Training on Web Scraping Prices for CPI

Reproducible Analytical Pipelines

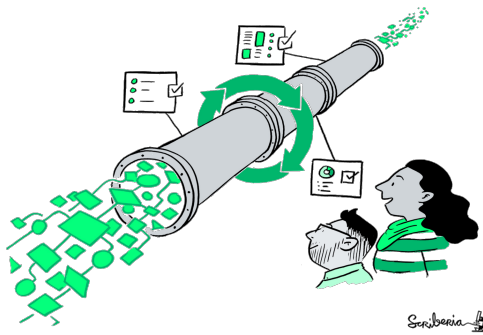
Christophe Bontemps & Serge Goussev

Why are Reproducible Analytical Pipelines good for you?

- It will make your (and your team's) (*working*) life easier.

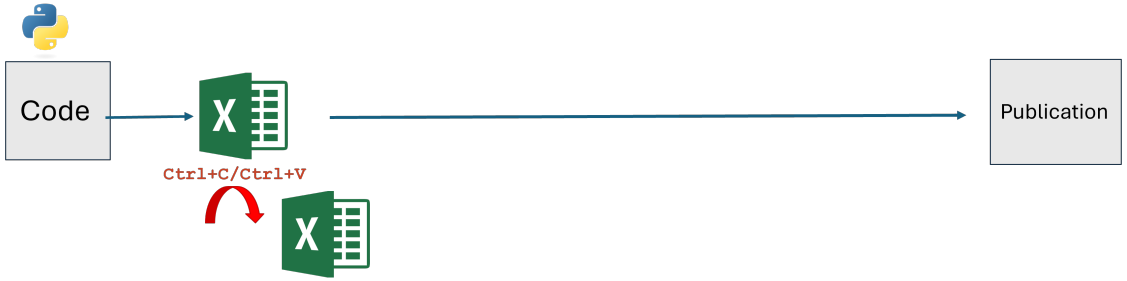
↪ Less confusion about where things are and how it works!
Lower cognitive load on day to day tasks!

- It is an efficient way to work
- It helps work faster
- It helps make the process of making official statistics more robust!
- It makes it easy to efficiently collaborate
- It will enhance your skills (and perhaps make you famous in your organization!)



USUAL PRACTICE: THEORY VS REALITY

You scrapped data from a website. Saving the file as Excel



Send that file by email to your collaborators

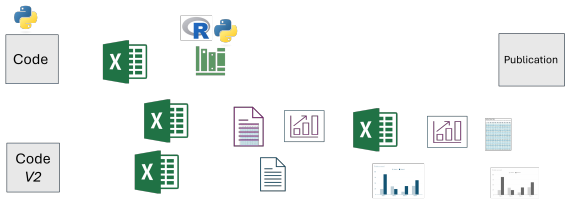
Usual practice: In the end

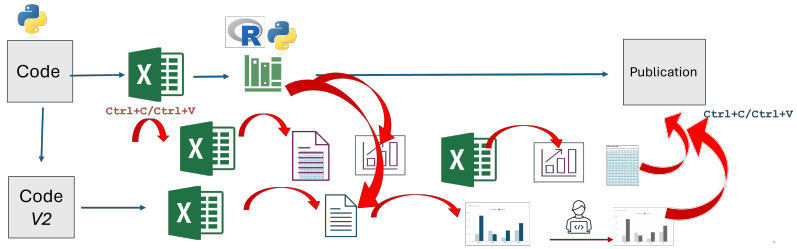
- Lots of files
- Cut and paste is not a reliable, reproducible approach!
- Your brain may remember (likely not!)...

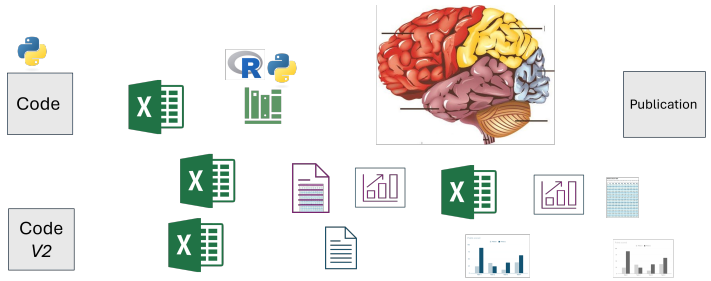
  ...all the steps...

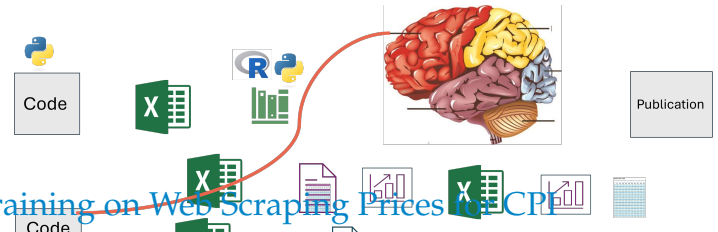
  .. in the right order..

  ...all of them !
- Or use (bad) "*tools*"









What are the issues?

- Errors due to cut and paste
- Errors are difficult to track
- Each operator has his/her own approach
- Several versions of code may coexist
- The steps aren't recorded
- Reproducibility is not granted

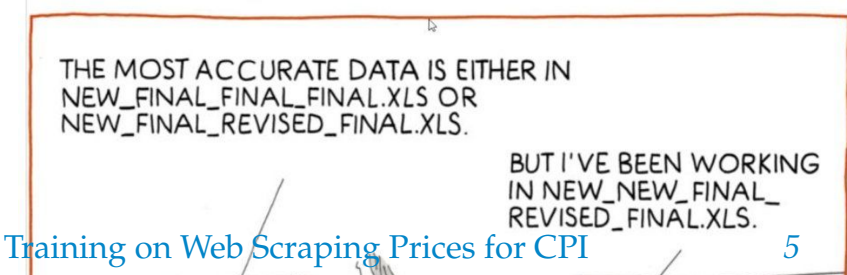
Excel: Why using Microsoft's tool caused Covid-19 results to be lost

By Leo Kelion
Technology desk editor

5 October 2020

The badly thought-out use of Microsoft's Excel software was the reason nearly 16,000 coronavirus cases went unreported in England.

And it appears that Public Health England (PHE) was to blame, rather than a third-party contractor.



Fundamental Principles of Official Statistics

- Clear mention of the **processes** used to produce statistics
- *To retain trust in official statistics, the **statistical agencies need to decide** according to strictly professional considerations, including scientific principles and professional ethics, **on the methods and procedures** for the collection, processing, storage and presentation of statistical data.*
- In short, **processes** are important!

Fundamental Principles of Official Statistics*

United Nations — For more information: unstats.un.org

The General Assembly,

Recalling recent resolutions[1] of the General Assembly and the Economic and Social Council highlighting the fundamental importance of official statistics for the national and global development agenda,

Bearing in mind the critical role of high-quality official statistical information in analysis and informed policy decision-making in support of sustainable development, peace and security, as well as for mutual knowledge and trade among the States and peoples of an increasingly connected world, demanding openness and transparency,

Bearing in mind also that the essential trust of the public in the integrity of official statistical systems and confidence in statistics depend to a large extent on respect for the fundamental values and principles that are the basis of any society seeking to understand itself and respect the rights of its members, and in this context that professional independence and accountability of statistical agencies are crucial,

Stressing that, in order to be effective, the fundamental values and principles that govern statistical work have to be guaranteed by legal and institutional frameworks and be respected at all political levels and by all stakeholders in national statistical systems,

Endorses the Fundamental Principles of Official Statistics set out below, as adopted by the Statistical Commission in 1994[2] and reaffirmed in 2013, and endorsed by the Economic and Social Council in its resolution 2013/21 of 24 July 2013:

* General Assembly Resolution 68/261 adopted on 29 January 2014. The "titles" of the Principles are not part of the original text.

[1] These include General Assembly resolution 64/267 on World Statistics Day and Economic and Social Council resolutions 2005/13 on the 2010 World Population and Housing Census Programme, 2006/6 on strengthening statistical capacity and 2013/21 on the Fundamental Principles of Official Statistics.

[2] For the original preamble used on the occasion of the initial adoption of the Fundamental Principles in 1994, see chapter V of the report of the Statistical Commission on its special session (*Official Records of the Economic and Social Council, 1994, Supplement No. 9* (E/1994/29)). More information on the Fundamental Principles and their history is available from the website of the Statistics Division.

Principle 1: ***Relevance, Impartiality, and Equal Access***
Official statistics provide an indispensable element in the information system of a democratic society, serving the Government, the economy and the public with data about the economic, demographic, social and environmental situation. To this end, official statistics that meet the test of practical utility are to be compiled and made available on an impartial basis by official statistical agencies to honour citizens' entitlement to public information.

Principle 2: ***Professional Standards, Scientific Principles, and Professional Ethics***
To retain trust in official statistics, the statistical agencies need to decide according to strictly professional considerations, including scientific principles and professional ethics, on the methods and procedures for the collection, processing, storage and presentation of statistical data.

Principle 3: ***Accountability and Transparency***
To facilitate a correct interpretation of the data, the statistical agencies are to present information according to scientific standards on the sources, methods and procedures of the statistics.

Principle 4: ***Prevention of Misuse***
The statistical agencies are entitled to comment on erroneous interpretation and misuse of statistics.

Principle 5: ***Sources of Official Statistics***
Data for statistical purposes may be drawn from all types of sources, be they statistical surveys or administrative records. Statistical agencies are to choose the source with regard to quality, timeliness, costs and the burden on respondents.

Principle 6: ***Confidentiality***
Individual data collected by statistical agencies for statistical compilation, whether they refer to natural or legal persons, are to be strictly confidential and used exclusively for statistical purposes.

Principle 7: ***Legislation***
The laws, regulations and measures under which the statistical systems operate are to be made public.

Principle 8: ***National Coordination***
Coordination among statistical agencies within countries is essential to achieve consistency and efficiency in the statistical system.

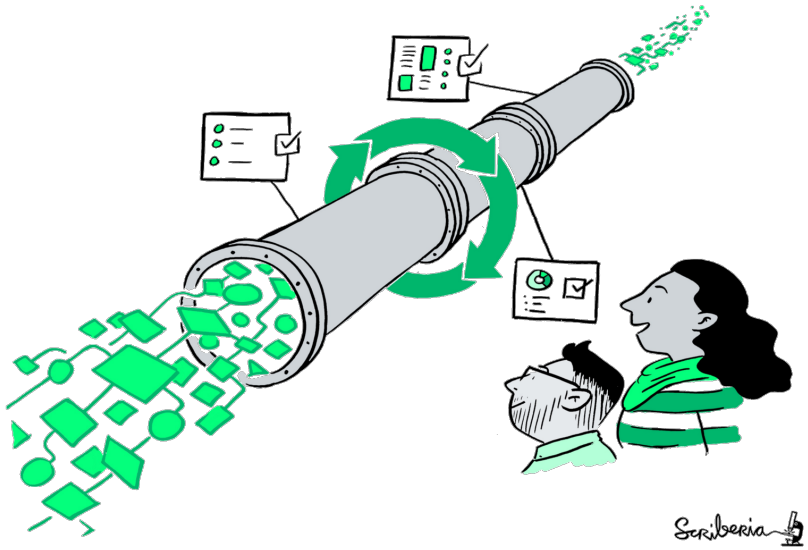
Principle 9: ***Use of International Standards***
The use by statistical agencies in each country of international concepts, classifications and methods promotes the consistency and efficiency of statistical systems at all official levels.

Principle 10: ***International Cooperation***
Bilateral and multilateral cooperation in statistics contributes to the improvement of systems of official statistics in all countries.

What is A Reproducible Analytical Pipeline?

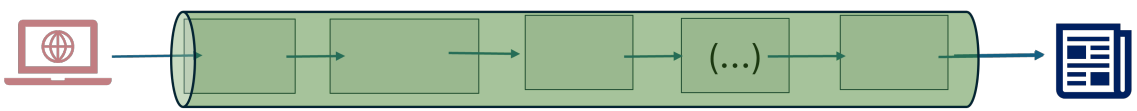
RAP could be thought of as an approach to working Adopting a set of technical skills and open-source best practices to adopt when creating a data output

- RAP is thus a robust process
- It is (*quite*) automated
- It is (*easily*) reproducible
- It minimizes the time to find and fix mistakes when they do occur
- It leads to fast processes (see Vanuatu Experience)



What does a RAP look like?

Idealy, Input (website) and output (report) are linked



RAP PRINCIPLES:

- Automation (*as much as you can*)
  - ↪ Avoid manual work
- Reusable (modular) code
  - ↪ Build blocs, update blocs, change blocs, test blocs
- Transparency
  - ↪ Show what you, do what you say
- Use open source tools
  - ↪ Free, reusable, huge community
- Version control
  - ↪ Easy to track code, easy to share, easy to update, ...
- Good coding practices
  - ↪ Write for humans, not for machines
- Testing
- Peer-review

RAP PRINCIPLES:

These principles translate into:

Good Practices

+

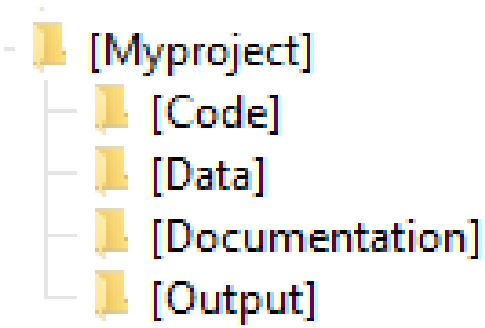
Good Tools

↪ We'll detail some of these practices and tools

GOOD PRACTICES: ORGANIZE YOUR WORK

Have a clear directory structure

- Separate files into data, code, docs, etc.
- Make directories portable (relative path)



Example of a well-organized directory structure.

Usual
mydata = pd.read_csv("c://ESCAP/Webscraping/Data/WebData.csv")

Better
Assuming your code is in c://ESCAP/Webscraping/Code/
mydata = pd.read_csv("../Data/WebData.csv")

GOOD PRACTICES: ORGANIZE YOUR WORK

Use naming conventions:

For **files/code**

- Avoid lazy names
- Meaningful files names
- Order of execution

Usual

```
prog1.py
prog2.py
Stat.py
progC.py
progP.py
```

Better

```
Scraping_Data.py
Cleaning_Data.py
Stats_Tables.py
Classification.py
Price_CPI.py
```

Even better

```
01_Scraping_data.py
02_Cleaning_data.py
03_Classification.py
04_Stats_Tables.py
04_Price_CPI.py
```

GOOD PRACTICES: ORGANIZE YOUR WORK

Use naming conventions:

For **outputs**

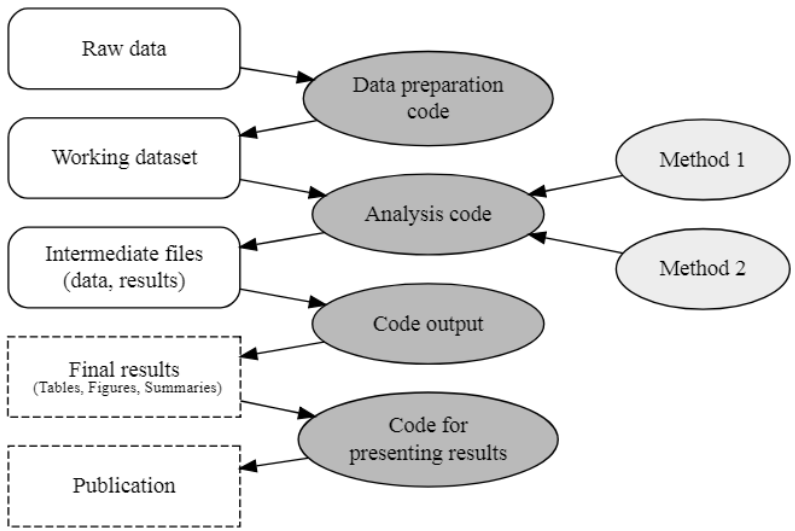
- Avoid numbering
- Explicit type of output

Usual	Better
`Table1.pdf`	`Stat_Desc_Table.pdf`
`Table2.pdf`	`Price_Stat_Table.pdf`
`Graph.jpg`	`Dress_Prices_Graphic.jpg`
`Model.csv`	`All_prices_Results.csv`

GOOD PRACTICES FOR AUTOMATION

Keep track of the workflow:

- Cut and paste should be avoided
- Every step of the process is coded
- Manage (and draw) the workflow

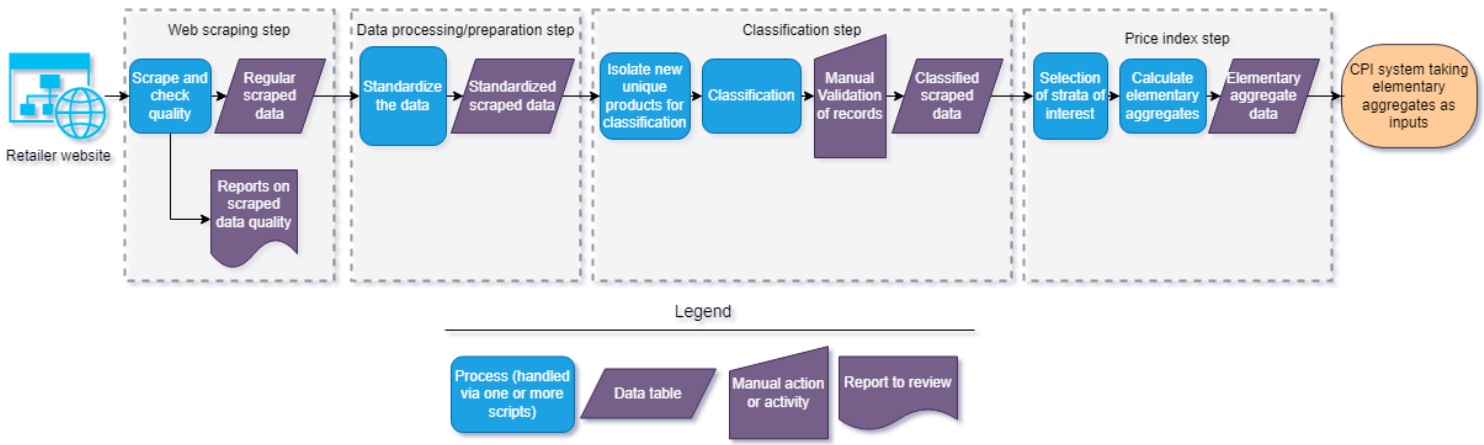


Example of a simple workflow.

GOOD PRACTICES FOR AUTOMATION

Keep track of the workflow:

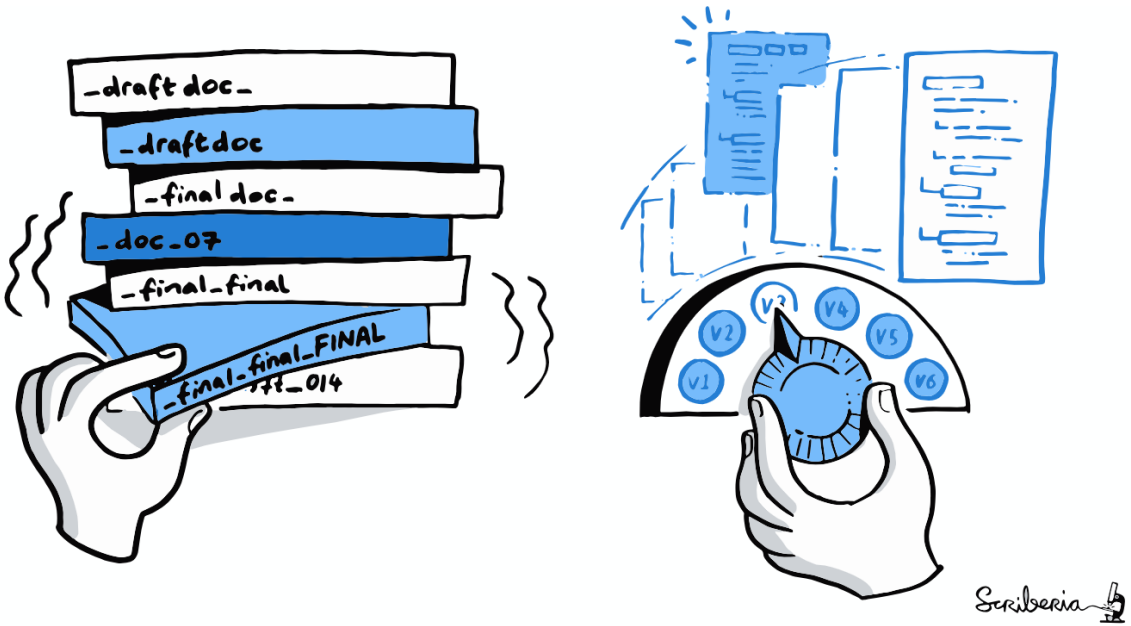
Here is a workflow from web scraping to elementary aggregate workflow. We'll cover it more next week!



Created by Serge Goussev.

VERY GOOD PRACTICES

Use a version control system (Git/GitHub)



More on Version Control later

GOOD CODING PRACTICES: CODE FOR OTHERS

Program with style:

Use *literate programming*

"Let us concentrate rather on explaining to humans what we want the computer to do"

D. Knuth (1984)

"(. . .) code is read much more often than it is written"

Guido van Rossum (2013 -PEP8)

PEP stands for *Python Enhancement Proposals*

Use conventions on layout (Comments, indentation,...)

Contents

- Introduction
- A Foolish Consistency is the Hobgoblin of Little Minds
- Code Lay-out
  - Indentation
  - Tabs or Spaces?
  - Maximum Line Length
  - Should a Line Break Before or After a Binary Operator?
  - Blank Lines
  - Source File Encoding
  - Imports
  - Module Level Dunder Names
- String Quotes
- Whitespace in Expressions and Statements
  - Pet Peeves
  - Other Recommendations
- When to Use Trailing Commas
- Comments
  - Block Comments

PEP 8 – Style Guide for Python Code

Author: Guido van Rossum <guido at python.org>, Barry Warsaw <barry at python.org>, Alyssa Coghlan <ncoghlan at gmail.com>
Status: Active
Type: Process
Created: 05-Jul-2001
Post-History: 05-Jul-2001, 01-Aug-2013

▶ Table of Contents

Introduction

This document gives coding conventions for the Python code comprising the standard library in the main Python distribution. Please see the companion informational PEP describing style guidelines for the C code in the C implementation of Python.

GOOD CODING PRACTICES: CODE FOR OTHERS

Program with style

- Avoid ambiguities
- Avoid changing units

Usual

```
df['sex'] = np.where(df['gender'] ==
'1001', 1, 2)
```

Better

```
df['female'] = np.where(df['gender'] ==
'1001', 1, 0)
df['male'] = np.where(df['gender'] !=
'1001', 1, 0)
```

Usual

```
df['gdp'] = df['gdp'] / 118.722
```

Better

```
df['gdp_US'] = df['gdp'] / 118.722
```

Even better

```
US_Vanu_exch_rate = 118.722
df['gdp_US'] = df['gdp']/
US_Vanu_exch_rate
```

Good practices: Modularity

Create reusable objects

- Store values

  Avoid repetitions

- Use functions
- Use independent blocks

Usual

```
Current_Data = Mydata[Mydata['year'] == 2023]
```

Better

```
Current_year = 2023
Current_Data= Mydata[Mydata['year'] ==
Current_year]
```

Usual

```
# - Exports for Beef -
data = Mydata[Mydata['export'] == 'Beef']
plt.plot(data['Year'], data['Value'])
plt.title('Export for Beef')
# - Also for Kava -
data = Mydata[Mydata['export'] == 'Kava']
plt.plot(data['Year'], data['Value'])
plt.title('Export for Kava')
# - Also for ...  -
```

Better

```
# Defining a generic function
def plot_export(export_type):
 data = Mydata[Mydata['export'] ==
                                    export_type]
 plt.plot(data['Year'], data['Value'])
 plt.title(f'Export for {export_type}')
```

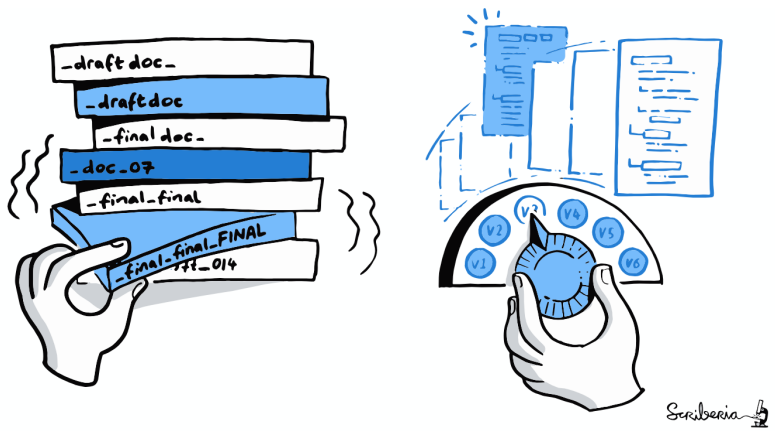
VERSION CONTROL KEEPS TRACKS OF YOUR WORK

Tracking three **W** questions:

What changes?

Who made the changes?

When were the changes made?

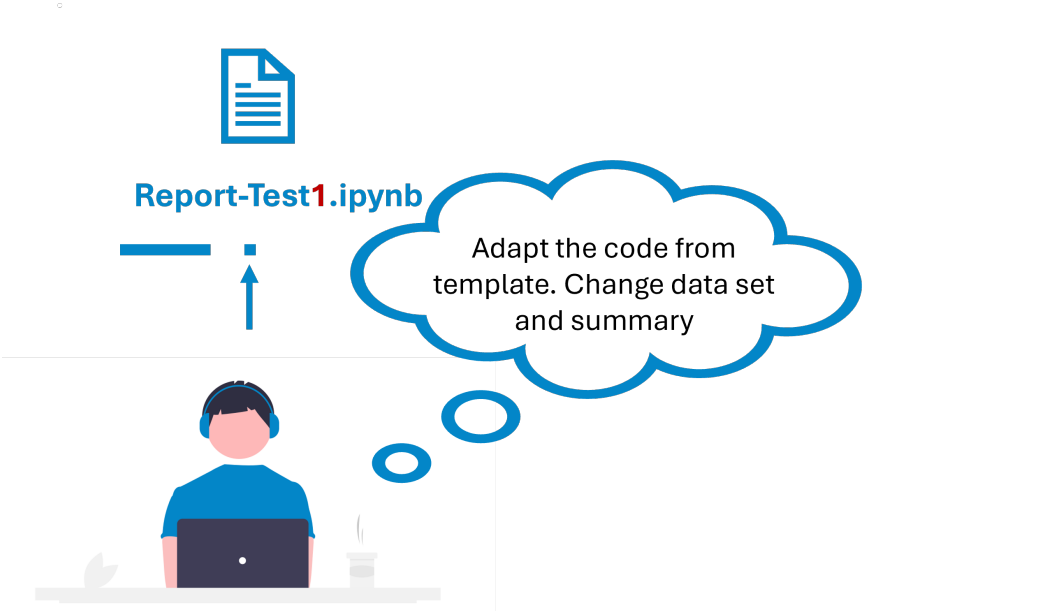


Source: The Turing Way project

Transparency, Accountability & Reproducibility

- Version control provides a detailed history of changes
- Each modification is attributed to a specific user
- Promotes accountability, transparency & reproducibility

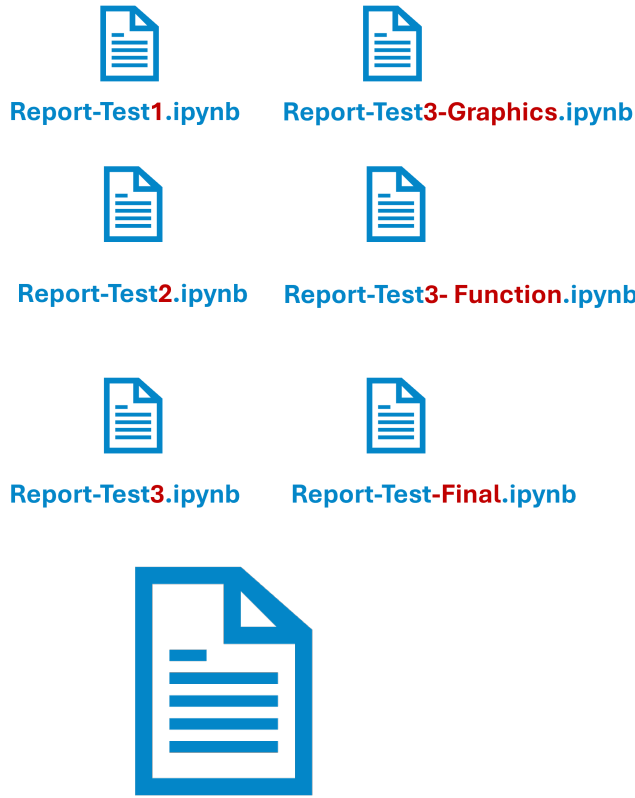
File evolution without Version Control



File evolution without Version Control

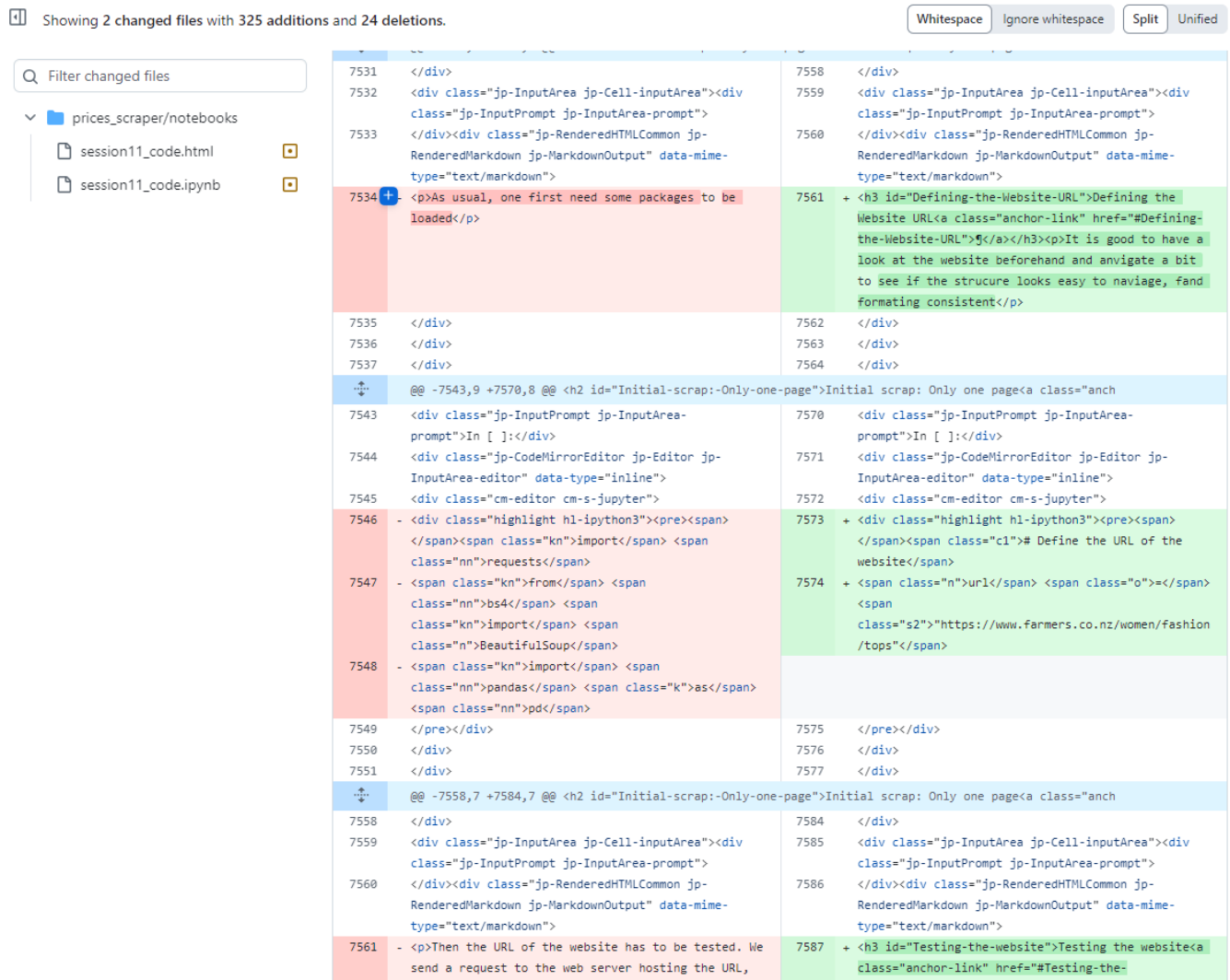
Usual ways to keep track of changes:

- New file after each change
- ↪ Need to open each file to see the change
- ↪ Names have to be explicit
- Only the last file with lots of comments
- Not fulfilling the 3 **W**...



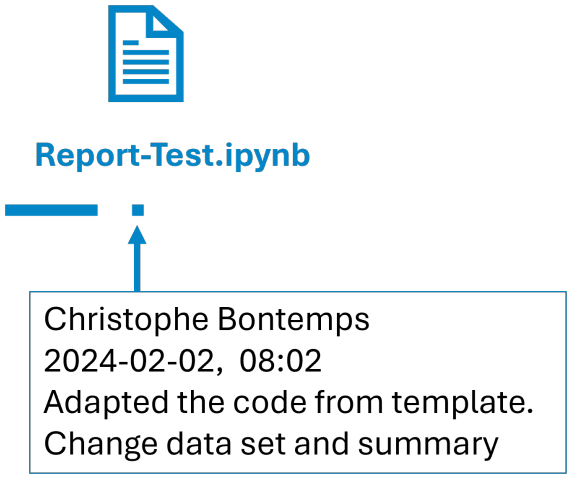
File evolution with Version Control

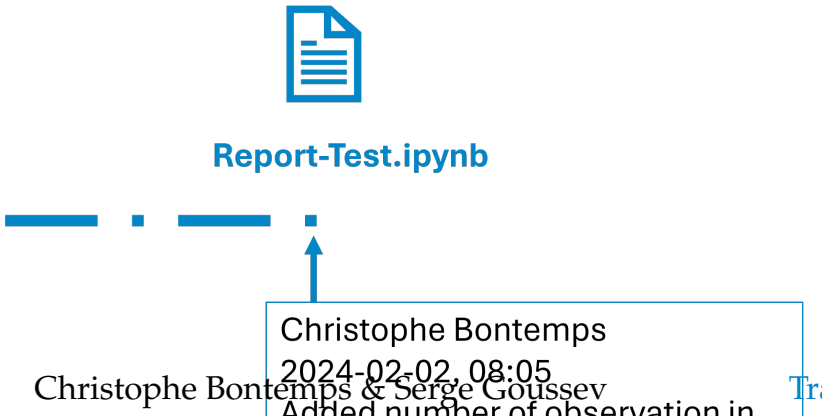
You can see exactly what has been going on!



File evolution with Version Control

Record a message (*commit*) for each change!





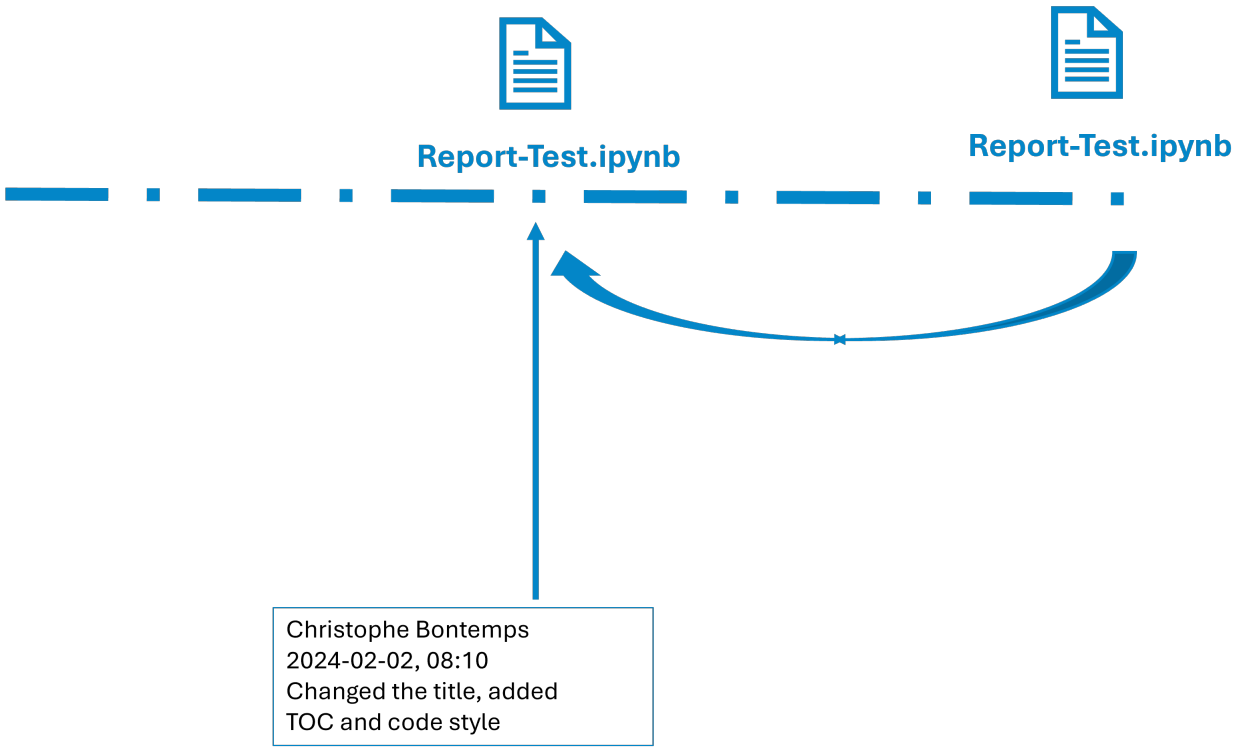
THE HISTORY OF THE FILE IS RECORDED!

Each version is documented (with *commits*)

Each version embeds the full history!

GOING BACK (*revert*) IS POSSIBLE

It is possible to review previous version...

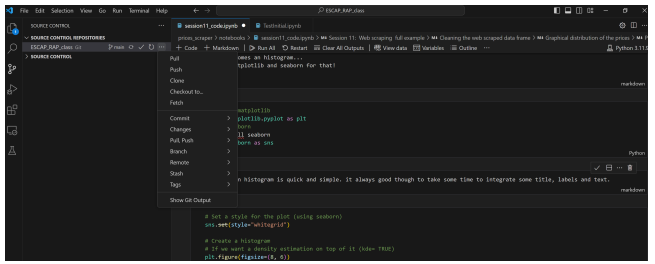


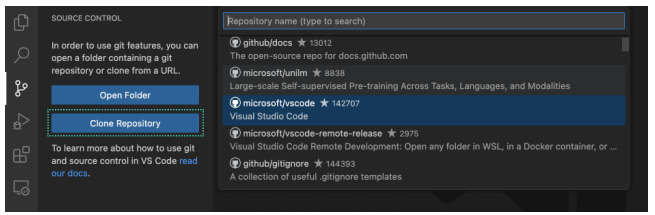
...to compare the changes...

Good News!

Version Control will help you

- Version Control is integrated in VS Code (*& RStudio*)
- ↪ Simple operations are easy
- Collaborate on a project
- ↪ Track changes of others
- Git seems "*unfriendly*" but it is your friend
- ↪ Takes time and patience
- Git works *mostly* in command mode
- ↪ You will learn that too!





"*Visual Studio Documentation*"

Version Control in a Nutshell

A Version Control systems:

- Keeps track of all changes
- Allows you to ignore anything you don't want to version control (such as internal data) in the (*.gitignore*)
- Allows reviewing stages of development
- Allow collaborating on projects
- Comes with different tools (Git, GitHub, GitLab, etc..)!

↪ GitLab can be set up on an internal NSO network.

- Backups your work

Takeaways

- There many levels of RAP (a full spectrum)
  ↪ Start small, be an advocate for others
  ↪ Increase complexity when ready
- Good practices starts with our own practices
  ↪ KISS: Keep it Simple, Stupid
- Automate little by little
- Version control is a life-changer
- Building a RAP is a collective process

Useful resources

- NHS Community of Practice
- This course website (created by Serge Goussev)
- Vanuatu Bureau of Statistics implementation of RAP
- SIAP's (free) online RAP course
- The UK government RAP website.
- UK best practice documentation.
- A free RAP course to teach you all you need to know.
- How the Data Science Campus sets its coding standards.
- A new open-source book from the Alan Turing institute setting out how to do reproducible data science.