

C: Selecting and dropping rows

Two conceptual modes for filtering data:

1. **Selecting**: choose rows to **keep** → drop everything else
2. **Dropping**: choose rows to **drop** → keep everything else

Methods for selecting:

- Select **by index** using `.loc[]`:

```
onondaga_row = means.loc['Onondaga']
```

Result: single record with 'Onondaga' as its index value

- Select **a cross section** from a dataframe with a multilevel index:

```
something = med_b.xs(3, level='type')
```

Result: all records with 3 as their value of the 'type' index

- Select **by position** using `.iloc[]` or **list subscripting**:

```
top_five = high_to_low.iloc[ 0:5 ]
```

```
top_five = high_to_low[ :5 ]
```

Result: first five records regardless of their index values

- Select **via a boolean** (True/False) series:

```
is_res = usable['PROP_CLASS'].between(200,299)
res = usable[ is_res ]
```

Result: records where 'PROP_CLASS' is between 200 and 299 inclusive

Variant: combining boolean and selection into one line

```
res = raw[ raw['YR_BLT'] >= 1980 ]
```

Result: records where raw['YR_BLT'] is at least 1980

- Selecting **via a query**:

```
trim = raw.query( "state == '36' and fuel == 'gas'" )
```

Result: records where 'state' is '36' and 'fuel' is 'gas'

Advantages: flexible, compact and clear

Note 1: ⚠ Argument is a string

Note 2: ⚠ Column names are **not** quoted

Note 3: Use backticks for names with spaces: `Cap MW`

Methods for dropping rows:

- Dropping records with missing data:

```
usable = raw.dropna( subset=key_vars )
```

Result:

Removes records with missing data in **key_vars**

Without the subset, removes records with **any** missing data

- Dropping duplicates:

```
weather = weather.drop_duplicates(subset='Local Hour')
```

Result:

Step 1: find records with duplicate values of 'Local Hour'

Step 2: drop all duplicates except the first one

Keep all records that aren't duplicates

- Dropping records by name:

```
trim = raw.drop(index=name_list)
```

Result:

Removes records having index values in name_list