df

.filter(like=’Yr’)

)

This pulls out columns which names might have Yr as sub string also in it.

(df

.filter(like=’Yr’)

.pipe(lamda df\_: df\_ [df\_.gt(2023).any(axis=’columns’)])

)

This gives data for 2023

Here any(asis=’columns’) gives us the Boolean output

(df

.rename(colums= lamda name: name.replace(‘Yr’, ‘Year’))

.filter(like = ‘Year’)

.pipe(lamda df\_:df\_ [df\_.gt(2023).any(axis=’columns’)])

)

This just changes the YR to Year first, and select all columns having Year, then greterthan 2023,

(df

.clip(upper=df [‘Year Built’].max()

.value\_counts()

.sort\_index()

)