

Selecting and filtering, and the warning that became an error

loc, iloc and boolean masks used deliberately — plus what SettingWithCopyWarning was warning about, why pandas 3 replaced it with ChainedAssignmentError, and what that changes for code you already have.

Pierre Robentz Cassion

Lesson 5 of 8

Python for Programme Data — Working the table

What this lesson covers

- Three ways to select, and when each is right
- Boolean masks
- The warning that became an error
- Adding and changing columns
- Sorting, and why it is not a ranking
- What comes next

Three ways to select, and when each is right — In Python

```
import pandas as pd
```

```
muac = read_register(RAW / "muac-screening-artibonite-2024.v1.csv")
```

Three ways to select, and when each is right

- By label, with `loc` — Rows by index value, columns by name

Three ways to select, and when each is right — In Python

```
muac.loc[muac["commune"] == "Gonaives", ["child_id", "muac_mm", "outcome"]]
```

Three ways to select, and when each is right

- By position, with `iloc` — Rows and columns by integer position

Three ways to select, and when each is right — In Python

```
muac.iloc[0]           # first row  
muac.iloc[:5, :3]      # first five rows, first three columns
```

Three ways to select, and when each is right

- A single column — is a Series; a list of columns is a DataFrame:

Three ways to select, and when each is right — In Python

```
muac["muac_mm"]           # Series  
muac[["muac_mm"]]         # DataFrame with one column  
muac[["commune", "muac_mm"]] # DataFrame with two
```

Boolean masks — In Python

```
severe = muac["muac_mm"] < 115
print(severe.sum())           # how many
print(severe.mean())         # what share
muac.loc[severe]
```

- Combining conditions

Boolean masks — In Python

```
# Parentheses are mandatory: & binds tighter than <  
under_five = (muac["age_months"] < 60) & (muac["muac_mm"] < 125)  
  
# Use & | ~ -- not and / or / not, which cannot operate elementwise  
either = (muac["muac_mm"] < 115) | (muac["oedema"] == True)
```

- Missing values are not False

Boolean masks — In Python

```
print(len(muac))  
print((muac["muac_mm"] < 125).sum())  
print(muac["muac_mm"].isna().sum())
```

```
# 4218  
# counts only measured children  
# the ones excluded silently
```

Boolean masks — In Python

```
measured = muac["muac_mm"].notna()
gam = measured & (muac["muac_mm"] < 125)

print(f"assessed: {measured.sum()}, GAM cases: {gam.sum()}")
print(f"GAM rate: {gam.sum() / measured.sum():.3f}")
```

Boolean masks

- The denominator is the point — Dividing by `len(muac)` instead of `measured.sum()` reports a lower prevalence for a...
- `isin`, `between`, `query`

Boolean masks — In Python

```
north = muac["commune"].isin(["Gonaives", "Terre-Neuve", "Anse-Rouge"])  
plausible = muac["muac_mm"].between(80, 220)
```

Boolean masks — In Python

```
impossible = muac["muac_mm"].notna() & ~muac["muac_mm"].between(80, 220)
```

Boolean masks — In Python

```
muac.query("age_months < 60 and muac_mm < 125")
```

The warning that became an error — Example

SettingWithCopyWarning:

A value is trying to be set on a copy of a slice from a DataFrame.

Try using `.loc[row_indexer, col_indexer] = value` instead

The warning that became an error — In Python

```
muac[muac["commune"] == "Gonaives"]["muac_mm"] = 0
```

The warning that became an error

- What pandas 3 does instead
- The behaviour is now defined — A selection never writes back to its parent — not sometimes, never
- Chained assignment tells you it did nothing

The warning that became an error — In Python

```
import pandas as pd

df = pd.DataFrame({"a": [1, 2, 3], "b": [10, 20, 30]})
df[df["a"] > 1]["b"] = 0
```

The warning that became an error — Example

```
ChainedAssignmentError: A value is being set on a copy of a  
DataFrame or Series through chained assignment. Such chained  
assignment never works ...
```

The warning that became an error — In Python

```
import warnings
```

```
warnings.simplefilter("error", pd.errors.ChainedAssignmentError)
```

The warning that became an error

- The correct form, and what changed for existing code

The warning that became an error — In Python

```
# One .loc, both axes, one operation.  
muac.loc[muac["commune"] == "Gonaives", "muac_mm"] = pd.NA
```

The warning that became an error

- `.copy()` and when you still want it

The warning that became an error — In Python

```
# This subset is going to be modified and is not a view of anything.  
gonaives = muac.loc[muac["commune"] == "Gonaives"].copy()  
gonaives["flag"] = gonaives["muac_mm"] < 115
```

Adding and changing columns — In Python

```
summary = (  
    muac  
    .loc[muac["muac_mm"].notna()]  
    .assign(  
        gam=lambda d: d["muac_mm"] < 125,  
        sam=lambda d: d["muac_mm"] < 115,  
    )  
)
```

Adding and changing columns — In Python

```
import numpy as np

muac["band"] = np.select(
    [muac["muac_mm"] < 115, muac["muac_mm"] < 125],
    ["severe", "moderate"],
    default="normal",
)
```

Adding and changing columns — In Python

```
muac.loc[muac["muac_mm"].isna(), "band"] = pd.NA
```

Sorting, and why it is not a ranking — In Python

```
muac.sort_values(["commune", "screening_date"], ascending=[True, False])
```

Sorting, and why it is not a ranking — In Python

```
muac.groupby("commune")["muac_mm"].mean().nsmallest(5)
```

What comes next

- You can select a subset and change it without wondering whether the change landed.

Read the full lesson, with runnable code

[https://data-analysis.cassion.dev/courses/python-for-programme-data/
python-05-selection-and-filtering](https://data-analysis.cassion.dev/courses/python-for-programme-data/python-05-selection-and-filtering)