|  |  |
| --- | --- |
| [Structuring ML Projects]  [2021] |  |
|  |  |
| [VERSION 0.1]  [TCS]  Authored by: [Yaswant Krishna, Baliram Pinate] | Miss You Bee |

# ContentsHi Bee

# Background.

# Folder Structure.

# Python coding guidelines.

# Useful Python Packages to Make your Life Easier.

# End of Project.

# Learning Resources.

Folder Structure 

Project name: Folder for hosting all files of a typical data science project.

Data: Folder for hosting code.

External: Third party data.

Interim: Transformed intermediate data, not ready for modelling.

Processed: Prepared data, ready for modeling.

Raw: Immutable original data.

Docs: Folder for hosting all documents of a project.

Data dictionaries: Place to put data description documents, typically received from a client.

Data report: Location to place documents describing results of data exploration.

Model: Folder for hosting all documents and reports related to modelling.

Project: Folder for hosting project documents and reports such as project planning docs and presentations.

Models: Serialized models.

Notebooks: Jupyter notebooks for exploration and prototyping.

SRC: Folder for hosting project source code.

Data: Folder containing scripts to download/generate data.

Features: Folder containing scripts to transform data for modelling.

Models: Folder containing scripts to train and predict.

Coding Guidelines 

The Zen of Python:

Beautiful is better than ugly.

Explicit is better than implicit.

Simple is better than complex.

Complex is better than complicated.

Flat is better than nested.

Sparse is better than dense.

Readability counts.

Special cases aren't special enough to break the rules.

Although practicality beats purity.

Errors should never pass silently.

Unless explicitly silenced.

In the face of ambiguity, refuse the temptation to guess.

There should be one-- and preferably only one --obvious way to do it.

Although that way may not be obvious at first unless you're Dutch.

Now is better than never.

Although never is often better than \*right\* now.

If the implementation is hard to explain, it's a bad idea.

If the implementation is easy to explain, it may be a good idea.

Namespaces are one honking great idea -- let's do more of those!

Imports: Imports are always put on top of the file. Imports should be grouped in the following order.

1. Standard library imports.
2. Related third party imports.
3. Local application/library specific imports.

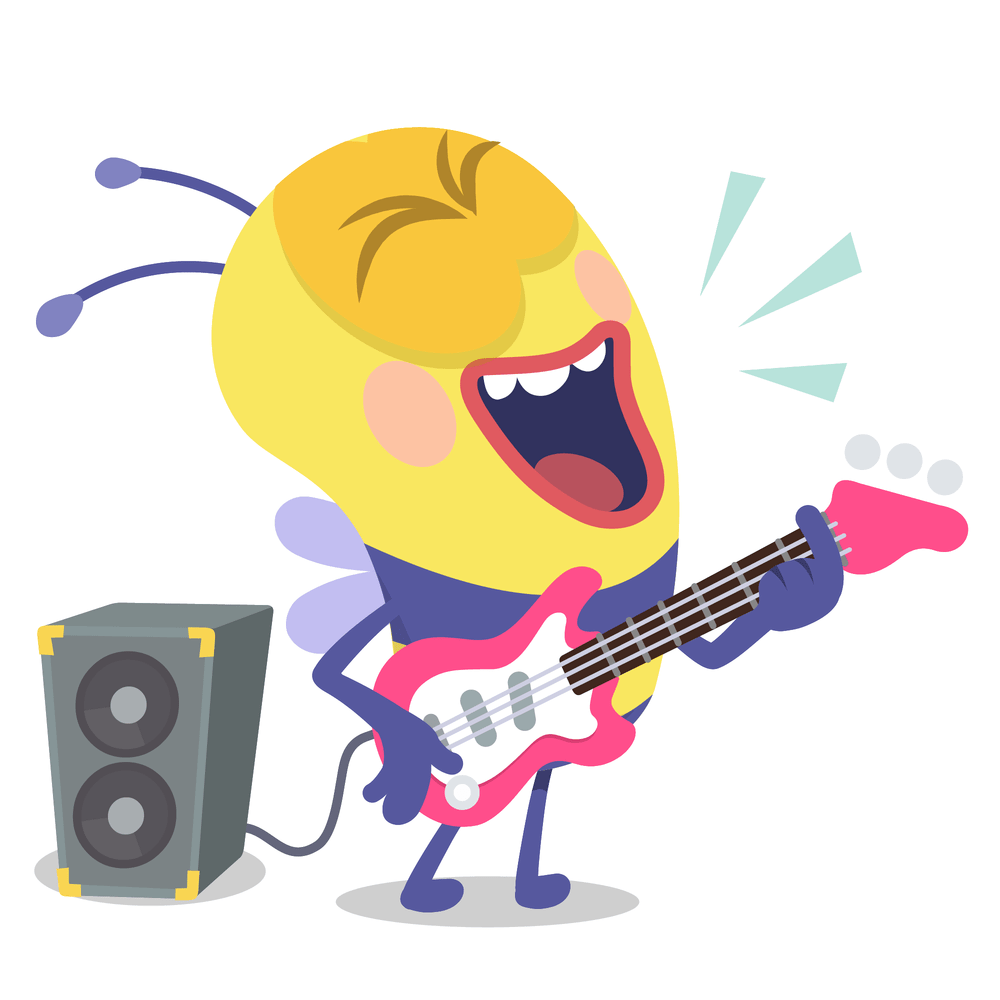
Naming Conventions:

1. Variable names should be lowercase and readable.
2. CONSTANTS should always be in uppercase.
3. Function names should be lowercase and use underscores to improve readability.
4. For class names use CapWords convention.
5. Modules should have short, all lowercase names.

Comments: Comments that contradict the code are worse than no comments.

1. Block Comments: Block comments generally apply to some (or all) code that follows them and are indented to the same level as that code.
2. Inline Comments: Use inline comments sparingly. An inline comment is a comment on the same line as a statement. Inline comments should be separated by at least two spaces from the statement. They should start with a # and a single space.
3. Documentation Strings: Write docstrings for all public modules, functions, classes, and methods.

Exception Handling: Always use try and except blocks wherever necessary to catch exceptions and avoid runtime errors.

Useful Python Packages to Make your Life Easier 

|  |  |  |  |
| --- | --- | --- | --- |
| s.no | Package | Remarks | Link |
| 1. | missingno | Helps to visualize missing values in a pandas dataframe easily. | https://pypi.org/project/missingno/ |
| 2. | tqdm | Instantly make your loops show a smart progress meter and helps you know how much time it takes to execute that loop. | https://pypi.org/project/tqdm/ |
| 3. | prophet | If you are working on time series data. Then this library makes your work easier. | https://facebook.github.io/prophet/ |
| 4. | pycaret | If you want to test various models on the prepared data using a few lines, then this library helps you the most. | https://pycaret.org/ |
| 5. | streamlit | Streamlit turns data scripts into shareable web apps in minutes. | https://streamlit.io/ |

END of Project 