

Motivation



Issues



RAP



3 Principles



Version Control



Resources



Training on Web Scraping Prices for CPI

Reproducible Analytical Pipelines

Christophe Bontemps & Serge Goussev



FUNDAMENTAL PRINCIPLES OF OFFICIAL STATISTICS

- ▶ Clear mention of the processes used to produce statistics



Fundamental Principles of Official Statistics*

For more information: unstats.un.org

The General Assembly

Recommending resolutions* of the General Assembly and the Economic and Social Council highlighting the fundamental importance of official statistics for the rational and effective functioning of agencies.

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 understanding and trade among the States and peoples of an increasingly connected world, demanding openness and transparency.

Recommiring 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 principles of official statistics, on 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 the production of official statistics by legal and institutional frameworks and be respected at all political levels and by all stakeholders in national statistical systems,

Endorsing the Fundamental Principles of Official Statistics, as recommended by the Statistical Commission in 1994* and reaffirmed in 2013, and endorsed by the Economic and Social Council in its resolution 205/21 of 24 July 2013;

* General Assembly resolution 65/211 adopted on 29 January 2014. The “10” of the Principles are part of the original text.

These include General Assembly resolution 46/240 on 20 December 1991 and Economic and Social Council resolution 205/13 on the 2013 World Programme of Action for Statistical Development on strengthening statistical capacity and 2030/2 on the Fundamental Principles of Official Statistics.

* For a copy of the principles, see the Statistical Commission on its special session (Official Report) of the Economic and Social Council. 1994. *Statistical Principles for Official Statistics*. United Nations. The Fundamental Principles and their history is available from the website of the Statistics Division.

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.*



Fundamental Principles of Official Statistics*

For more information: unstats.un.org

The General Assembly

Reviewing recent resolutions¹ of the General Assembly and the Economic and Social Council highlighting the fundamental importance of official statistics for the national development process, the statistical agencies:

Desiring in view 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;

Recommending 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 principles of official statistics, on 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 the production of official statistics must be legal and institutional frameworks and be respected at all political levels and by all stakeholders in national statistical systems;

Endorsing the Fundamental Principles of Official Statistics, developed by the Statistical Commission in 1994² and reaffirmed in 2013, and endorsed by the Economic and Social Council in its resolution 205/21 of 24 July 2013;

* General Assembly resolution 65/217 adopted on 29 January 2011. The "Values of the Principles" are part of the original text.

These include General Assembly resolution 46/200 on the role of official statistics and Economic and Social Council resolution 205/21 on the 2010 World Population Conference, both of which stress the importance of strengthening statistical capacity and 2030/2030 on the implementation of the Sustainable Development Goals.

² For a copy of the principles, please refer to the report of the initial adoption of the Fundamental Principles in 1994, see the document E/CN.3/Sub.2/1994/13, Statistical Commission on its special session (Official Report) of the Economic and Social Council. 1994. *Sustainable development and the role of official statistics*. The text of 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 conditions. 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 must follow strict professional standards, 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

The statistical agencies are to present information according to scientific standards on the sources 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, including administrative records, surveys and samples, provided that the statistical agencies are able 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: Legitimacy

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 countries.

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*

For more information: unstats.un.org

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 conditions in the country. 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 demonstrate adherence to professional conventions, 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
 The statistical agencies are to present information according to scientific standards on the sources 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, including administrative records, surveys and experiments. 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: Legitimacy
 The laws, regulations and measures under which the statistical systems operate are to be made public.

Principle 8: National Coordination
 Coordination among statistical agencies or 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 countries.

* General Assembly resolution 65/217 (adopted on 29 January 2011). The "10th" of the Principles are part of the original text.

This includes General Assembly resolution 46/200 on the Fundamental Principles of Official Statistics and Economic and Social Council resolution 2005/13 on the 2010 World Programme of Action for Statistical Development on strengthening statistical capacity and 2010/2011 on the Harmonization of Statistical Official Statistics.

For further reading, please see the report of the initial adoption of the Fundamental Principles in 1994, see also the report of the Statistical Commission on its special session (Official Report of the Economic and Social Council, 1994). Supplementary material on the history of the development of the Fundamental Principles and their history is available on the website of the Statistics Division.

USUAL PRACTICE: THEORY VS REALITY



Comment 1

USUAL PRACTICE: THEORY VS REALITY



Comment 2

USUAL PRACTICE: THEORY VS REALITY



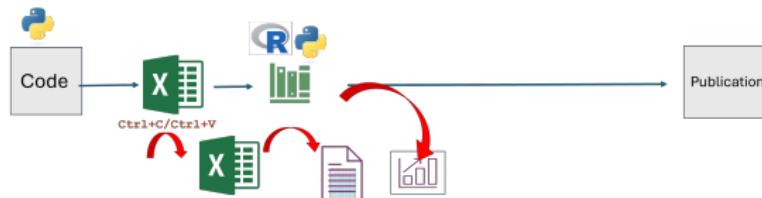
Comment 3

USUAL PRACTICE: THEORY VS REALITY



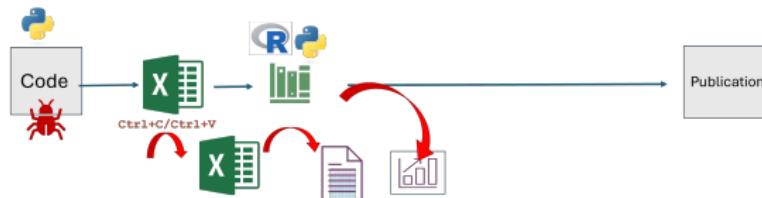
Comment 4

USUAL PRACTICE: THEORY VS REALITY



Comment 5

USUAL PRACTICE: THEORY VS REALITY



Comment 6

USUAL PRACTICE: THEORY VS REALITY



Comment 7

USUAL PRACTICE: THEORY VS REALITY



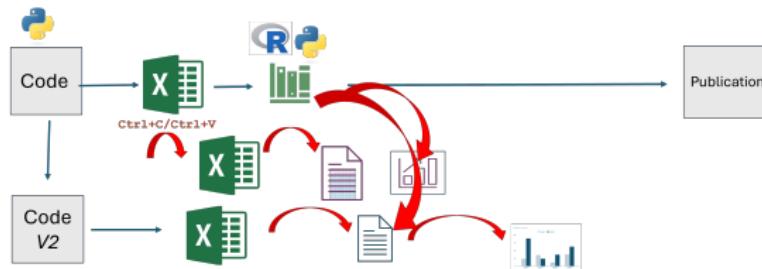
Comment 8

USUAL PRACTICE: THEORY VS REALITY



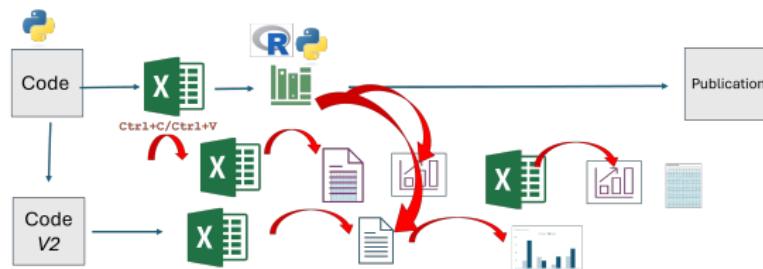
Comment 9

USUAL PRACTICE: THEORY VS REALITY



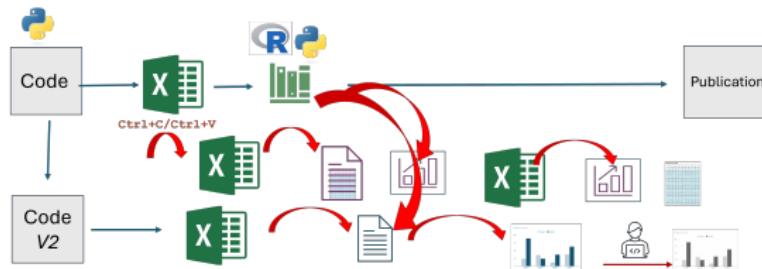
Comment 10

USUAL PRACTICE: THEORY VS REALITY



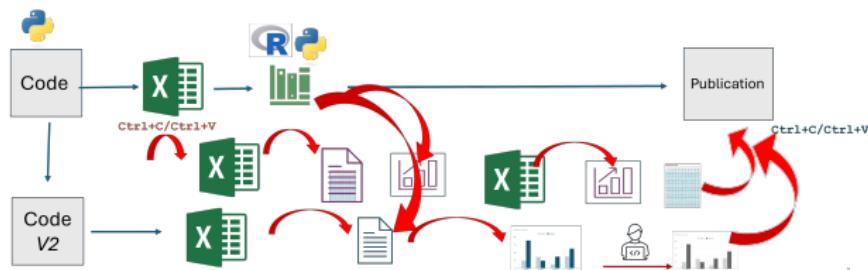
Comment 9

USUAL PRACTICE: THEORY VS REALITY



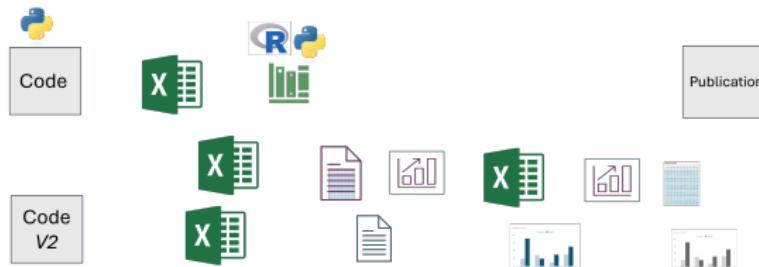
Comment 10

USUAL PRACTICE: THEORY VS REALITY



Comment 9

USUAL PRACTICE: THEORY VS REALITY



Comment 10

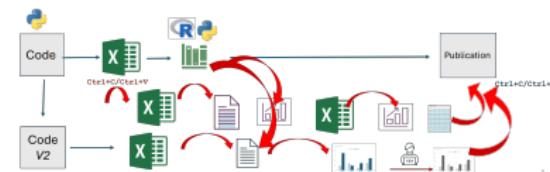
USUAL PRACTICE: IN THE END

- ▶ Lots of files



USUAL PRACTICE: IN THE END

- ▶ Lots of files
- ▶ Cut and paste is not a reliable, reproducible approach!



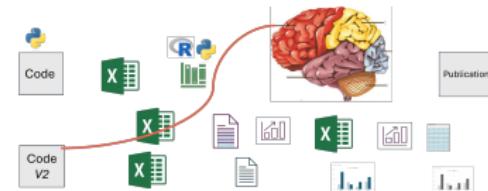
USUAL PRACTICE: IN THE END

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



USUAL PRACTICE: IN THE END

- ▶ Lots of files
- ▶ Cut and paste is not a reliable, reproducible approach!
- ▶ Your brain may remember..
...all the steps...



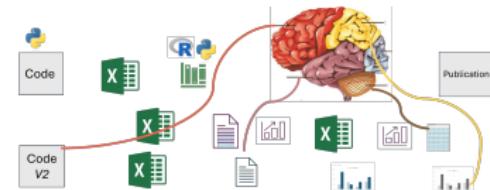
USUAL PRACTICE: IN THE END

- ▶ Lots of files
- ▶ Cut and paste is not a reliable, reproducible approach!
- ▶ Your brain may remember..
...all the steps...
.. in the right order..



USUAL PRACTICE: IN THE END

- ▶ Lots of files
- ▶ Cut and paste is not a reliable, reproducible approach!
- ▶ Your brain may remember..
...all the steps...
.. in the right order..
...all of them !



USUAL PRACTICE: IN THE END

- ▶ Lots of files
- ▶ Cut and paste is not a reliable, reproducible approach!
- ▶ Your brain may remember..
...all the steps...
.. in the right order..
...all of them !
- ▶ Or use (bad) "tools"



WHAT ARE THE ISSUES?

- ▶ Errors due to cut and paste

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

By Leo Kelton
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.

WHAT ARE THE ISSUES?

- ▶ Errors due to cut and paste
- ▶ Errors are difficult to track

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

By Leo Kelton
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.

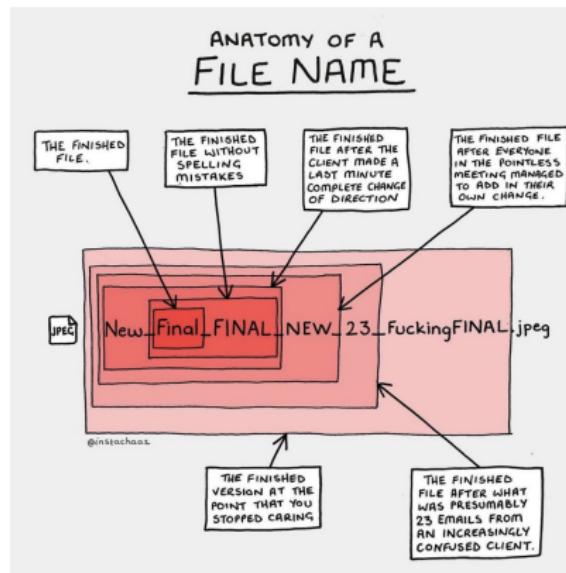
WHAT ARE THE ISSUES?

- ▶ Errors due to cut and paste
- ▶ Errors are difficult to track
- ▶ Each operator has his/her own approach



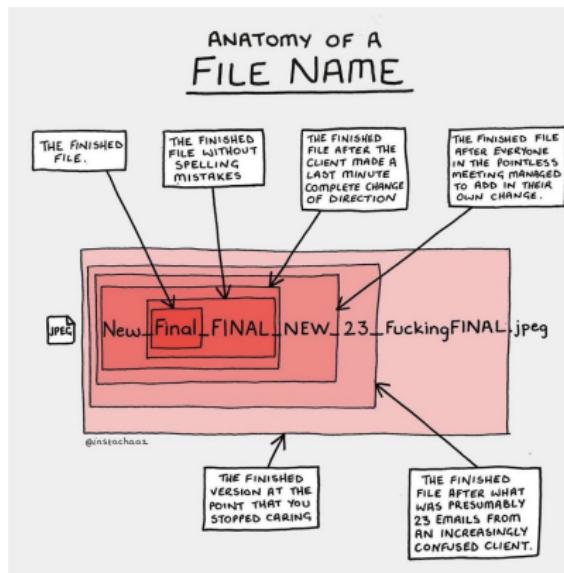
WHAT ARE THE ISSUES?

- ▶ Errors due to cut and paste
- ▶ Errors are difficult to track
- ▶ Each operator has his/her own approach
- ▶ Several versions of code may coexist



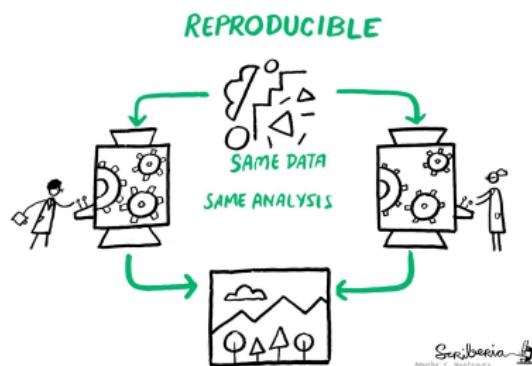
WHAT ARE THE ISSUES?

- ▶ 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



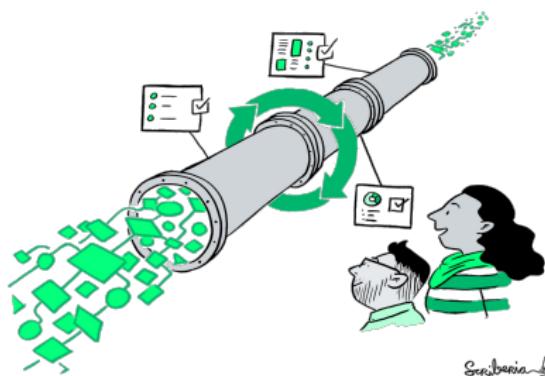
WHAT ARE THE ISSUES?

- ▶ 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



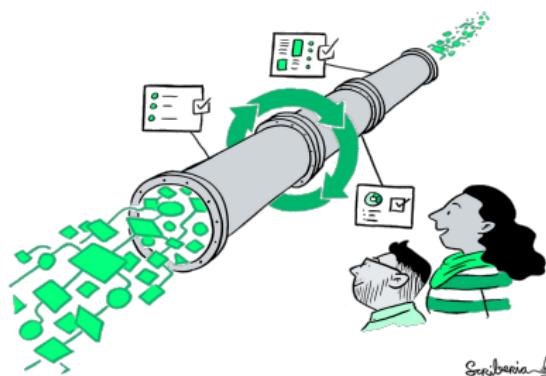
WHAT IS A REPRODUCIBLE ANALYTICAL PIPELINE?

- ▶ It is a process



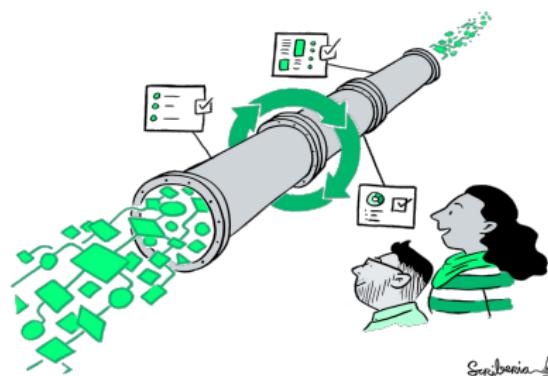
WHAT IS A REPRODUCIBLE ANALYTICAL PIPELINE?

- ▶ It is a process
- ▶ It is automated



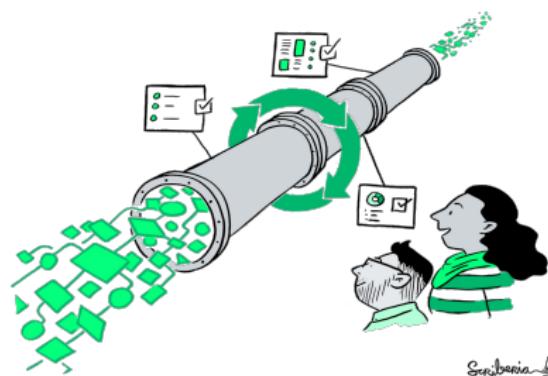
WHAT IS A REPRODUCIBLE ANALYTICAL PIPELINE?

- ▶ It is a process
- ▶ It is automated
- ▶ It is easily reproducible



WHAT IS A REPRODUCIBLE ANALYTICAL PIPELINE?

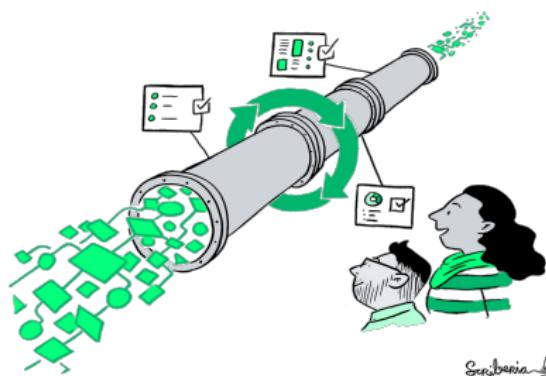
- ▶ It is a process
- ▶ It is automated
- ▶ It is easily reproducible
- ▶ It minimises mistakes



Serbiania

WHAT IS A REPRODUCIBLE ANALYTICAL PIPELINE?

- ▶ It is a process
- ▶ It is automated
- ▶ It is easily reproducible
- ▶ It minimises mistakes
- ▶ It is fast



Scriberia

WHAT IS A REPRODUCIBLE ANALYTICAL PIPELINE?

- ▶ It is a process
- ▶ It is automated
- ▶ It is easily reproducible
- ▶ It minimises mistakes
- ▶ It is fast
- ▶ It builds trust



Scriberia

WHAT DOES A RAP LOOK LIKE?



C

Comment 1

WHAT DOES A RAP LOOK LIKE?



C

Comment 2

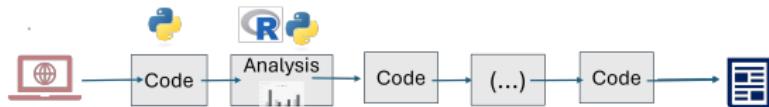
WHAT DOES A RAP LOOK LIKE?



C

Comment 3

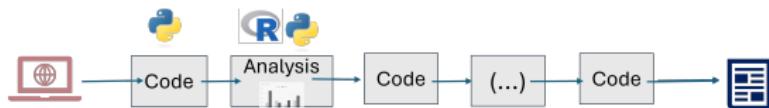
WHAT DOES A RAP LOOK LIKE?



c

Comment 4

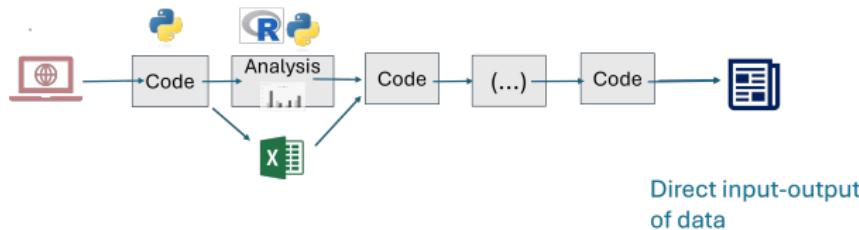
WHAT DOES A RAP LOOK LIKE?



C

Comment 5

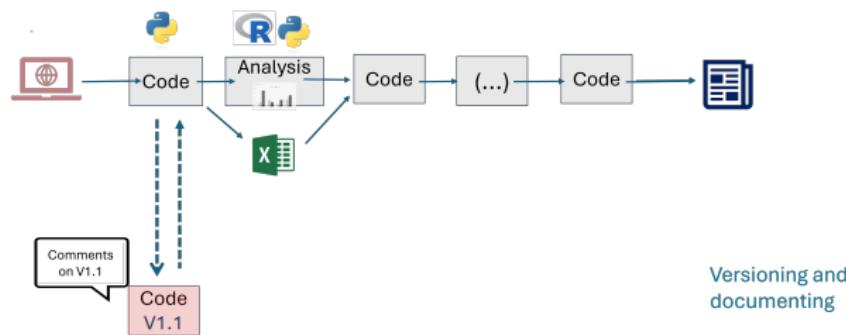
WHAT DOES A RAP LOOK LIKE?



C

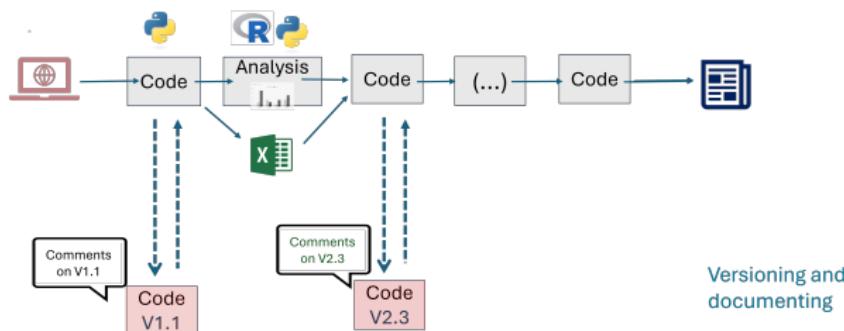
Comment 6

WHAT DOES A RAP LOOK LIKE?



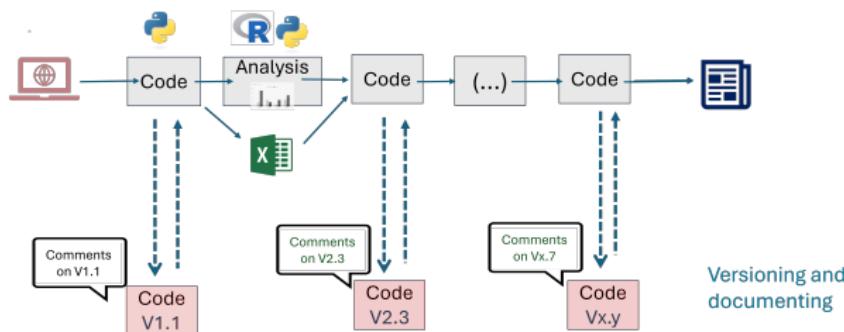
Comment 7

WHAT DOES A RAP LOOK LIKE?



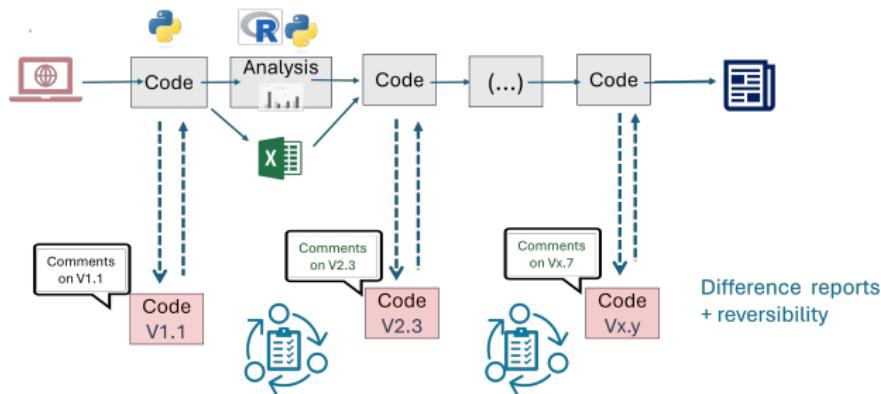
Comment 8

WHAT DOES A RAP LOOK LIKE?



Comment 9

WHAT DOES A RAP LOOK LIKE?



C

Comment 10

Motivation
oooo

Issues
o

RAP
oo

3 Principles
●oooooooooooo

Version Control
oooooooooooo

Resources
o

3 MAIN PRINCIPLES:

1. Organize your work

Motivation
oooo

Issues
o

RAP
oo

3 Principles
●oooooooooooo

Version Control
oooooooooooo

Resources
o

3 MAIN PRINCIPLES:

1. Organize your work
2. Code for others (including your future self)

Motivation
oooo

Issues
o

RAP
oo

3 Principles
●oooooooooooo

Version Control
oooooooooooo

Resources
o

3 MAIN PRINCIPLES:

1. Organize your work
2. Code for others (including your future self)
3. DRY: Do **not** Repeat Yourself

Motivation
oooo

Issues
o

RAP
oo

3 Principles
●oooooooooooo

Version Control
oooooooooooo

Resources
o

3 MAIN PRINCIPLES:

1. Organize your work
2. Code for others (including your future self)
3. DRY: Do **not** Repeat Yourself

3 MAIN PRINCIPLES:

1. Organize your work
2. Code for others (including your future self)
3. DRY: Do **not** Repeat Yourself

Apply this in context (colleagues, code, software,...)

Motivation
oooo

Issues
o

RAP
oo

3 Principles
o●oooooooooooo

Version Control
oooooooooooo

Resources
o

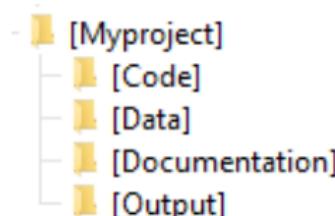
ORGANIZE YOUR WORK

Have a clear directory structure

ORGANIZE YOUR WORK

Have a clear directory structure

- ▶ Separate files into data, code, docs, etc.

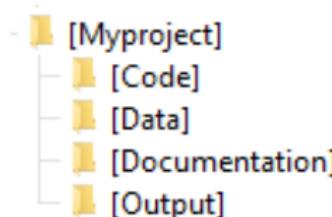


Example of a well-organized directory structure.

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.

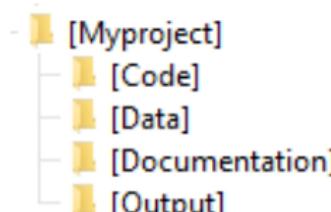
Usual

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

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")
```

ORGANIZE YOUR WORK

Use naming conventions: For files/code

- ▶ Avoid lazy names

Usual

prog1.ipynb
prog2.ipynb
Stat.ipynb
progC.ipynb
progP.ipynb

ORGANIZE YOUR WORK

Use naming conventions: For files/code

- ▶ Avoid lazy names
- ▶ Meaningful files names

Usual	Better
prog1.ipynb	Scraping_Data.ipynb
prog2.ipynb	Cleaning_Data.ipynb
Stat.ipynb	Stats_Tables.ipynb
progC.ipynb	Classification.ipynb
progP.ipynb	Price_CPI.ipynb

ORGANIZE YOUR WORK

Use naming conventions: For files/code

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

Usual	Even better
prog1.ipynb	01_Scraping_data.ipynb
prog2.ipynb	02_Cleaning_data.ipynb
Stat.ipynb	03_Classification.ipynb
progC.ipynb	04_Stats_Tables.ipynb
progP.ipynb	04_Price_CPI.ipynb

ORGANIZE YOUR WORK

Use naming conventions:
For outputs

- ▶ Avoid numbering
- Usual
 - Table1.pdf
 - Table2.pdf
 - Graph.jpg
 - Model.csv

ORGANIZE YOUR WORK

Use naming conventions: For outputs

- ▶ Avoid numbering
- ▶ Explicit type of output

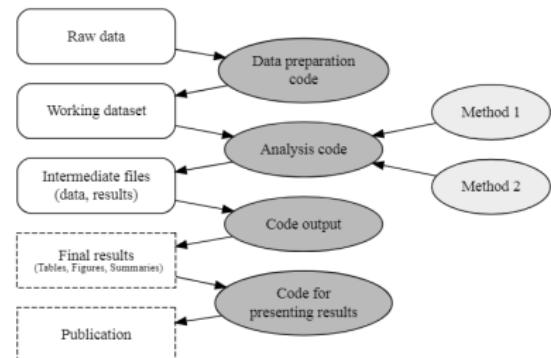
Usual
Table1.pdf
Table2.pdf
Graph.jpg
Model.csv

Better
Stat_Desc_Table.pdf
Price_Stat_Table.pdf
Dress_Prices_Graphic.jpg
All_prices_Results.csv

ORGANIZE YOUR WORK

Keep track of the workflow:

- ▶ Cut and paste should be avoided

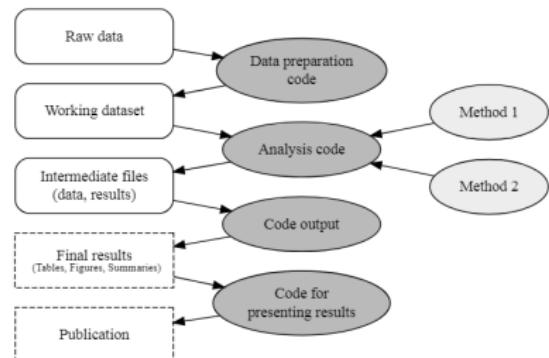


Example of a simple workflow.

ORGANIZE YOUR WORK

Keep track of the workflow:

- ▶ Cut and paste should be avoided
- ▶ Every step of the process is coded

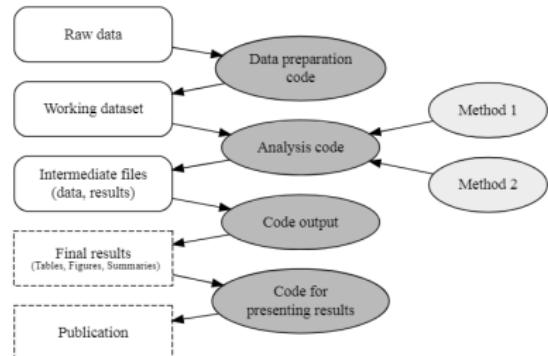


Example of a simple workflow.

ORGANIZE YOUR WORK

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.

ORGANIZE YOUR WORK

Use a version control system (Git/GitHub)



More on Version Control later

CODE FOR OTHERS (INCLUDING YOUR "future self")

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 FOR OTHERS (INCLUDING YOUR "future self")

Program with style:

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

Guido van Rossum (2013 -PEP8)

PEP stands for *Python Enhancement Proposals*

CODE FOR OTHERS (INCLUDING YOUR "future self")

Program with style:

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
 - Inline Comments
 - Documentation Strings
- Naming Conventions
 - Overriding Principle
 - Descriptive: Naming Styles
 - Prescriptive: Naming Conventions
 - Names to Avoid
 - ASCII Compatibility
 - Package and Module Names
 - Class Names
 - Type Variable Names
 - Exception Names
 - Global Variable Names
 - Function and Variable Names
 - Function and Method Arguments
 - Method Names and Instance Variables
 - Constants
 - Designing for Inheritance
 - Public and Internal Interfaces
 - Preserving Recommendations
 - Future-proof Annotations

PEP 8 – Style Guide for Python Code

Author: Guido van Rossum <guido at python.org>, Barry Warsaw <barry at python.org>, Alyssa Coghlan <cohoglan 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.

This document and [PEP 257](#) (Docstring Conventions) were adapted from Guido's original Python Style Guide essay, with some additions from Barry's style guide [2].

This style guide evolves over time as additional conventions are identified and past conventions are rendered obsolete by changes in the language itself.

Many projects have their own coding style guidelines. In the event of any conflicts, such project-specific guides take precedence for that project.

A Foolish Consistency is the Hobgoblin of Little Minds

One of Guido's key insights is that code is read much more often than it is written. The guidelines provided here are intended to improve the readability of code and make it consistent across the wide spectrum of Python code. As PEP 20 says, "Readability counts".

A style guide is about consistency. Consistency with this style guide is important. Consistency within a project is more important. Consistency within one module or function is the most important.

However, know when to be inconsistent – sometimes style guide recommendations just aren't applicable. When you run into one of those situations, look at other examples and decide what other folks do and do that. That's ask!

CODE FOR OTHERS

Program with style

- ▶ Avoid ambiguities

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)
```

CODE FOR OTHERS

Program with style

- ▶ Avoid ambiguities
- ▶ Avoid changing units

Usual

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

CODE FOR OTHERS

Program with style

- ▶ Avoid ambiguities
- ▶ Avoid changing units

Usual

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

Better

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

CODE FOR OTHERS

Program with style

- ▶ Avoid ambiguities
- ▶ Avoid changing units

Usual

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

Even better

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

DO NOT REPEAT YOURSELF

Create reusable objects

- ▶ Store values

Usual

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

DO NOT REPEAT YOURSELF

Create reusable objects

- ▶ Store values

Usual

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

- Avoid repetitions

Better

```
Current_year = 2023
```

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

DO NOT REPEAT YOURSELF

Create reusable objects

► Store

values

Avoid

repetitions

Usual

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

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

...

DO NOT REPEAT YOURSELF

Create reusable objects

- ▶ Store values
- ▶ Avoid repetitions
- ▶ Use functions

Better

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

```
plot_export("Beef")  
plot_export("Kava")
```

OTHER PRINCIPLES

- ▶ Discuss with colleagues that may use your work

OTHER PRINCIPLES

- ▶ Discuss with colleagues that may use your work
- ▶ Automatize as much as you can

OTHER PRINCIPLES

- ▶ Discuss with colleagues that may use your work
- ▶ Automatize as much as you can
- ↪ Reduces your brain's memory burden

OTHER PRINCIPLES

- ▶ Discuss with colleagues that may use your work
- ▶ Automatize as much as you can
- ↪ Reduces your brain's memory burden
- ▶ There are easy steps everybody can do

OTHER PRINCIPLES

- ▶ Discuss with colleagues that may use your work
- ▶ Automatize as much as you can
- ↪ Reduces your brain's memory burden
- ▶ There are easy steps everybody can do
- ↪ Write small programs, one for each task

OTHER PRINCIPLES

- ▶ Discuss with colleagues that may use your work
- ▶ Automatize as much as you can
- Reduces your brain's memory burden
- ▶ There are easy steps everybody can do
- Write small programs, one for each task
- ▶ Use open source program

OTHER PRINCIPLES

- ▶ Discuss with colleagues that may use your work
- ▶ Automatize as much as you can
- ↪ Reduces your brain's memory burden
- ▶ There are easy steps everybody can do
- ↪ Write small programs, one for each task
- ▶ Use open source program
- ↪ Easier to share, easier to automatize

OTHER PRINCIPLES

- ▶ Discuss with colleagues that may use your work
- ▶ Automatize as much as you can
- ↪ Reduces your brain's memory burden
- ▶ There are easy steps everybody can do
- ↪ Write small programs, one for each task
- ▶ Use open source program
- ↪ Easier to share, easier to automatize
- ↪ Also cost-effective

OTHER PRINCIPLES

- ▶ Discuss with colleagues that may use your work
- ▶ Automatize as much as you can
- ↪ Reduces your brain's memory burden
- ▶ There are easy steps everybody can do
- ↪ Write small programs, one for each task
- ▶ Use open source program
- ↪ Easier to share, easier to automatize
- ↪ Also cost-effective
- ▶ Test your work regularly:

OTHER PRINCIPLES

- ▶ Discuss with colleagues that may use your work
- ▶ Automatize as much as you can
- ↪ Reduces your brain's memory burden
- ▶ There are easy steps everybody can do
- ↪ Write small programs, one for each task
- ▶ Use open source program
- ↪ Easier to share, easier to automatize
- ↪ Also cost-effective
- ▶ Test your work regularly:

OTHER PRINCIPLES

- ▶ Discuss with colleagues that may use your work
- ▶ Automatize as much as you can
- ↪ Reduces your brain's memory burden
- ▶ There are easy steps everybody can do
- ↪ Write small programs, one for each task
- ▶ Use open source program
- ↪ Easier to share, easier to automatize
- ↪ Also cost-effective
- ▶ Test your work regularly:

“Do what has been said, say what has been done, and check that what has been said has really been done !”

OTHER PRINCIPLES

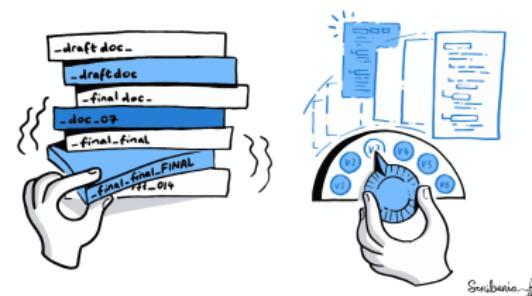
- ▶ Discuss with colleagues that may use your work
- ▶ Automatize as much as you can
- ↪ Reduces your brain's memory burden
- ▶ There are easy steps everybody can do
- ↪ Write small programs, one for each task
- ▶ Use open source program
- ↪ Easier to share, easier to automatize
- ↪ Also cost-effective
- ▶ Test your work regularly:

"Code what has been said, say what has been coded, and check that what has been said has really been coded!"

VERSION CONTROL KEEPS TRACKS OF YOUR WORK

Tracking three W questions:

What changes?



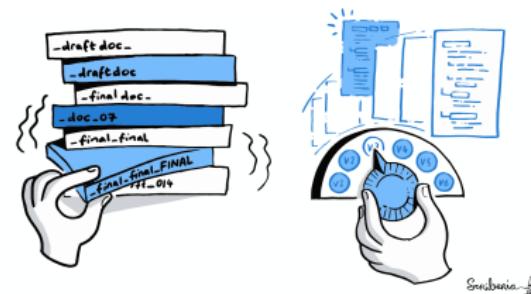
Source: The Turing Way project

VERSION CONTROL KEEPS TRACKS OF YOUR WORK

Tracking three W questions:

What changes?

Who made the changes?



Source: The Turing Way project

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

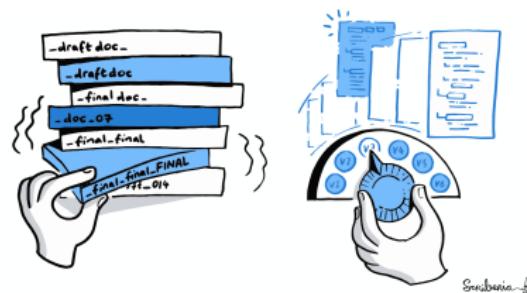
VERSION CONTROL KEEPS TRACKS OF YOUR WORK

Tracking three W questions:

What changes?

Who made the changes?

When were the changes made?



Source: The Turing Way project

Motivation
oooo

Issues
o

RAP
oo

3 Principles
oooooooooooo

Version Control
o●oooooooooooo

Resources
o

TRANSPARENCY, ACCOUNTABILITY & REPRODUCIBILITY

- ▶ Version control provides a detailed history of changes

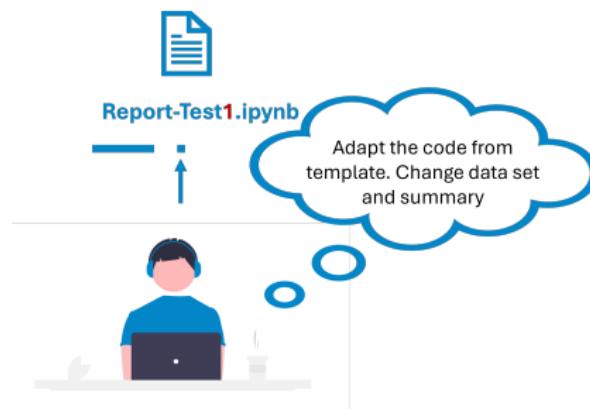
TRANSPARENCY, ACCOUNTABILITY & REPRODUCIBILITY

- ▶ Version control provides a detailed history of changes
- ▶ Each modification is attributed to a specific user

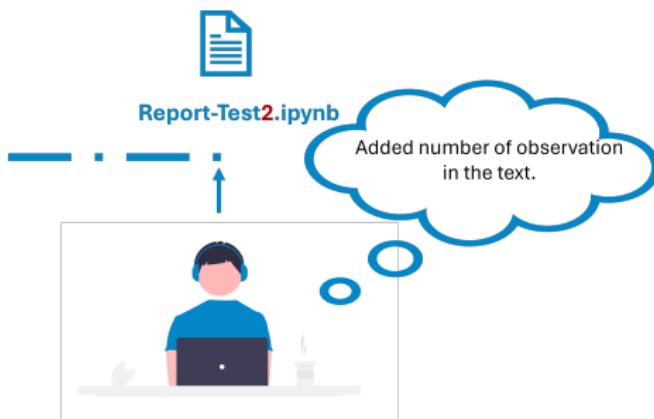
TRANSPARENCY, ACCOUNTABILITY & REPRODUCIBILITY

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

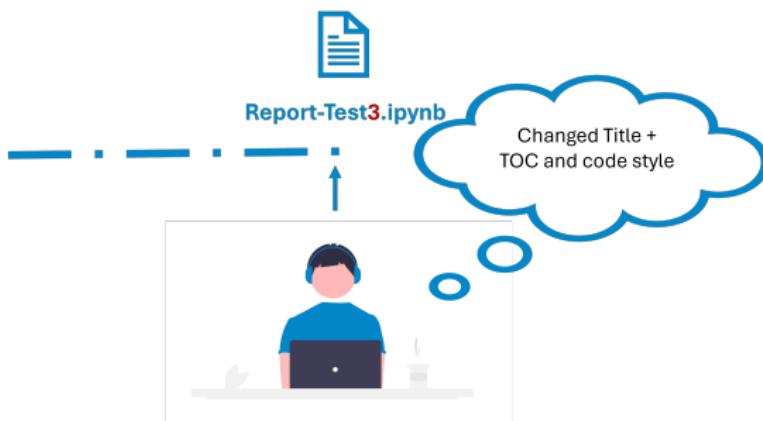
THE EVOLUTION OF A FILE <-XXXX TO REVISE THIS



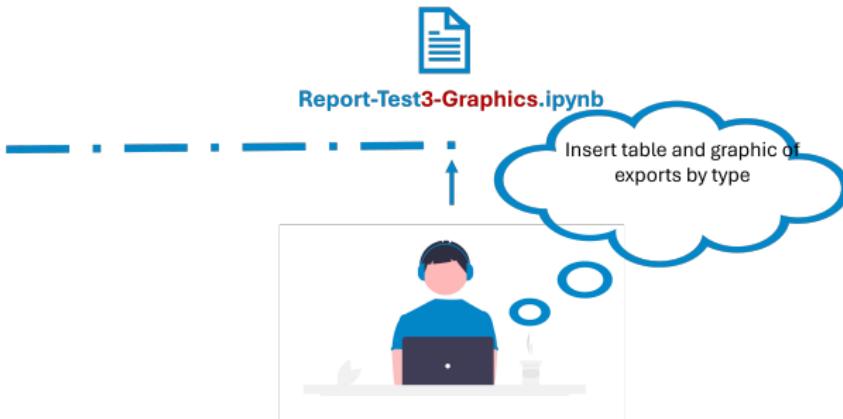
THE EVOLUTION OF A FILE <-XXXX TO REVISE THIS



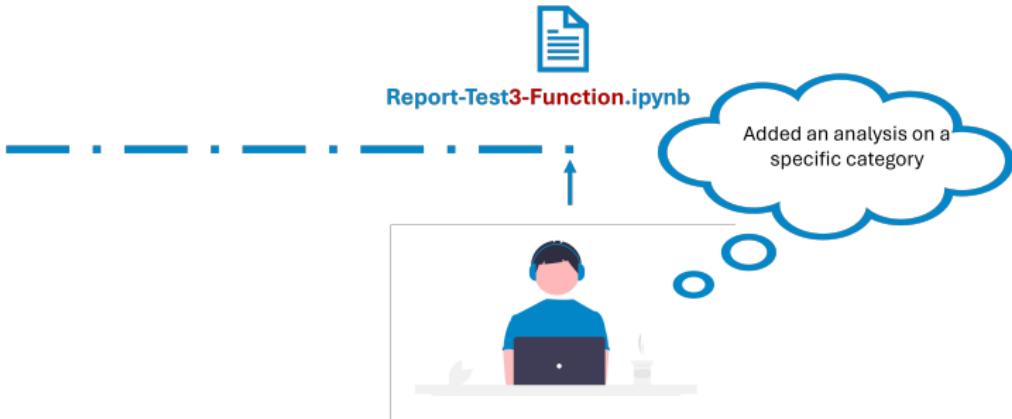
THE EVOLUTION OF A FILE <-XXXX TO REVISE THIS



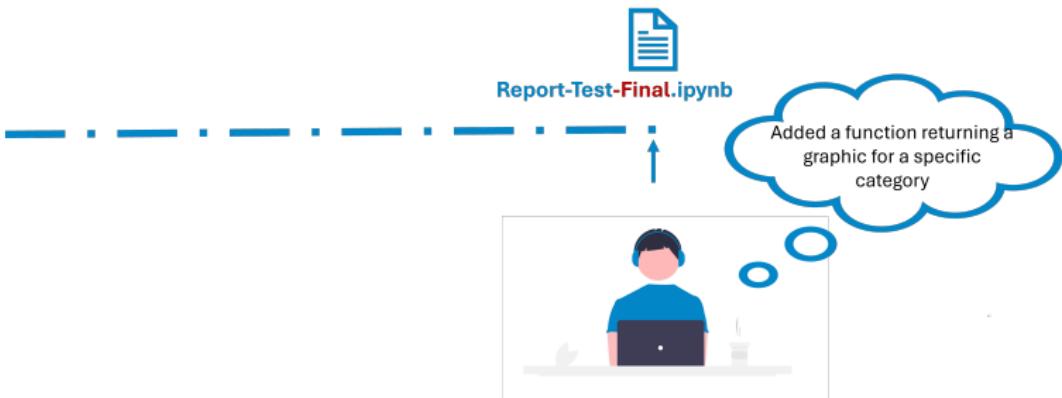
THE EVOLUTION OF A FILE <-XXXX TO REVISE THIS



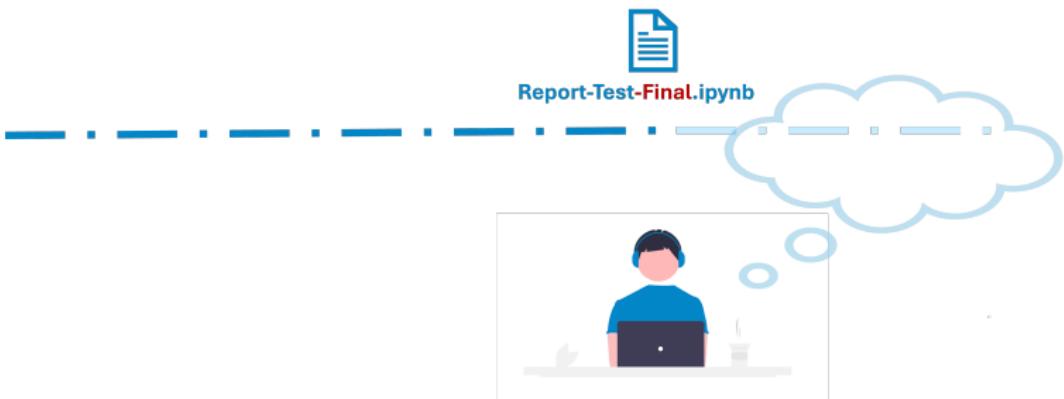
THE EVOLUTION OF A FILE <-XXXX TO REVISE THIS



THE EVOLUTION OF A FILE <-XXXX TO REVISE THIS



THE EVOLUTION OF A FILE <-XXXX TO REVISE THIS



THE EVOLUTION OF A FILE <-XXXX TO REVISE THIS

Usual ways to keep track of changes:

THE EVOLUTION OF A FILE <-XXXX TO REVISE THIS

Usual ways to keep track of changes:

- ▶ New file after each change



[Report-Test1.ipynb](#)



[Report-Test3-Graphics.ipynb](#)



[Report-Test2.ipynb](#)



[Report-Test3- Function.ipynb](#)



[Report-Test3.ipynb](#)



[Report-Test-Final.ipynb](#)

THE EVOLUTION OF A FILE <-XXXX TO REVISE THIS

Usual ways to keep track of changes:

- ▶ New file after each change
- Need to open each file to see the change



[Report-Test1.ipynb](#)



[Report-Test3-Graphics.ipynb](#)



[Report-Test2.ipynb](#)



[Report-Test3-Function.ipynb](#)



[Report-Test3.ipynb](#)



[Report-Test-Final.ipynb](#)

THE EVOLUTION OF A FILE <-XXXX TO REVISE THIS

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



[Report-Test1.ipynb](#)



[Report-Test3-Graphics.ipynb](#)



[Report-Test2.ipynb](#)



[Report-Test3-Function.ipynb](#)



[Report-Test3.ipynb](#)



[Report-Test-Final.ipynb](#)

THE EVOLUTION OF A FILE <-XXXX TO REVISE THIS

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



Report-Test3-Graphics-
Functions-Final-
Chris.ipynb

THE EVOLUTION OF A FILE <-XXXX TO REVISE THIS

Usual ways to keep track of changes:

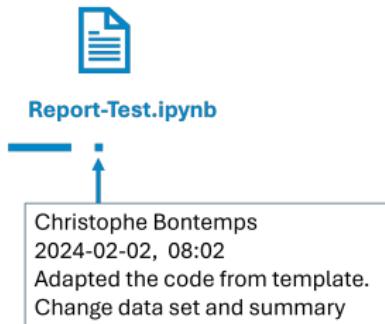
- ▶ New file after each change
- Need to open each file to see the change
- Names have to be explicit
- ▶ Only the last file with lots of comments
- ▶ Not fulfilling the 3 W...



Report-Test3-Graphics-
Functions-Final-
Chris.ipynb

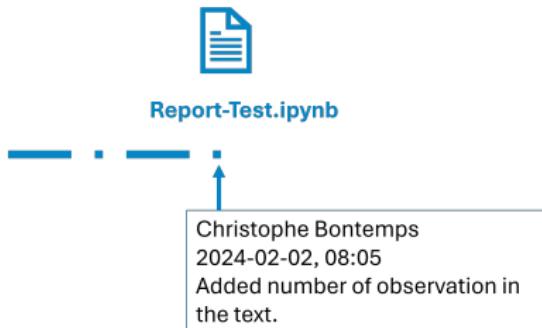
THE EVOLUTION WITH VERSION CONTROL

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



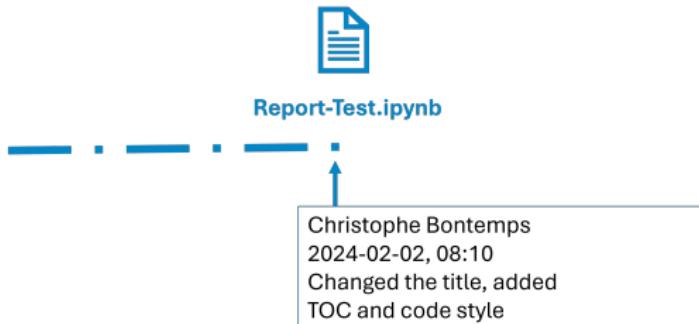
THE EVOLUTION WITH VERSION CONTROL

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



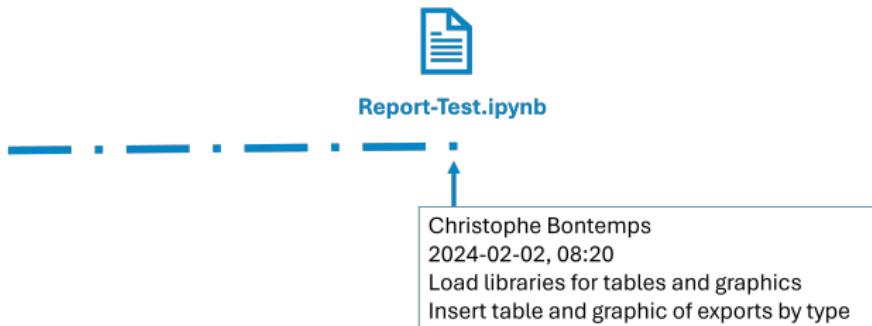
THE EVOLUTION WITH VERSION CONTROL

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



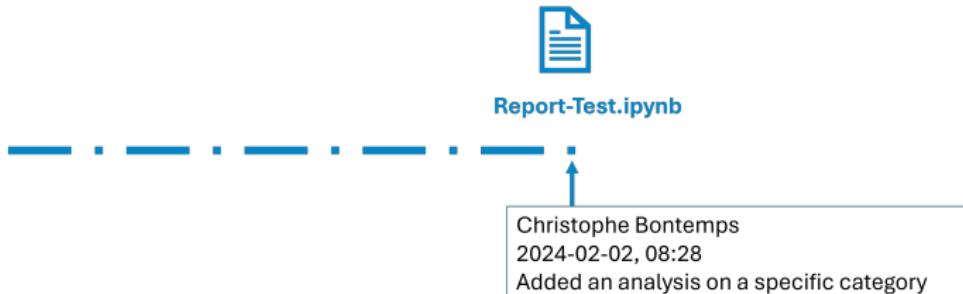
THE EVOLUTION WITH VERSION CONTROL

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



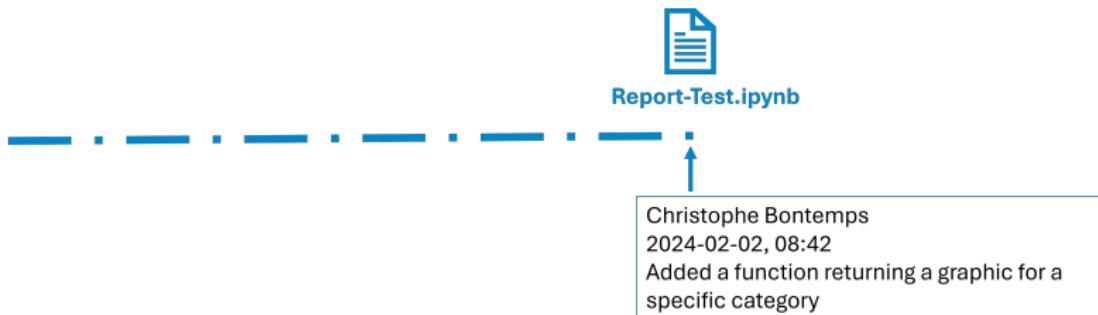
THE EVOLUTION WITH VERSION CONTROL

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



THE EVOLUTION WITH VERSION CONTROL

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



THE EVOLUTION WITH VERSION CONTROL

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



THE HISTORY OF THE FILE IS RECORDED!

Each version is documented (with *commits*)



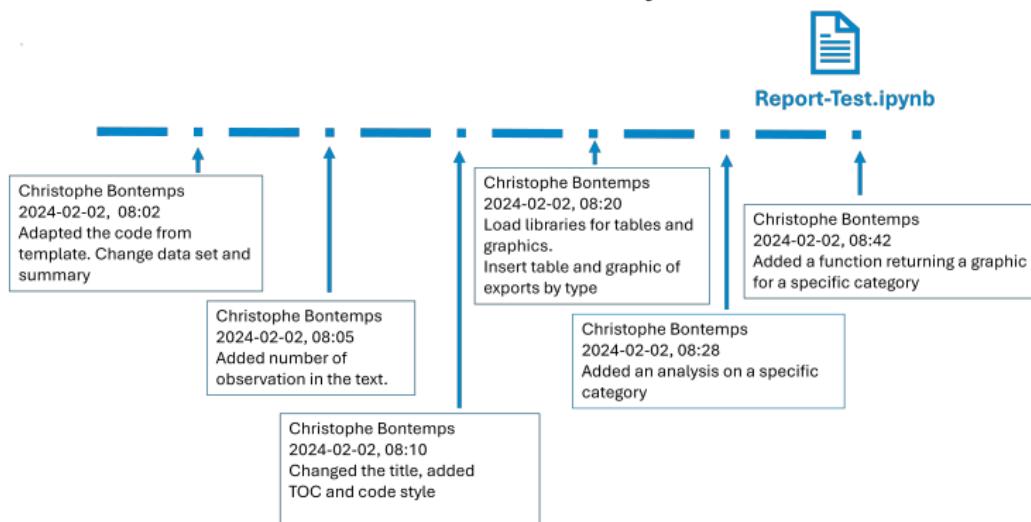
THE HISTORY OF THE FILE IS RECORDED!

Each version is documented (with *commits*)



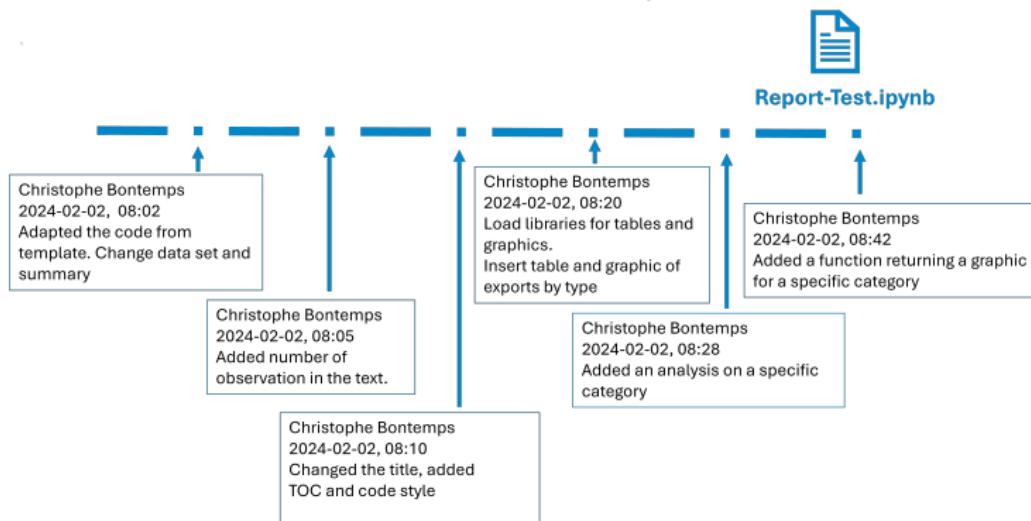
THE HISTORY OF THE FILE IS RECORDED!

Each version embeds the full history!



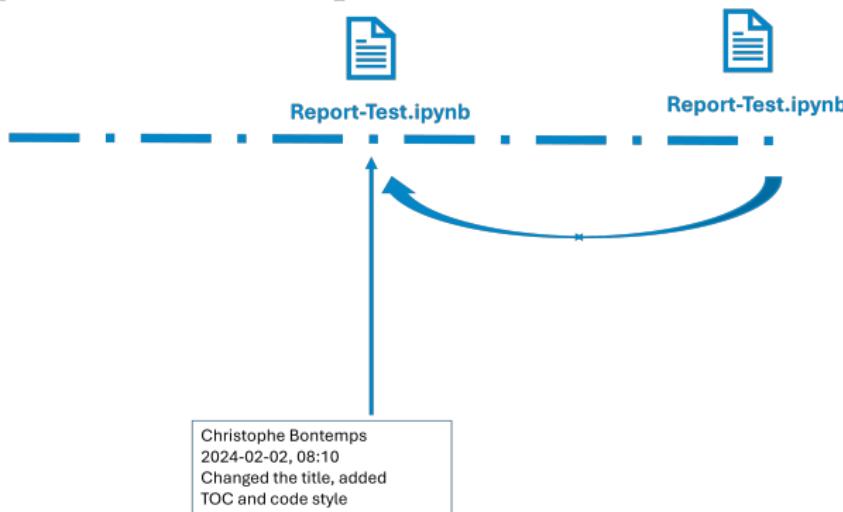
THE HISTORY OF THE FILE IS RECORDED!

Each version embeds the full history!



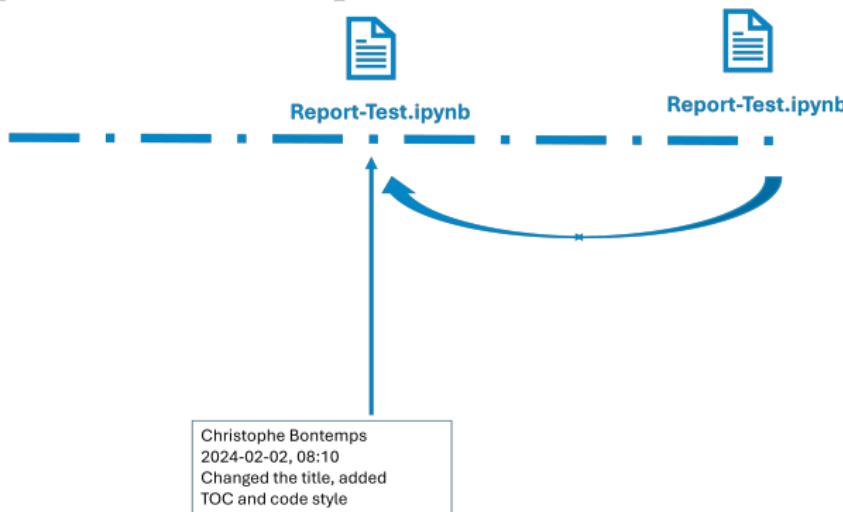
GOING BACK AND "UNDO" IS POSSIBLE

It is possible to review previous version...



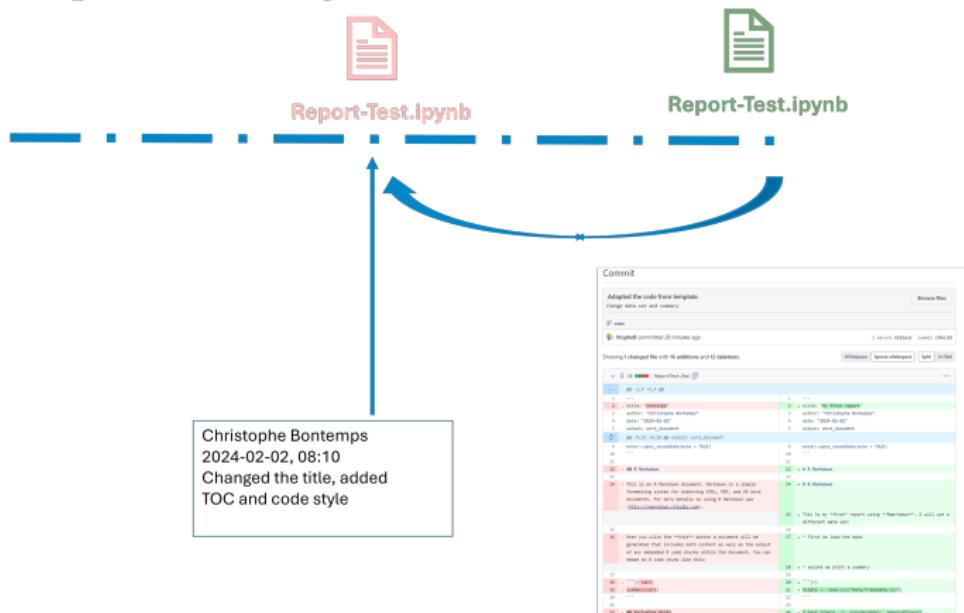
GOING BACK AND "UNDO" IS POSSIBLE

It is possible to review previous version...



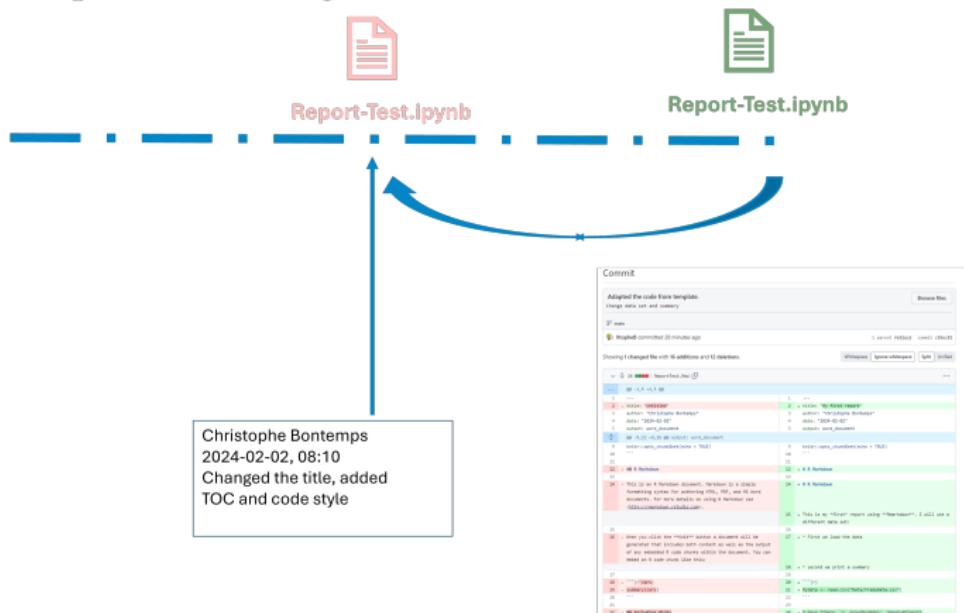
GOING BACK AND "UNDO" IS POSSIBLE

...to compare the changes...



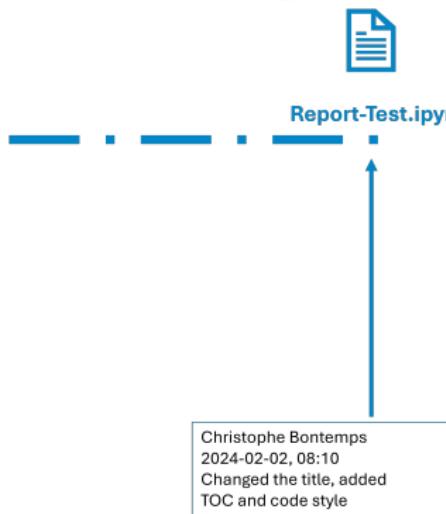
GOING BACK AND "UNDO" IS POSSIBLE

...to compare the changes...



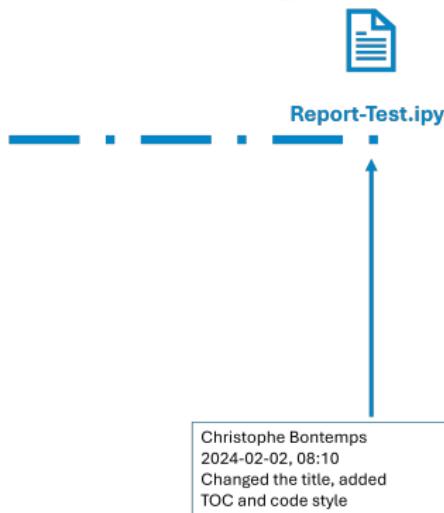
GOING BACK AND "UNDO" IS POSSIBLE

... and to revert to a previous version...



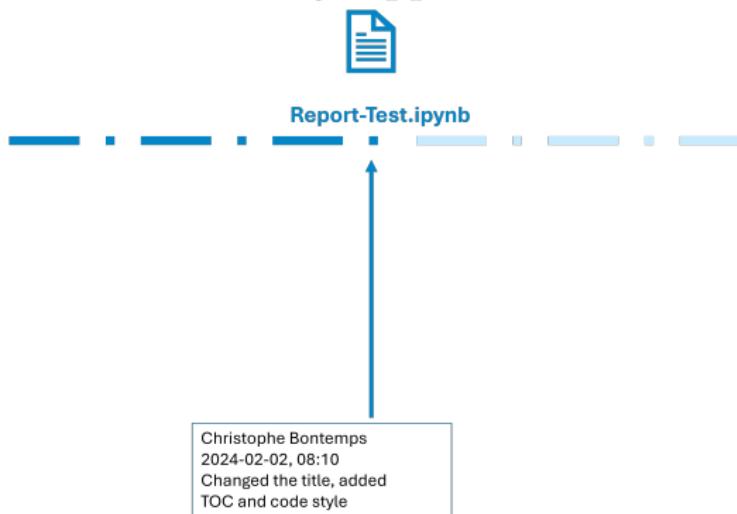
GOING BACK AND "UNDO" IS POSSIBLE

... and to revert to a previous version...



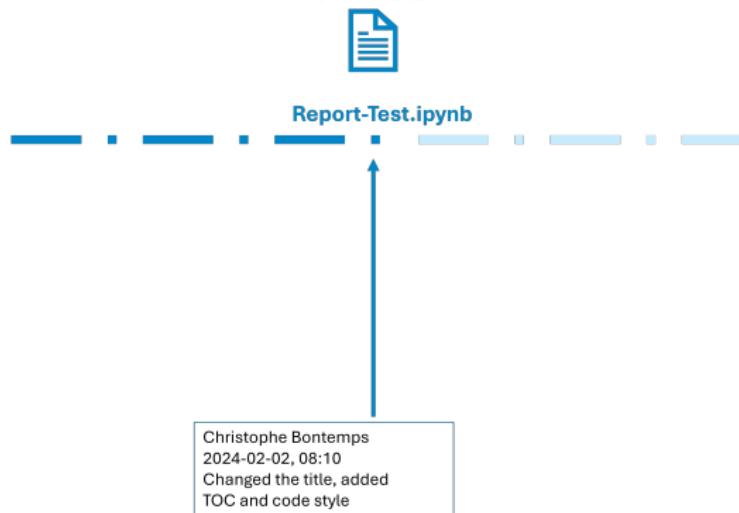
GOING BACK AND "UNDO" IS POSSIBLE

... or *undo* as if nothing happened



GOING BACK AND "UNDO" IS POSSIBLE

... or *undo* as if nothing happened



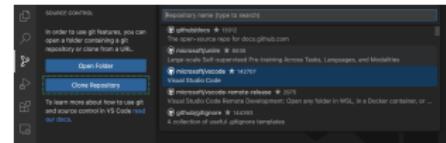
GOOD AND BAD NEWS <-XXXX TO REVISE THIS

Real life is more complex:

GOOD AND BAD NEWS <-XXXX TO REVISE THIS

Real life is more complex:

- + Version Control is integrated in Visual Studio (& RStudio)

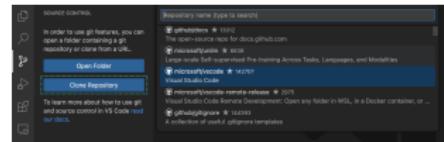


"Visual Studio Documentation"

GOOD AND BAD NEWS <-XXXX TO REVISE THIS

Real life is more complex:

- + Version Control is integrated in Visual Studio (& RStudio)
- Simple operations are easy



"Visual Studio Documentation"

GOOD AND BAD NEWS <-XXXX TO REVISE THIS

Real life is more complex:

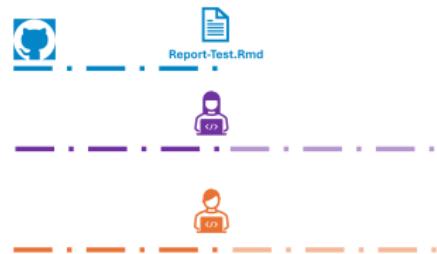
- + Version Control is integrated in Visual Studio (& RStudio)
- Simple operations are easy
- + Collaborate on a project



GOOD AND BAD NEWS <-XXXX TO REVISE THIS

Real life is more complex:

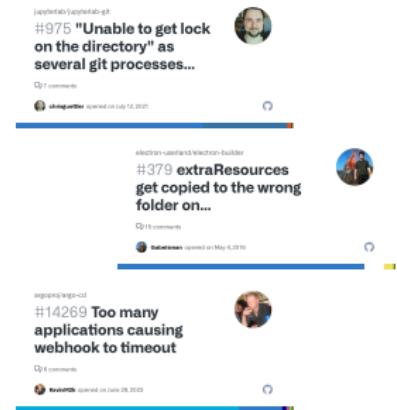
- + Version Control is integrated in Visual Studio (& RStudio)
- Simple operations are easy
- + Collaborate on a project
- Track changes of others



GOOD AND BAD NEWS <-XXXX TO REVISE THIS

Real life is more complex:

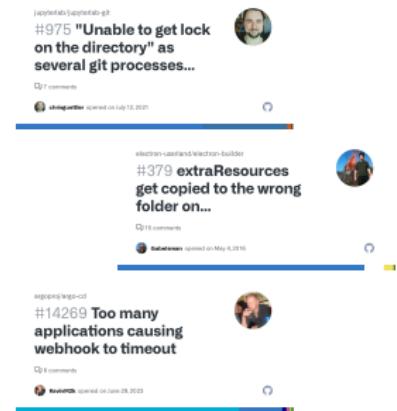
- + Version Control is integrated in Visual Studio (& RStudio)
- Simple operations are easy
- + Collaborate on a project
- Track changes of others
- Git is a bit “*unfriendly*”



GOOD AND BAD NEWS <-XXXX TO REVISE THIS

Real life is more complex:

- + Version Control is integrated in Visual Studio (& RStudio)
- Simple operations are easy
- + Collaborate on a project
- Track changes of others
- Git is a bit "*unfriendly*"
- Complex situations appear easily



GOOD AND BAD NEWS <-XXXX TO REVISE THIS

Real life is more complex:

- + Version Control is integrated in Visual Studio (& RStudio)
 - Simple operations are easy
 - + Collaborate on a project
 - Track changes of others
 - Git is a bit “*unfriendly*”
 - Complex situations appear easily
 - Git works *mostly* in command mode

```
git help git
git clone [repository] [<--help> [<--path>] [<-c name[:value]>]
           [<exec-path>=<path>] [<--http-path>] [<-m man-path>] [<-info-path>]
           [<p paginate>] [<-no-pager>] [<-no-replace-objects>] [<-bare>]
           [<git-dir>] [<--work-tree=>path] [<--tree=>path] [<--namespace=>name]
           [<commands> [<args>]]
```

These are common Git commands used in various situations:

```
status      Working tree status (useful with -u)
clone      Clone a repository into a new directory
init       Create an empty Git repository or reinitialize an existing one
```

Ashish Vishwakarma

VERSION CONTROL IN A NUTSHELL <-XXXX To REVISE THIS

Version control system:

- ▶ Allows to travel back in time



GitHub logo

VERSION CONTROL IN A NUTSHELL <-XXXX To REVISE THIS

Version control system:

- ▶ Allows to travel back in time
- ▶ Keeps track of all changes



GitHub logo

VERSION CONTROL IN A NUTSHELL <-XXXX To REVISE THIS

Version control system:

- ▶ Allows to travel back in time
- ▶ Keeps track of all changes
- ▶ Allows to "undo" at any point



GitHub logo

VERSION CONTROL IN A NUTSHELL <-XXXX To REVISE THIS

Version control system:

- ▶ Allows to travel back in time
- ▶ Keeps track of all changes
- ▶ Allows to "undo" at any point
- ▶ Allows reviewing stages of development



GitHub logo

VERSION CONTROL IN A NUTSHELL <-XXXX To REVISE THIS

Version control system:

- ▶ Allows to travel back in time
- ▶ Keeps track of all changes
- ▶ Allows to "undo" at any point
- ▶ Allows reviewing stages of development
- ▶ Allow collaborating on projects



GitHub logo

VERSION CONTROL IN A NUTSHELL <-XXXX To REVISE THIS

Version control system:

- ▶ Allows to travel back in time
- ▶ Keeps track of all changes
- ▶ Allows to "undo" at any point
- ▶ Allows reviewing stages of development
- ▶ Allow collaborating on projects
- ▶ Backups your work



GitHub logo

Motivation
oooo

Issues
o

RAP
oo

3 Principles
oooooooooooo

Version Control
oooooooooooo●

Resources
o

TAKEAWAYS <-XXXX TO REVISE THIS

TAKEAWAYS <-XXXX TO REVISE THIS

- ▶ Version control is very useful 🎉

TAKEAWAYS <-XXXX TO REVISE THIS

- ▶ Version control is very useful 
- ▶ Version control requires patience, training and experience

TAKEAWAYS <-XXXX TO REVISE THIS

- ▶ Version control is very useful 
- ▶ Version control requires patience, training and experience
- ▶ Version control is essential for transparency, and reproducibility of any project

TAKEAWAYS <-XXXX TO REVISE THIS

- ▶ Version control is very useful 
- ▶ Version control requires patience, training and experience
- ▶ Version control is essential for transparency, and reproducibility of any project
- ▶ Installing Git within RStudio can be tedious but worth it!

TAKEAWAYS <-XXXX TO REVISE THIS

- ▶ Version control is very useful 
- ▶ Version control requires patience, training and experience
- ▶ Version control is essential for transparency, and reproducibility of any project
- ▶ Installing Git within RStudio can be tedious but worth it!
- ▶ There is guidance, tutorials and helping blogs...

USEFUL RESOURCES

- ▶ 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.