Rules of naming in Python and coding

How should I re-name?

PYTHON KA CHILLA

1

Use lowercase letters and underscores:

For both files and folders, it's common to use lowercase letters with words separated by underscores (_). Avoid spaces.

- Example: **data_processing.py**, **data_visualization.py**

2

Be descriptive and specific:

The file and folder names should clearly describe their contents or purpose. Avoid overly generic names.

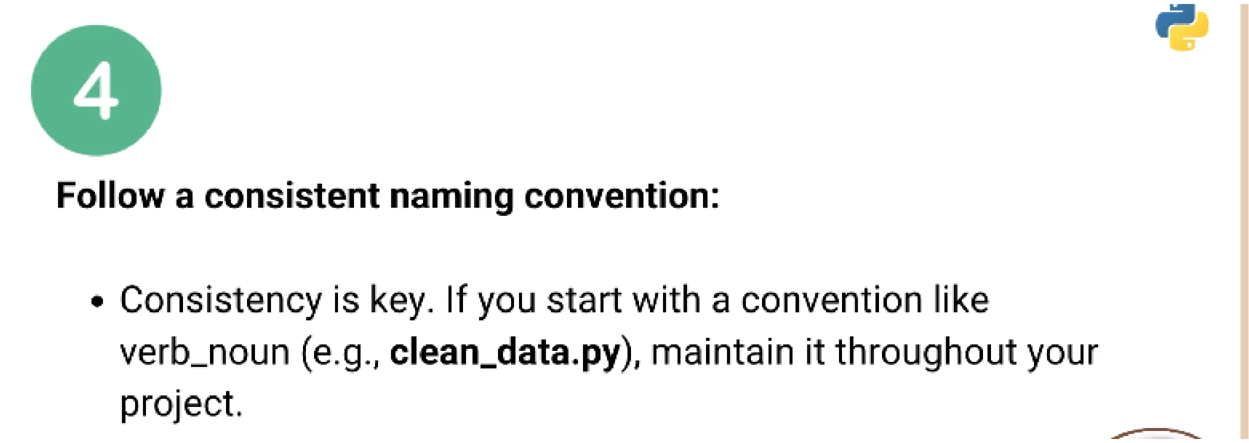
- Example: Instead of **script.py**, use **linear_regression_model.py**

3

Use folders to group related files:

Group files logically into folders based on their functionality or stage in the data science pipeline.

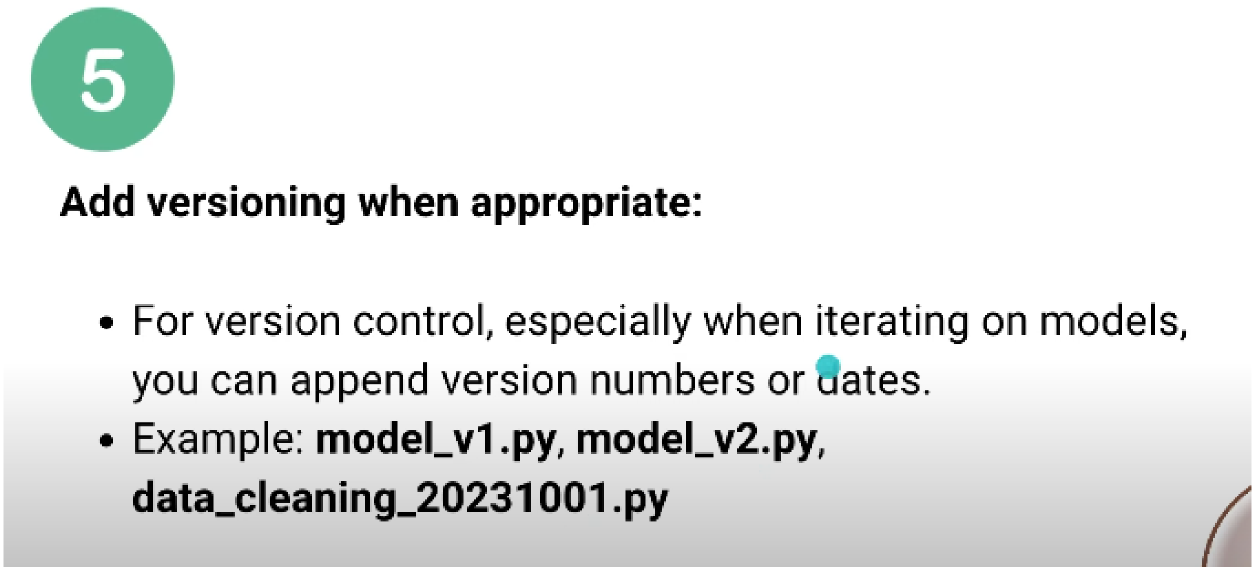
- Example: **data/**, **scripts/**, **models/**, **notebooks/**



4

Follow a consistent naming convention:

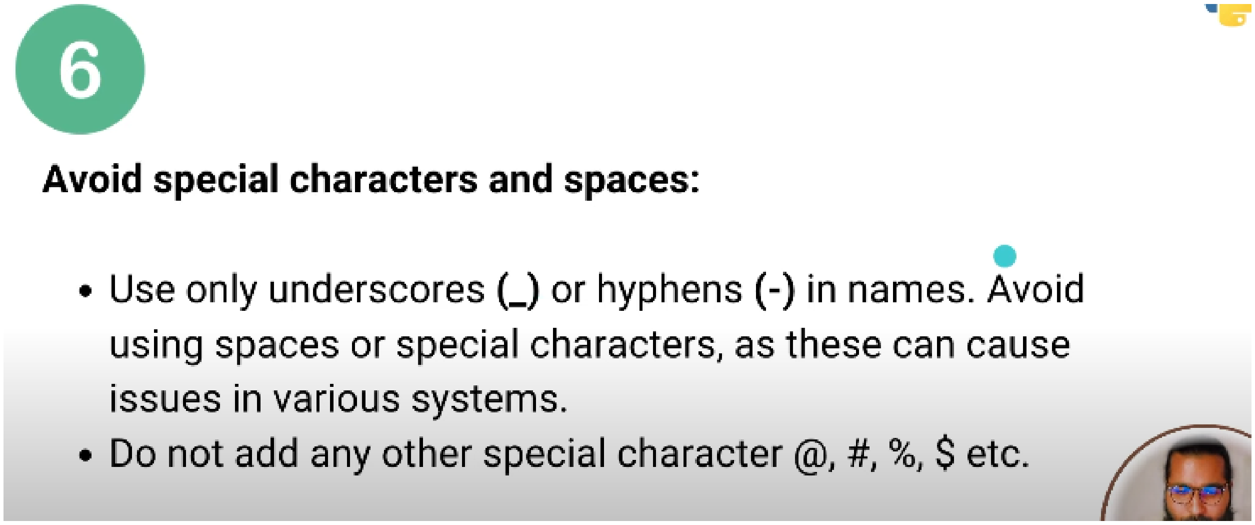
- Consistency is key. If you start with a convention like verb_noun (e.g., **clean_data.py**), maintain it throughout your project.



5

Add versioning when appropriate:

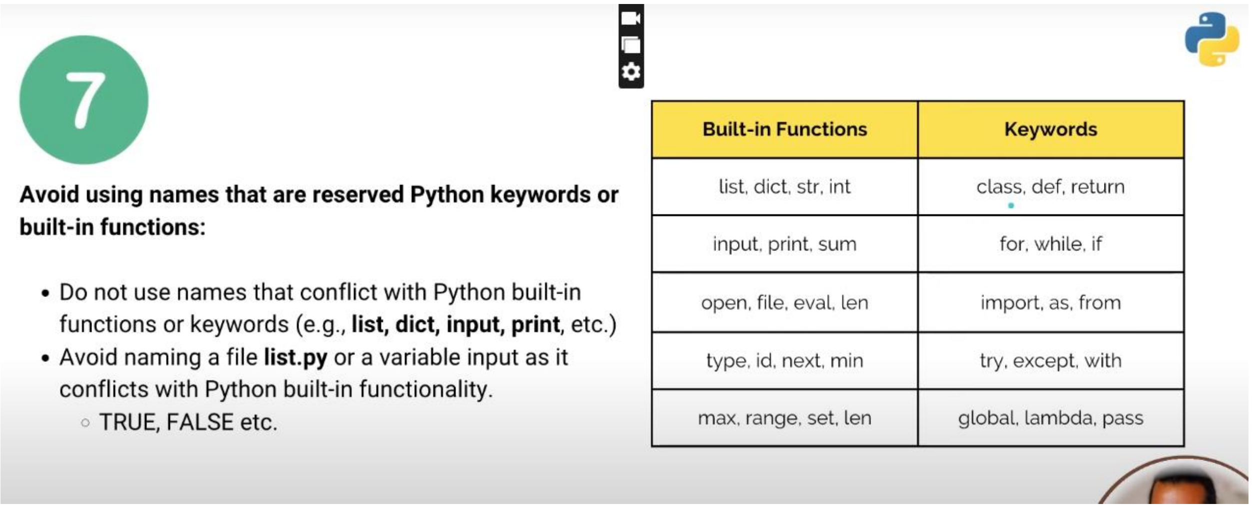
- For version control, especially when iterating on models, you can append version numbers or dates.
- Example: **model_v1.py**, **model_v2.py**, **data_cleaning_20231001.py**



6

Avoid special characters and spaces:

- Use only underscores **(_)** or hyphens **(-)** in names. Avoid using spaces or special characters, as these can cause issues in various systems.
- Do not add any other special character @, #, %, $ etc.

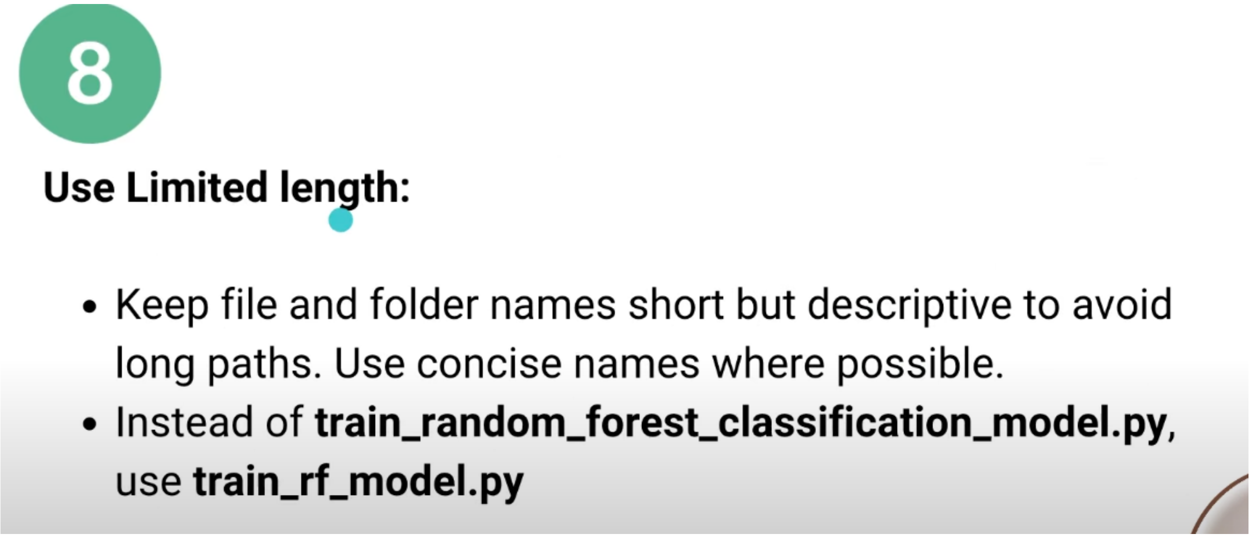


7

Avoid using names that are reserved Python keywords or built-in functions:

- Do not use names that conflict with Python built-in functions or keywords (e.g., **list, dict, input, print**, etc.)
- Avoid naming a file **list.py** or a variable input as it conflicts with Python built-in functionality.
    - TRUE, FALSE etc.

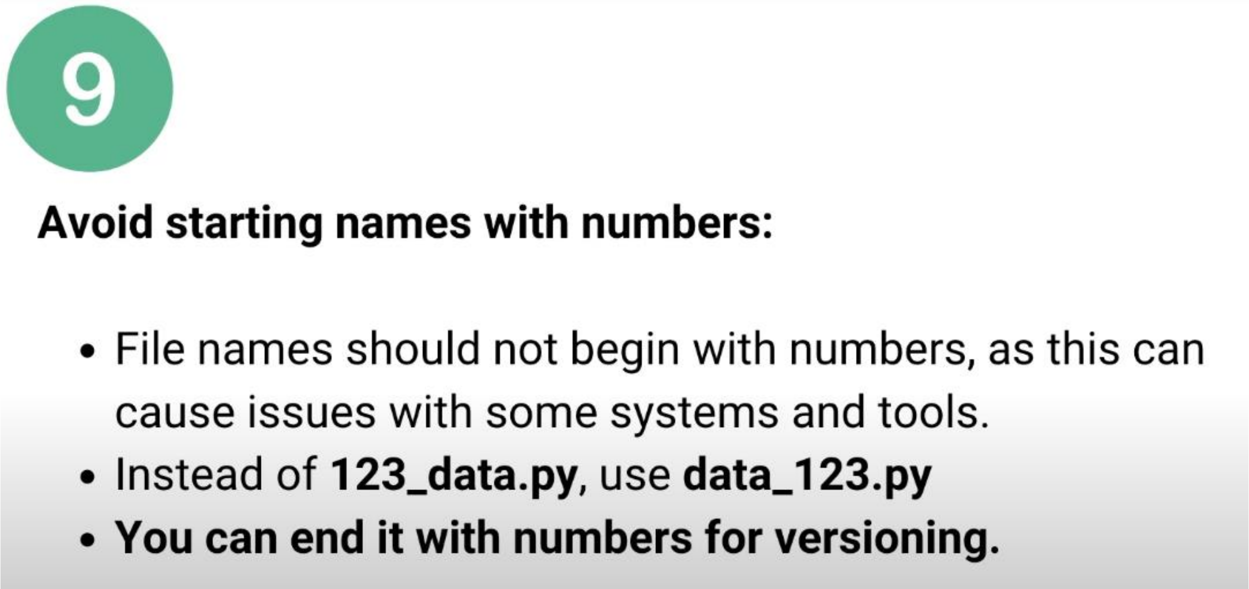
Built-in Functions	Keywords
list, dict, str, int	class, def, return
input, print, sum	for, while, if
open, file, eval, len	import, as, from
type, id, next, min	try, except, with
max, range, set, len	global, lambda, pass



8

Use Limited length:

- Keep file and folder names short but descriptive to avoid long paths. Use concise names where possible.
- Instead of **train_random_forest_classification_model.py**, use **train_rf_model.py**



9

Avoid starting names with numbers:

- File names should not begin with numbers, as this can cause issues with some systems and tools.
- Instead of **123_data.py**, use **data_123.py**
- **You can end it with numbers for versioning.**

10

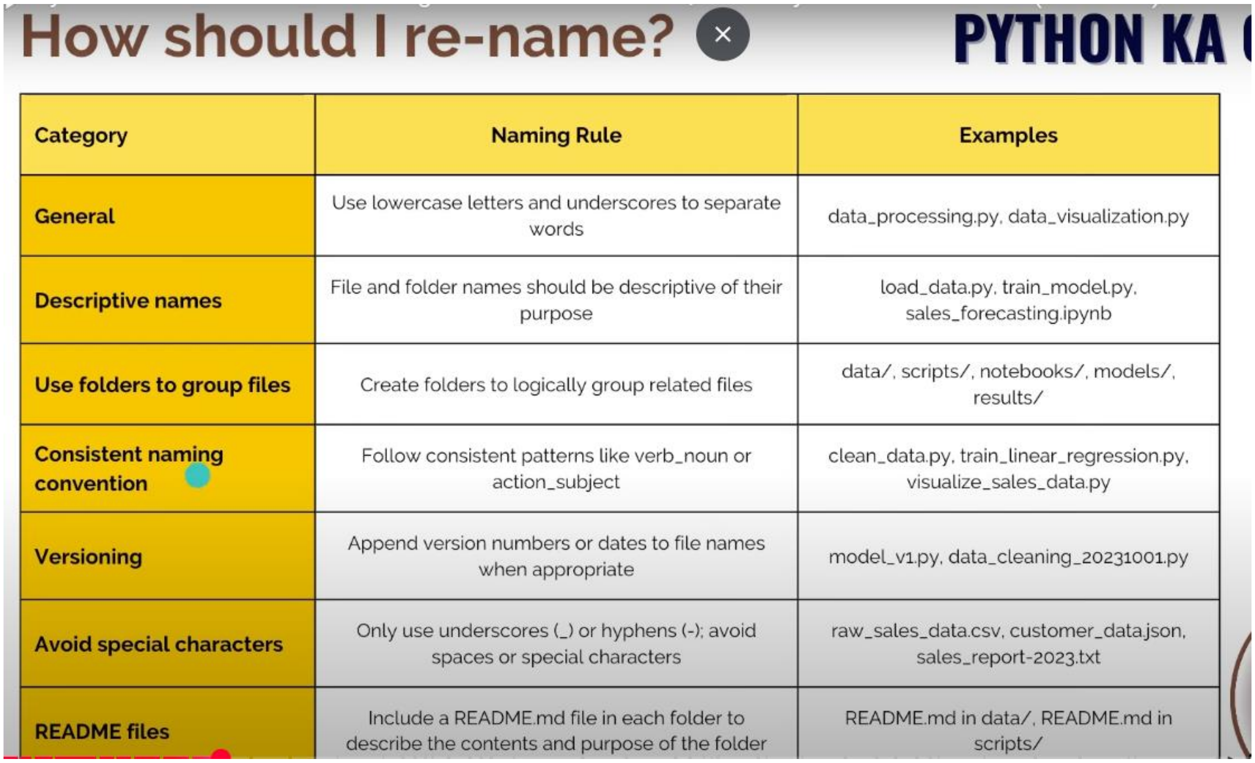
Use plural forms for collections:

- Use plural names for directories/folders containing multiple files of the same type (e.g., **scripts/, datasets/**)
- **scripts/, models/, plots/**

11

Use README files:

- Use a **README.md** in each folder to explain the contents, structure, and any specific instructions.



How should I re-name?

PYTHON KA (

Category	Naming Rule	Examples
General	Use lowercase letters and underscores to separate words	data_processing.py, data_visualization.py
Descriptive names	File and folder names should be descriptive of their purpose	load_data.py, train_model.py, sales_forecasting.ipynb
Use folders to group files	Create folders to logically group related files	data/, scripts/, notebooks/, models/, results/
Consistent naming convention	Follow consistent patterns like verb_noun or action_subject	clean_data.py, train_linear_regression.py, visualize_sales_data.py
Versioning	Append version numbers or dates to file names when appropriate	model_v1.py, data_cleaning_20231001.py
Avoid special characters	Only use underscores (_) or hyphens (-); avoid spaces or special characters	raw_sales_data.csv, customer_data.json, sales_report-2023.txt
README files	Include a README.md file in each folder to describe the contents and purpose of the folder	README.md in data/, README.md in scripts/