

# README Files

## Examples:

A GitHub repository's `README.md` file is often the first thing potential users or contributors see when they access your repo. It's a crucial document that provides important information about your project.

When you create a repository on GitHub, you have the option to initialize it with a `README.md` file. Once created, GitHub will automatically render the `README` file and display its content on your repository's main page.

A well-written `README` is a showcase of your technical skills, attention to detail, and your ability to communicate complex concepts. It demonstrates your writing skills and your ability to explain and document your code, which is a highly sought-after skill in the industry, and one which even good coders often lack.

A well-documented project can also attract more users and contributors, increasing the visibility and impact of your work, which in turn is appealing to employers. Having a project with a thorough `README` on GitHub shows potential employers that you can manage a project, contribute to open source, and work collaboratively: all important skills in a professional setting.

- 
1. Include here a brief description of the project, what technologies are used etc.
  2. What have you built?
  3. What technologies have you used?
  4. Why have you used those?
  5. Insert an image/screenshot of what you have built so far here.
  6. Does what you have built in this milestone connect to the previous one?
  7. If so explain how.
  8. What technologies are used?
  9. Why have you used them?
  10. Have you run any commands in the terminal?
  11. Insert screenshot of what you have built working.
  12. Continue this process for every milestone, making sure to display clear understanding of each task and the concepts behind them as well as understanding of the technologies used.
  13. Also don't forget to include code snippets and screenshots of the system you are building, it gives proof as well as it being an easy way to evidence your experience!
  14. Maybe write a conclusion to the project, what you understood about it and also how you would improve it or take it further.
  15. Read through your documentation, do you understand everything you've written? Is everything clear and cohesive?
- 

## Markdown

### Markdown

**Cheat Sheet:** <https://www.markdownguide.org/cheat-sheet/>

You can either edit your `README` file on the remote repo itself on Github, or inside your local copy of your repo using VSCode, and then push the resulting changes to the remote repo.

```
# This is a Title (H1 heading)

## This is an H2 heading
[this is a hyperlink](https://www.google.com)

### This is an H3 Heading

- This
- Is
- A
- Bulletpoint
- List

1. This
2. Is
3. A
4. Numbered
5. List

This is how you add an image:
![image info](pictures/image.png)

You can also use any HTML you want in a Markdown file:
<br>
<p align=center><img src=images/example_image.png width=900></p>
<br>
```

## Title

The title of your `README` file should ideally be the same as your project name. Make sure it's descriptive and concise. This will be the first impression users have of your project, so it should quickly convey what your project does.

```
# This is a Title (H1 heading)
```

```
# AirBnB
Data Science Specialization Project With AiCore
Build a framework to systematically train, tune, and evaluate models on several tasks that are tackled by the Airbnb team
```

## Description

This section should briefly explain what your project does and why it exists. It could include the problem your project solves, the target users, and any major features. Aim to make this part as compelling as possible, since it's often the first thing people will read. For example if your repository is a piece of software you have written, briefly outline the problem or the use-case that it addresses, or if it's a data project, describe the question you were trying to solve and outline how your project achieves it.

Here's a checklist of items to include in your description. Not all projects will need all of them, but you should at least ask yourself whether each one applies:

- What was your motivation? Why did you build this project?
- What problem does it solve?
- Why did you make the decisions you made in creating your solution?
- What did you learn?
- What makes your project stand out?
- If your project has a lot of different features, consider adding a "Features" section and listing them here

```
# AirBnB
Data Science Specialization Project With AiCore
Build a framework to systematically train, tune, and evaluate models on
```

This project involves cleaning and analyzing Airbnb data to gain insights. The dataset used in this project contains various features related to Airbnb listings, such as property details, amenities, ratings, and pricing.

```
## Introduction
Airbnb is a popular online marketplace for renting vacation homes, apartments, and lodgings. The data collected from Airbnb listings can provide valuable information about rental trends, guest preferences, and host performances. This project aims to clean the raw Airbnb data, handle missing values, and perform data analysis to extract meaningful insights. Overall, the project's aim is to empower users with meaningful information and trends in the vacation rental market, making it a valuable resource for various stakeholders interested in the Airbnb industry.
```

```
### Problem
The Airbnb Data Cleaning and Analysis Project aims to clean and analyze raw Airbnb data to provide valuable insights and trends in the vacation rental market. The dataset contains diverse information related to Airbnb listings, including property details, amenities, guest ratings, and pricing. By addressing missing data, handling inconsistencies, and performing data analysis, this project seeks to transform the raw data into a structured and meaningful format.
```

Specifically, we want to address the following:

- How can we handle missing data and inconsistencies in the raw Airbnb dataset?
- How can we extract meaningful insights and patterns from the cleaned dataset?
- How can we identify trends in the vacation rental market, such as pricing patterns, amenities preferences, and guest ratings?
- How can we provide a user-friendly solution for exploring and visualizing the data?

```
### Solution
Through the Airbnb Project, users gain access to a cleaned and structured dataset with valuable insights into the vacation rental market. By addressing missing data, performing thorough analysis the project allows property owners, travelers, investors, and data enthusiasts to explore trends, make informed decisions and gain a deeper understanding of the Airbnb market. The user-friendly implementation and data exploration capabilities make the project a valuable resource for anyone interested in the vacation rental market.
```

Whether you are looking to optimize your rental offerings, plan your travels, or analyze investment opportunities, this project equips you with the tools to gain actionable insights from Airbnb data.

```
### Motivation
The Airbnb Data Cleaning and Analysis Project was undertaken as part of the AiCore Data Science Specialization. It served as my first attempt at building a prediction model using neural networks and was a valuable learning exercise utilizing a real-world example.
```

As part of the data science specialization, the project aimed to enhance my understanding of data cleaning, data analysis, and the application of neural networks for predictive modeling. By working on a real-world dataset like Airbnb data, I could explore the intricacies of handling missing data, extracting insights, and identifying trends in the vacation rental market.

Throughout the project, I gained hands-on experience in data preprocessing, data visualization, and implementing neural networks for regression tasks. The aim was to develop a comprehensive solution that could be applied to other similar datasets and provide meaningful insights to diverse stakeholders in the vacation rental market.

The project's motivation was not only to build a prediction model but also to grasp the practical challenges faced while working with real data. It allowed me to refine my data science skills, learn from iterative improvements, and enhance my overall understanding of the data science workflow.

```
### Target Users
The target users of this project are individuals or organizations interested in gaining a deeper understanding of the Airbnb rental market. Potential users include:
```

**Property Owners and Hosts:** Hosts looking to optimize their rental offerings by understanding pricing trends, amenities preferences, and guest ratings.

**Travelers and Guests:** Travelers seeking valuable insights into popular locations, amenities availability, and the overall quality of accommodations.

**Real Estate Investors:** Investors exploring potential investment opportunities in the vacation rental market.

**Data Enthusiasts:** Individuals interested in data analysis, visualization, and gaining insights from real-world datasets.

```
## Major Features
**Data Cleaning:** The project includes a robust data cleaning process to handle missing values, data type conversion, and inconsistency removal. It ensures a clean and reliable dataset for analysis.
```

**Neural Network Predictive Model:** As a significant feature, the project implements a neural network predictive model for regression tasks. It is my first attempt at building a prediction model using neural networks, providing valuable hands-on experience in machine learning.

**Real-World Learning Exercise:** The project serves as a learning exercise utilizing a real-world example of Airbnb data. It allowed me to grasp practical challenges faced while working with real data and refine data science skills.

**Data Analysis and Visualization:**

**Customizable Label Column:** The solution offers the flexibility to specify a target label column during analysis. Users can focus on specific aspects of the rental market by customizing the analysis based on their interests.

**Data Saving:** The cleaned data is saved as "clean\_tabular\_data.csv" to facilitate future analysis and usage.

**User-Friendly Execution:** The project is designed with user-friendliness in mind, providing easy execution of data cleaning and analysis scripts. Individuals with varying programming experience can explore the Airbnb dataset effortlessly.

**Modular Code Structure:** The implementation follows a modular code structure, promoting code reusability and maintainability. Each major feature is encapsulated in separate functions, making it easy to understand and modify the code.

**Real-World Dataset - Airbnb:** The project employs a real-world dataset from Airbnb, ensuring the relevance and practical applicability of the analysis and model building.

```
### What Makes this Project Unique
The project stands out due to its user-friendly implementation, allowing individuals with varying programming experience to explore the Airbnb dataset effortlessly. The ability to specify a target label column during analysis also provides users with customization options based on their specific interests and requirements.
```

## AirBnB

Data Science Specialization Project With AiCore Build a framework to systematically train, tune, and evaluate models on several tasks that are tackled by the Airbnb team

This project involves cleaning and analyzing Airbnb data to gain insights and understand trends in the rental market. The dataset used in this project contains various features related to Airbnb listings, such as property details, amenities, ratings, and pricing.

### Introduction

Airbnb is a popular online marketplace for renting vacation homes, apartments, and lodging. The data collected from Airbnb listings can provide valuable information about rental trends, guest preferences, and host performances. This project aims to clean the raw Airbnb data, handle missing values, and perform data analysis to extract meaningful insights. Overall, the project's aim is to empower users with meaningful information and trends in the vacation rental market, making it a valuable resource for various stakeholders interested in the Airbnb industry.

### Problem

The Airbnb Data Cleaning and Analysis Project aims to clean and analyze raw Airbnb data to provide valuable insights and trends in the vacation rental market. The dataset contains diverse information related to Airbnb listings, including property details, amenities, guest ratings, and pricing. By addressing missing data, handling inconsistencies, and performing data analysis, this project seeks to transform the raw data into a structured and meaningful format.

Specifically, we want to address the following:

- How can we handle missing data and inconsistencies in the raw Airbnb dataset?
- How can we extract meaningful insights and patterns from the cleaned dataset?
- How can we identify trends in the vacation rental market, such as pricing patterns, amenities preferences, and guest ratings?
- How can we provide a user-friendly solution for exploring and visualizing the data?

### Solution

Through the Airbnb Project, users gain access to a cleaned and structured dataset with valuable insights into the vacation rental market. By addressing missing data, performing thorough analysis the project allows property owners, travelers, investors, and data enthusiasts to explore trends, make informed decisions and gain a deeper understanding of the Airbnb market. The user-friendly implementation and data exploration capabilities make the project a valuable resource for anyone interested in the vacation rental market.

Whether you are looking to optimize your rental offerings, plan your travels, or analyze investment opportunities, this project equips you with the tools to gain actionable insights from Airbnb data.

### Motivation

The Airbnb Data Cleaning and Analysis Project was undertaken as part of the AiCore Data Science Specialization. It served as my first attempt at building a prediction model using neural networks and was a valuable learning exercise utilizing a real-world example.

As part of the data science specialization, the project aimed to enhance my understanding of data cleaning, data analysis, and the application of neural networks for predictive modeling. By working on a real-world dataset like Airbnb data, I could explore the intricacies of handling missing data, extracting insights, and identifying trends in the vacation rental market.

Throughout the project, I gained hands-on experience in data preprocessing, data visualization, and implementing neural networks for regression tasks. The aim was to develop a comprehensive solution that could be applied to other similar datasets and provide meaningful insights to diverse stakeholders in the vacation rental market.

The project's motivation was not only to build a prediction model but also to grasp the practical challenges faced while working with real data. It allowed me to refine my data science skills, learn from iterative improvements, and enhance my overall understanding of the data science workflow.

### Target Users

The target users of this project are individuals or organizations interested in gaining a deeper understanding of the Airbnb rental market. Potential users include:

**Property Owners and Hosts:** Hosts looking to optimize their rental offerings by understanding pricing trends, amenities preferences, and guest ratings.

**Travelers and Guests:** Travelers seeking valuable insights into popular locations, amenities availability, and the overall quality of accommodations.

**Real Estate Investors:** Investors exploring potential investment opportunities in the vacation rental market.

**Data Enthusiasts:** Individuals interested in data analysis, visualization, and gaining insights from real-world datasets.

### Major Features

**Data Cleaning:** The project includes a robust data cleaning process to handle missing values, data type conversion, and inconsistency removal. It ensures a clean and reliable dataset for analysis.

**Neural Network Predictive Model:** As a significant feature, the project implements a neural network predictive model for regression tasks. It is my first attempt at building a prediction model using neural networks, providing valuable hands-on experience in machine learning.

**Real-World Learning Exercise:** The project serves as a learning exercise utilizing a real-world example of Airbnb data. It allowed me to grasp practical challenges faced while working with real data and refine data science skills.

**Data Analysis and Visualization:**

**Customizable Label Column:** The solution offers the flexibility to specify a target label column during analysis. Users can focus on specific aspects of the rental market by customizing the analysis based on their interests.

**Data Saving:** The cleaned data is saved as "clean\_tabular\_data.csv" to facilitate future analysis and usage.

**User-Friendly Execution:** The project is designed with user-friendliness in mind, providing easy execution of data cleaning and analysis scripts. Individuals with varying programming experience can explore the Airbnb dataset effortlessly.

**Modular Code Structure:** The implementation follows a modular code structure, promoting code reusability and maintainability. Each major feature is encapsulated in separate functions, making it easy to understand and modify the code.

**Real-World Dataset - Airbnb:** The project employs a real-world dataset from Airbnb, ensuring the relevance and practical applicability of the analysis and model building.

### What Makes this Project Unique

The project stands out due to its user-friendly implementation, allowing individuals with varying programming experience to explore the Airbnb dataset effortlessly. The ability to specify a target label column during analysis also provides users with customization options based on their specific interests and requirements.

## Table of Contents

For larger `README` files, a table of contents is a useful addition. It makes your document easier to navigate. Link each section in the table of contents to the corresponding header in your `README` to allow for quick access.

Tools like [DocToc](#) can automatically generate a table of contents for you.

```
# Table of Contents
```

```
## Introduction
Content here...
```

```
## Section 1
Content here...
```

```
### Subsection 1.1
Content here...
```

```
### Subsection 1.2
Content here...
```

```
# Table of Contents
```

```
1. [Introduction](#introduction)
  - [Problem](#problem)
  - [Solution](#solution)
  - [Target Users](#target-users)
  - [Motivation](#motivation)
2. [Major Features](#major-features)
  - [Data Cleaning](#data-cleaning)
  - [Real-World Learning Exercise](#real-world-learning-exercise)
  - [Customizable Label Column](#customizable-label-column)
  - [Data Saving](#data-saving)
  - [User-Friendly Execution](#user-friendly-execution)
  - [Modular Code Structure](#modular-code-structure)
  - [Real-World Dataset - Airbnb](#real-world-dataset---airbnb)
  - [What Makes this Project Unique](#what-makes-this-project-unique)
3. [Installation](#installation)
4. [Usage](#usage)
5. [Contributing](#contributing)
6. [License](#license)
7. [Technical Breakdown](#technical-breakdown)
  - [Data Loading](#data-loading)
  - [Data Cleaning](#data-cleaning-1)
  - [Save Cleaned Data](#save-cleaned-data)
  - [Label Selection](#label-selection)
  - [Learnings](#learnings)
```

## Table of Contents

- 1. Introduction
  - Problem
  - Solution
  - Target Users
  - Motivation
- 2. Major Features
  - Data Cleaning
  - Real-World Learning Exercise
  - Customizable Label Column
  - Data Saving
  - User-Friendly Execution
  - Modular Code Structure
  - Real-World Dataset - Airbnb
  - What Makes this Project Unique
- 3. Installation
- 4. Usage
- 5. Contributing
- 6. License
- 7. Technical Breakdown
  - Data Loading
  - Data Cleaning
  - Save Cleaned Data
  - Label Selection
  - Learnings

## Installation

If your project is a piece of software, like an application or a package that others might download and use, you should provide detailed instructions on how to get your project running on the reader's local environment. This could include information on necessary software, environment setup, installation of dependencies, and any final steps needed to get the project up and running.

```
## Installation
To run the code in this project, you'll need to have Python installed on your system. Additionally, the following Python libraries are required and can be installed using pip:

pandas

numpy

To install the required libraries, use the following command:
```python pip install pandas numpy```
```

```
Installation

To run the code in this project, you'll need to have Python installed on your system. Additionally, the following Python libraries are required and can be installed using pip:

pandas

numpy

To install the required libraries, use the following command:

python pip install pandas numpy
```

---

## Usage

It is also helpful to give a walkthrough or guide to your project in action. If your repository is a data project, this would be the section where you present your key findings, or it is a piece of software that others might want to use, it should show examples of how to use it. This could include code snippets, command line instructions, screenshots, gifs or videos. The goal here is to demonstrate the value of your project by showing it in action. If your project is a library, then a list of the available functions would go in this section, possibly alongside some worked examples.

```
## Usage
Clone the repository to your local machine:

```python git clone https://github.com/Millie-Jackson/airbnb-data-analysis.git```

Navigate to the project directory:

```python cd airbnb```

Run the data cleaning and analysis scripts:

```python python tabular_data.py```
```

```
Usage

Clone the repository to your local machine:

python git clone https://github.com/Millie-Jackson/airbnb-data-analysis.git

Navigate to the project directory:

python cd airbnb

Run the data cleaning and analysis scripts:

python python tabular_data.py
```

---

## Contributing

If you are happy for people to contribute to the project, eg. by suggesting and coding new features, you would include a contribution guide. In this section, outline the process for interested contributors to help improve your project. This could include information on how to file issues, how to propose improvements, and the process for submitting pull requests. You could also mention any specific coding standards or practices you'd like contributors to follow, or any specific features you are looking for that you are hoping others might be able to provide.

```
## Contributing
Contributions to this project are welcome!
If you find any issues or have suggestions for improvement, please feel free to open an issue or submit a pull request.
```

---

## License

This part informs users about the license under which your project is released, which defines what they are legally allowed to do with your project. If you're unsure what license to use, platforms like [Choose a License](#) can provide guidance.

```
## License
This project is licensed under the MIT License.
```

---

## Technical Breakdown (optional)

```

# Technical Breakdown
**Data Loading:** The code starts by loading the raw Airbnb data from a CSV file ("listing.csv") using the pandas library's read_csv function.

**Data Cleaning:** The data cleaning process involves removing missing values, converting data types, and handling inconsistencies.

The tabular_data.py script calls the "clean_tabular_data(df)" function to perform data cleaning. It serves as the core data cleaning process.

Makes a copy of the original DataFrame ("df_before_update") to track changes.

Calls the remove_rows_with_missing_ratings(df), combine_description_strings(df), and set_default_feature_values(df) functions to clean the data.

Compares if the DataFrame ("df") has been modified after the update and displays a message indicating whether the original DataFrame has been modified.

Reindexes the DataFrame and removes the old index.
(screenshots/clean_tabular_data.png)

The "remove_rows_with_missing_ratings(df)" function is called to remove rows with missing values in the rating columns (Accuracy_rating, Communication_rating).
(screenshots/remove_rows_with_missing_ratings.png)

The "combine_description_strings(df)" function is called to combine and clean the strings in the "Description" column by removing missing values and extra spaces.
(screenshots/combine_description_strings.png)

The "set_default_feature_values(df)" function is called to set default values for the feature columns (guests, beds, bathrooms, bedrooms) to 0 if they are missing.
(screenshots/set_default_feature_values.png)

**Save Cleaned Data:** After data cleaning, the cleaned DataFrame ("df") is saved as "clean_tabular_data.csv" using the to_csv method of pandas.

**Label Selection:**
The program loads the cleaned data from "clean_tabular_data.csv" using pandas to proceed with further analysis.

The program calls the "load_airbnb(label='Price_Night')" function to load the cleaned data for analysis, with "Price_Night" as the label column.

The "load_airbnb" function checks if the specified label column exists in the data; if not, it raises a "ValueError" with an appropriate error message.

The program then filters out non-numeric columns to include only numerical features in the analysis.
(screenshots/load_airbnb.png)

```

## Learnings (optional)

**Guide:** <https://github.com/AI-Core/ExampleDocumentation/blob/main/ExampleDocumentation.md>

```

# Learnings
The Airbnb Data Cleaning and Analysis Project provided a valuable learning experience as it marked several first-time achievements in my data science journey.

**Pandas:** Working on this project introduced me to the powerful Pandas library in Python. I learned to efficiently manipulate and analyze data using Pandas data structures like DataFrames and Series. The library's functions and methods streamlined data cleaning tasks and enabled smooth data exploration.

**Data Cleaning:** Prior to this project, my data cleaning exercises were limited to tutorials and practice datasets. However, working with the real-world Airbnb dataset presented practical challenges like missing values, data inconsistencies, and diverse data types. I gained hands-on experience in addressing these real data issues, enhancing my data wrangling skills significantly.

The project's focus on data cleaning, data analysis, and building a predictive model with neural networks provided a comprehensive understanding of the data science workflow. It allowed me to apply theoretical knowledge acquired through courses to tackle real-world data complexities.

Throughout the project, I learned the importance of data preprocessing and the impact it has on the quality of analysis and model performance. It provided valuable insights into data-driven decision-making and reinforced the significance of clean, reliable data for accurate predictions.

The project's hands-on nature and the exposure to a real-world dataset offered practical perspectives on data science tasks, laying a strong foundation for future data-driven projects.

## Milestone 1-3
Throughout Milestone 1, 2, and 3, I developed a data science project for Airbnb data analysis.

I have implemented a robust data cleaning process to handle missing values, data type conversion, and inconsistencies in the raw Airbnb data.

**Python:**
I have utilized Python as the primary programming language for this project. Python's versatility, extensive libraries, and strong community support made it the ideal choice for this data science project.

**Pandas:**
Pandas is a powerful library in Python for data manipulation and analysis. I have used Pandas to handle data cleaning tasks, data exploration, and data analysis.

**GitHub:**
I have utilized GitHub as the version control platform to manage the project's codebase and track changes throughout development. GitHub provided a collaborative environment for managing the project's code.

**AST:**
The ast (Abstract Syntax Trees) module in Python has been used to parse Python source code and extract information about the code's abstract syntax tree.

```

### Learnings

The Airbnb Data Cleaning and Analysis Project provided a valuable learning experience as it marked several first-time achievements in my data science journey.

**Pandas:** Working on this project introduced me to the powerful Pandas library in Python. I learned to efficiently manipulate and analyze data using Pandas data structures like DataFrames and Series. The library's functions and methods streamlined data cleaning tasks and enabled smooth data exploration.

**Data Cleaning:** Prior to this project, my data cleaning exercises were limited to tutorials and practice datasets. However, working with the real-world Airbnb dataset presented practical challenges like missing values, data inconsistencies, and diverse data types. I gained hands-on experience in addressing these real data issues, enhancing my data wrangling skills significantly.

The project's focus on data cleaning, data analysis, and building a predictive model with neural networks provided a comprehensive understanding of the data science workflow. It allowed me to apply theoretical knowledge acquired through courses to tackle real-world data complexities.

Throughout the project, I learned the importance of data preprocessing and the impact it has on the quality of analysis and model performance. It provided valuable insights into data-driven decision-making and reinforced the significance of clean, reliable data for accurate predictions.

The project's hands-on nature and the exposure to a real-world dataset offered practical perspectives on data science tasks, laying a strong foundation for future data-driven projects.

### Milestone 1-3

Throughout Milestone 1, 2, and 3, I developed a data science project focused on cleaning the Airbnb data. The project encompasses multiple key functionalities and features.

I have implemented a robust data cleaning process to handle missing values, data type conversion, and inconsistencies in the raw Airbnb dataset. The cleaned dataset ensures the reliability and quality of subsequent analyses. The project allows users to specify a target label column during analysis. This customization option enables users to focus on specific aspects of the rental market based on their interests. The cleaned data is saved as "clean\_tubular\_data.csv" to facilitate future analysis and usage. This feature ensures the availability of a reliable and cleaned dataset for further tasks. The project is designed with user-friendliness in mind, providing easy execution of data cleaning and analysis scripts. It accommodates users with varying programming experience. To promote code reusability and maintainability, the implementation follows a modular code structure. Each major feature is encapsulated in separate functions, making the code easy to understand and modify.

#### Python:

I have utilized Python as the primary programming language for this project. Python's versatility, extensive libraries, and strong community support make it an ideal choice for data science tasks. Python's simplicity and readability allow for rapid development and easy debugging. Additionally, its extensive ecosystem of libraries makes it an ideal choice for data science and machine learning tasks.

#### Pandas:

Pandas is a powerful library in Python for data manipulation and analysis. I have used Pandas to handle data cleaning tasks, data exploration, and data wrangling. Pandas provides intuitive data structures (DataFrames and Series) and powerful tools for data manipulation and future analysis. It simplifies complex data cleaning tasks and enables efficient data exploration.

#### GitHub:

I have utilized GitHub as the version control platform to manage the project's codebase and track changes throughout development. GitHub provides version control and collaborative features, enabling seamless collaboration among team members. Although this aspect hasn't been used in this solo project, it is good to learn these for future projects. It allows me to track changes, manage issues, and maintain a well-organized codebase.

#### AST:

The ast (Abstract Syntax Trees) module in Python has been used to parse Python source code and extract information about the code's abstract syntax structure. The ast module is used to parse complex strings in the "Description" column during data cleaning. By utilizing the abstract syntax trees, we can handle intricate structures and perform data transformations effectively.

## Badges (optional)

**Documentation:** <https://shields.io/badges>

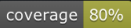








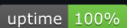
**Resources:** <https://github.com/alexandresanlim/Badges4-README.md-Profile>

**Resources:** <https://github.com/anuraghazra/github-readme-stats>

**Resources:** <https://github.com/sorouchchehresa/github-readme-linkedin>

Badges are small, dynamic status messages that you can embed in your **README** to provide quick information about the project. For example, build status, number of people following the repository, and latest package version. Tools like [Shields.io](https://shields.io) offer a wide variety of badges you can use. You can also find Shields.io's repo [here](#).

## Examples

- code coverage percentage: 
- stable release version: 
- package manager release: 
- status of third-party dependencies: 
- static code analysis grade: 
- SemVer version observance: 
- amount of [Liberapay](#) donations per week: 
- Python package downloads: 
- Chrome Web Store extension rating: 
- Uptime Robot percentage: 

## GIFs

Animated GIFs can be an engaging way to demonstrate the functionality of your project. Tools like [LICEcap](#) for Windows and macOS or [Peek](#) for Linux make it easy to record your screen and save the result as a GIF. [GIPHY Capture](#) is another tool for macOS that allows you to create high-quality GIFs.

## Screenshots

Including screenshots in your [README](#) can give readers a quick understanding of what your project does. Use your operating system's built-in screenshot tools (like **Snipping Tool** or **Snip & Sketch** on Windows, or **SHIFT+CMD+4** on MacOS), then you can annotate and upload these screenshots directly to GitHub when editing your [README](#).

---

## Code formatting and highlighting

Markdown supports inline code and code blocks, but if you want more control over the display and highlighting of your code, you might use something like [Carbon](#). This allows you to create and share beautiful images of your source code.

---

## Charts and diagrams

If your project has a complex architecture, it might be helpful to include a chart or diagram. Tools like [draw.io](#) or [Mermaid](#) allow you to create flowcharts, sequence diagrams, Gantt charts, and more.