**Final Report of Internship Program 2021**

*On*

***“ANALYSIS OF FITNESS DATA”***

**MEDTOUREASY**



24th April 2021

**ACKNOWLDEGMENTS**

The internship opportunity that I had with MedTourEasy was a great change for learning and understanding the intricacies of the subject of Data Visualizations in Data Analytics; and also, for personal as well as professional development. I am very obliged for having a chance to interact with so many professionals who guided me throughout the internship

project and made it a great learning curve for me.

Firstly, I express my deepest gratitude and special thanks to the Training & Developement Team of MedTourEasy who gave me an opportunity to carry out my internship at their esteemed organization. Also, I express my thanks to the team for making me understand the details of the Data Science profile and training me in the

same so that I can carry out the project properly and with maximum client satisfaction

and also for spearing his valuable time in spite of his busy schedule.

I would also like to thank the team of MedTourEasy and my colleagues who made the working environment productive and very conducive.

**TABLE OF CONTENTS**

Acknowledgments .................................... i

Abstract .....................................................iii

**Sr. No. Topic Page No.**

**1** Introduction

1.1 About the Company 5

1.2 About the Project 5

1.3 Objectives and Deliverables 6

**2** Methodology 7

2.1 Flow of the Project 7

2.2 Use Case Diagram 8

2.3 Language and Platform Used 9

**3** Implementation 14

3.1 Gathering Requirements and Defining Problem Statement 14

3.2 Obtain and review raw data 14

3.3 Data preprocessing 16

3.4 Dealing with missing values 18

3.5 Plot running data 20

3.6 Running statistics 21

3.7 Visualization with averages 23

3.8 Did I reach my goals? 24

3.9 Am I progressing? 25

3.10 Training intensity 26

3.11 Detailed summary report 27

**4** Sample Screenshots and Observations 28

5 Conclusion 35

6 References 36

**ABSTRACT**

The understanding of the term "physical fitness" was determined for a randomly selected sample (n = 94) of a population using a self-administered mailed questionnaire. Subjects were asked to state and give a reason for their perceived level of physical fitness, to state their perceived performance level in a number of physical fitness tests (muscular strength, daily physical work capacity, fatness, level of regular physical exercise, exercise speed, and body flexibility), and to rate how well these tests measure physical fitness. The reason most frequently stated for perceived level of physical fitness was the level of habitual physical activity (43%); significantly less frequently (P less than 0.01-0.0001) cited were reasons related to health (23%), physical performance (12%), and obesity (3%). The variation in perceived level of physical fitness was best explained by the variation in imagined regular exercise and fatness (r2 = 0.66, P less than 0.0001) with no significant additional contribution from imagined performance in remaining fitness tests. The measurement of regular exercise was most favored as a test of physical fitness. These results, taken together with evidence of the physical and psychological health benefits of regular exercise, imply that the most appropriate measure of physical fitness for the average person is an assessment of the habitual physical activity level.

1.1 About the Company

MedTourEasy, a global healthcare company, provides you the informational resources needed to evaluate your global options. It helps you find the right healthcare solution based on specific health needs, affordable care while meeting the quality standards that you expect to have in healthcare. MedTourEasy improves access to healthcare for people everywhere. It is an easy to use platform and service that helps patients to get medical second opinions and to schedule affordable, high-quality medical treatment abroad.

1.2 About the Project

One day, my old running friend and I were chatting about our running styles, training habits, and achievements, when I suddenly realized that I could take an in-depth analytical look at my training. I have been using a popular GPS fitness tracker called [Runkeeper](https://runkeeper.com/) for years and decided it was time to analyze my running data to see how I was doing.

Since 2012, I've been using the Runkeeper app, and it's great. One key feature: its excellent data export. Anyone who has a smartphone can download the app and analyze their data like we will in this notebook.

After logging your run, the first step is to export the data from Runkeeper (which I've done already). Then import the data and start exploring to find potential problems. After that, create data cleaning strategies to fix the issues. Finally, analyze and visualize the clean time-series data.

I exported seven years worth of my training data, from 2012 through 2018. The data is a CSV file where each row is a single training activity. Let's load and inspect it.

Each of the above sub-section has been represented in the form of dashboards which are created using Python language on Jupyter IDE and different packages. These Data Frames use a wide array of functions and packages in pandas to create intuitive and drillable Data Frames, which can then be used by the firm to analyze the situation and draw conclusions about the same

1.3 Objectives and Deliverables

This project focuses on importing the data and start exploring to find potential problems. After that, create data cleaning strategies to fix the issues. Finally, analyze and visualize the clean time-series data.Also to find Totals for different training types like Cycling , Running and Walking.

The project consists of detailed as follows :

With the explosion in fitness tracker popularity

, runners all of the world are collecting data with

gadgets (smartphones, watches, etc.) to keep themselves motivated. They look for answers to

questions like:

**• How fast, long, and intense was my run today?**

**• Have I succeeded with my training goals?**

**• Am I progressing?**

**• What were my best achievements?**

**• How do I perform compared to others?**

This data was exported from Runkeeper

. The data is a CSV file where each row is a single training

activity

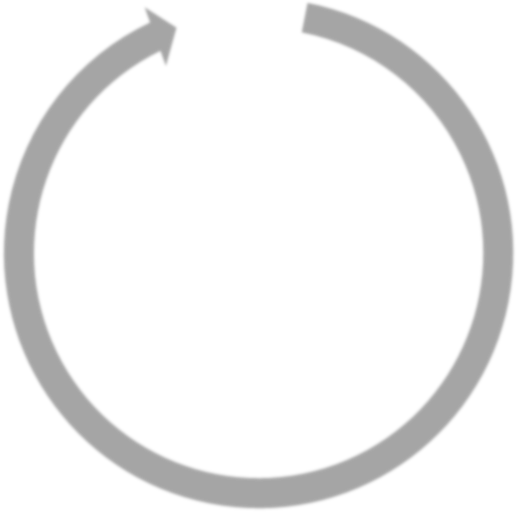
. In this project, we will create import, clean, and analyze my data to answer the above

questions.

**I. METHODOLOGY**

2.1 Flow of the Project

The project followed the following steps to accomplish the desired objectives and deliverables. Each step has been explained in detail in the following section.



**Gathering**

**Requirements**

**& Defining**

**Problem**

**Publishing Data**

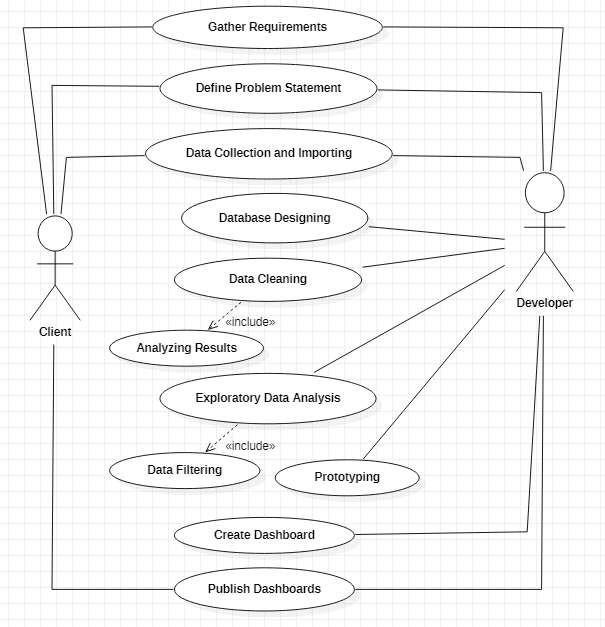
**Development**

**of Designing**

**Prototyping Data**

**Data**

2.2 Use Case Diagram



Above figure shows the use case of the project. There are two main actors in the same: The Client and Developer. The developer will first gather requirements and define the problem statement then collecting the required data and importing it. Then the developer will design databases so as to identify various constraints and relations in the data. Next step is to clean the data to remove irregular values, blank values etc. Next, exploratory data analysis is conducted to filter the data according to the requirements of the project. Then a prototype of the dashboards is created using PowerBI to get a clear view of the visualizations to be developed. Finally, dashboard is developed and analyzed to publish the

Results to the client.

2.3 Language and Platform Used

2.3.1 Language: Python

**Python** is an [open source](https://simple.wikipedia.org/wiki/Open_source) [programming language](https://simple.wikipedia.org/wiki/Programming_language) that was made to be easy-to-read and powerful. A [Dutch](https://simple.wikipedia.org/wiki/Netherlands) programmer named [Guido van Rossum](https://simple.wikipedia.org/wiki/Guido_van_Rossum) made Python in 1991. He named it after the television show [Monty Python's Flying Circus](https://simple.wikipedia.org/wiki/Monty_Python%27s_Flying_Circus). Many Python examples and tutorials include jokes from the show. Python is a good programming language for beginners. It is a high-level language, which means a programmer can focus on what to do instead of how to do it. Writing programs in Python takes less time than in some other languages.

**Python use:**

Python is used by hundreds of thousands of programmers and is used in many places. Sometimes only Python code is used for a program, but most of the time it is used to do simple jobs while another programming language is used to do more complicated tasks.

Its [standard library](https://simple.wikipedia.org/w/index.php?title=Standard_library&action=edit&redlink=1) is made up of many [functions](https://simple.wikipedia.org/wiki/Computable_function) that come with Python when it is installed. On the [Internet](https://simple.wikipedia.org/wiki/Internet) there are many other [libraries](https://simple.wikipedia.org/w/index.php?title=Library_(computing)&action=edit&redlink=1) available that make it possible for the Python language to do more things. These libraries make it a powerful language; it can do many different things.

Some things that Python is often used for are:

* Web development
* Scientific programming
  + [Data science](https://simple.wikipedia.org/wiki/Data_science)
  + [Machine learning](https://simple.wikipedia.org/wiki/Machine_learning)
  + [Numerical analysis](https://simple.wikipedia.org/wiki/Numerical_analysis)
  + [Statistics](https://simple.wikipedia.org/wiki/Statistics)
* Desktop [GUIs](https://simple.wikipedia.org/wiki/GUI) applications
* Network programming
* [Game](https://simple.wikipedia.org/wiki/Video_game) programming.
* Complex [algorithms](https://simple.wikipedia.org/wiki/Algorithm) creation
* Automation scripts
* [Machine learning](https://simple.wikipedia.org/wiki/Machine_learning) and artificial intelligence
* Audio and video application

2.3.2 IDE: **Anaconda**

**Anaconda** is a [distribution](https://en.wikipedia.org/wiki/Software_distribution) of the [Python](https://en.wikipedia.org/wiki/Python_(programming_language)) and [R](https://en.wikipedia.org/wiki/R_(programming_language)) programming languages for [scientific computing](https://en.wikipedia.org/wiki/Scientific_computing) ([data science](https://en.wikipedia.org/wiki/Data_science), [machine learning](https://en.wikipedia.org/wiki/Machine_learning) applications, large-scale data processing, [predictive analytics](https://en.wikipedia.org/wiki/Predictive_analytics), etc.), that aims to simplify [package management](https://en.wikipedia.org/wiki/Package_management) and deployment. The distribution includes data-science packages suitable for Windows, Linux, and macOS. It is developed and maintained by Anaconda, Inc., which was founded by Peter Wang and [Travis Oliphant](https://en.wikipedia.org/wiki/Travis_Oliphant) in 2012.[[8]](https://en.wikipedia.org/wiki/Anaconda_(Python_distribution)#cite_note-8) As an Anaconda, Inc. product, it is also known as **Anaconda Distribution** or **Anaconda Individual Edition**, while other products from the company are Anaconda Team Edition and Anaconda Enterprise Edition, both of which are not free

Major features are:

The Anaconda distribution comes with the following applications along with Anaconda Navigator.

* JupyterLab
* Jupyter Notebook
* Qt Console
* Spyder
* Glueviz
* Orange3
* RStudio
* Visual Studio Code

> **JupyterLab:**

This is an extensible working environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.

>**Jupyter Notebook:**

This is a web-based, interactive computing notebook environment. We can edit and run human-readable docs while describing the data analysis.

> **Qt Console:**

It is the PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips and more.

> **Spyder:**

Spyder is a scientific Python Development Environment. It is a powerful Python IDE with advanced editing, interactive testing, debugging and introspection features.

> **VS Code:**

It is a streamlined code editor with support for development operations like debugging, task running and version control.

> **Glueviz:**

This is used for multidimensional data visualization across files. It explores relationships within and among related datasets.

>**Orange 3:**

It is a component-based data mining framework. This can be used for data visualization and data analysis. The workflows in Orange 3 are very interactive and provide a large toolbox.

>**Rstudio:**

It is a set of integrated tools designed to help you be more productive with R. It includes R essentials and notebooks.

• **Compiled with Latest Python release:**

Anaconda 5.3 is compiled with Python 3.7, taking advantage of Python’s speed and feature improvements.

• **Better Reliability:**

The reliability of Anaconda has been improved in the latest release by capturing and storing the package metadata for installed packages.

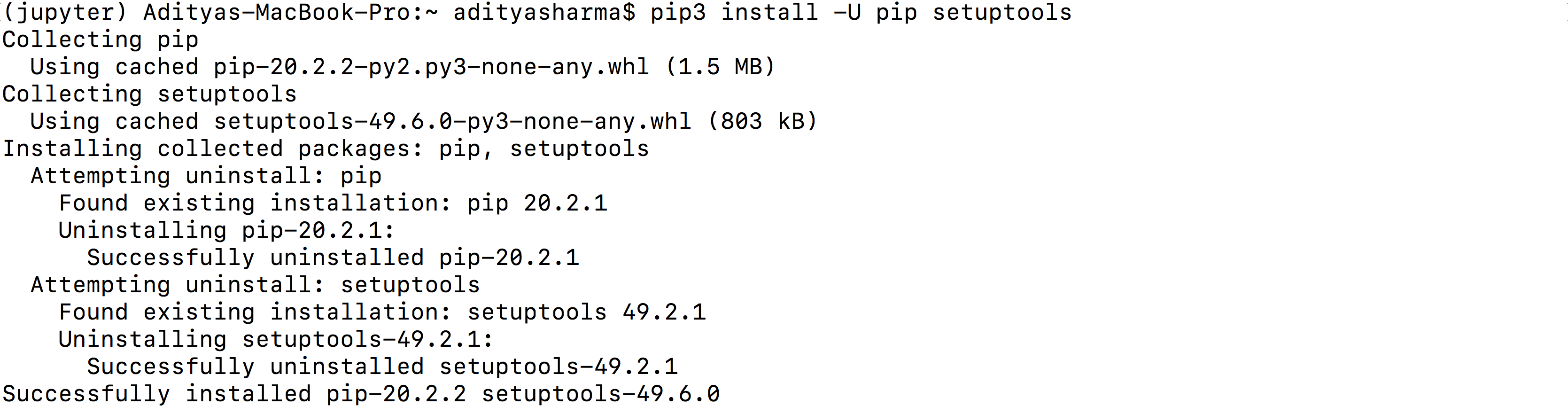
• **Enhanced CPU Peformance:**

The Intel Math Kernel Library 2019 for Deep Neural Networks(MKL 2019) has been introduced in Anaconda 5.3 distribution. Users deploying Tensorflow can make use of MKL 2019 for Deep Neural Networks. These Python binary packages are provided to achieve high CPU performance.

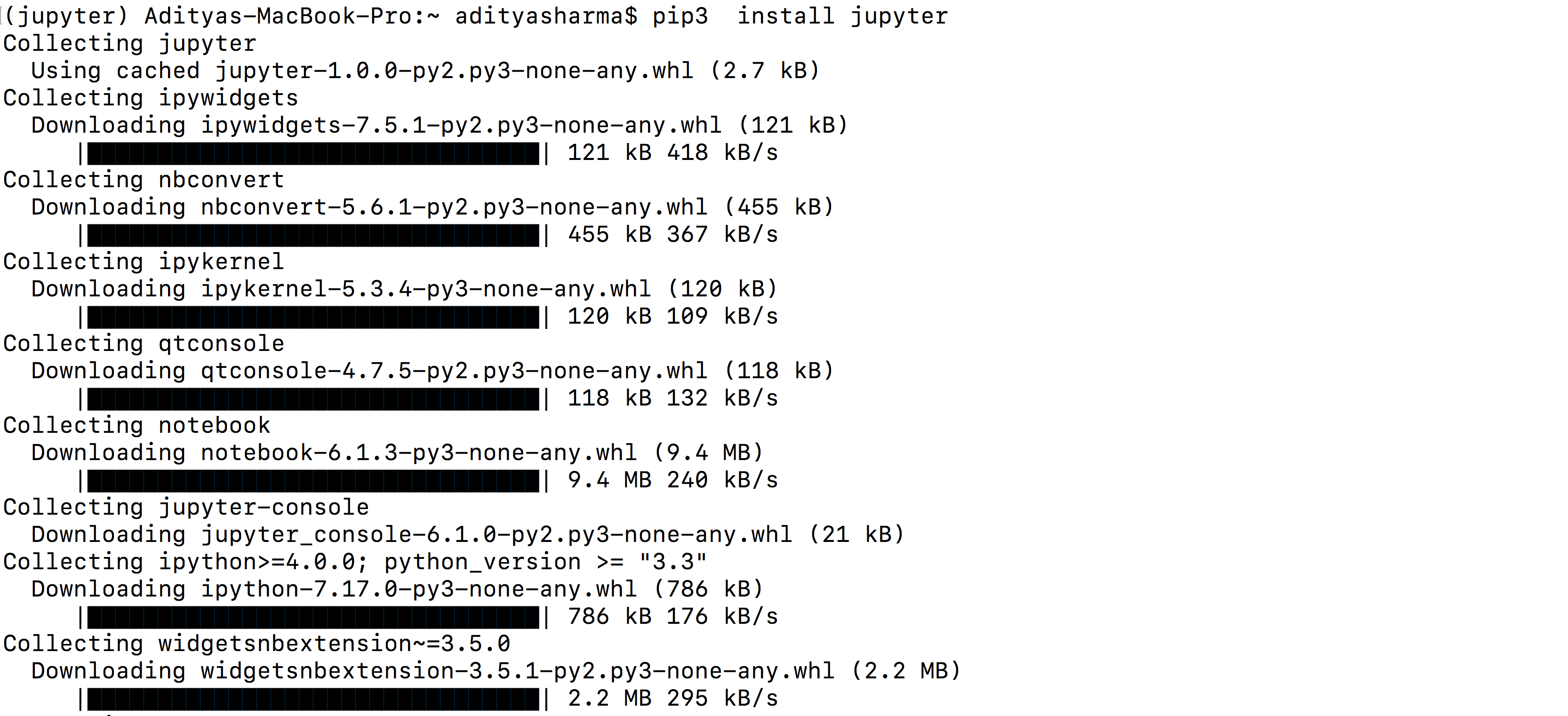
• **New packages are added:** There are over 230 packages which has been updated and added in the new release.

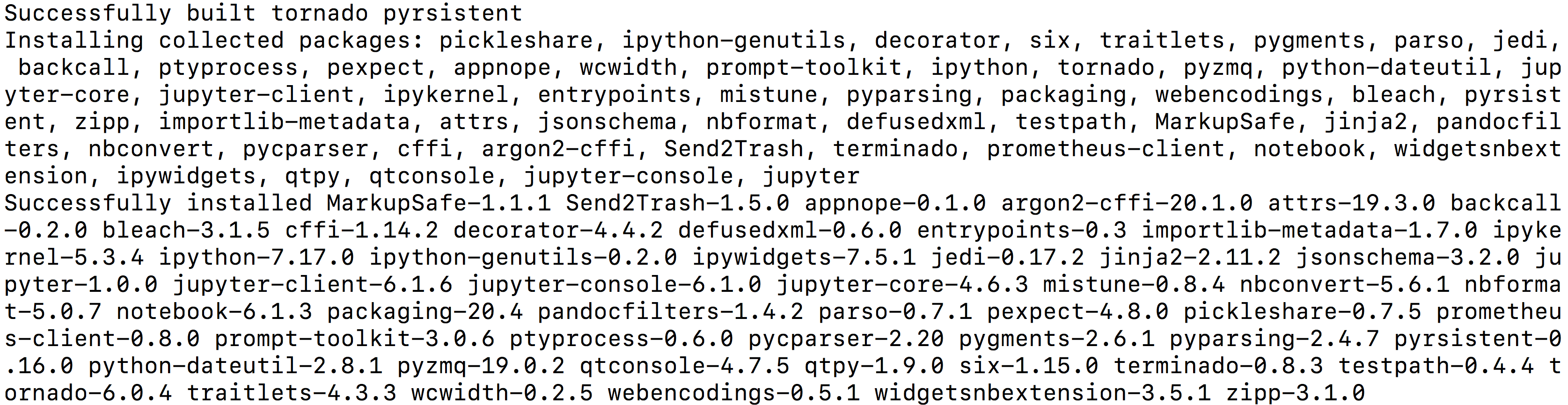
**Installing Packages:**

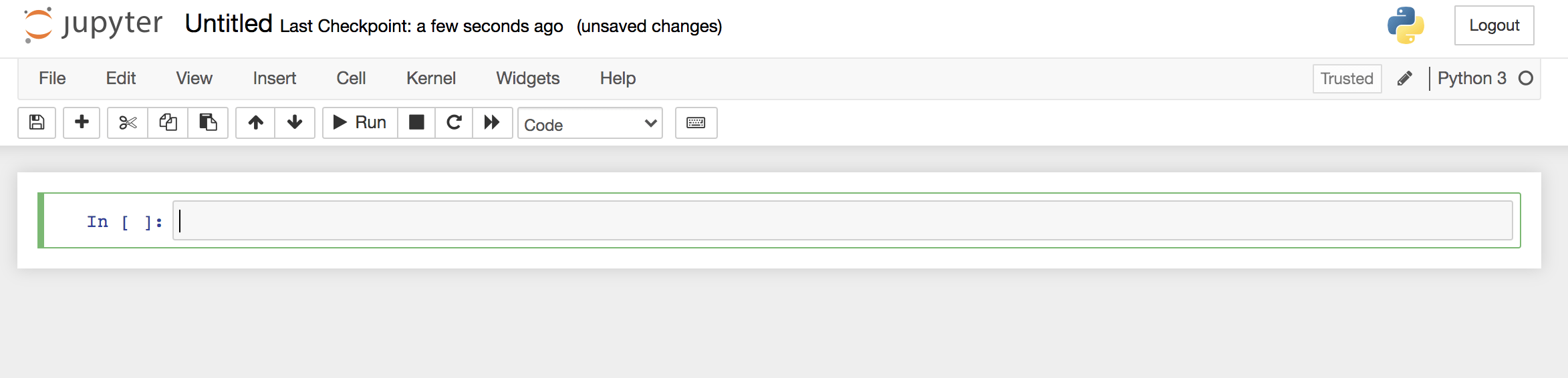
**python -m pip install -U pip setuptools**

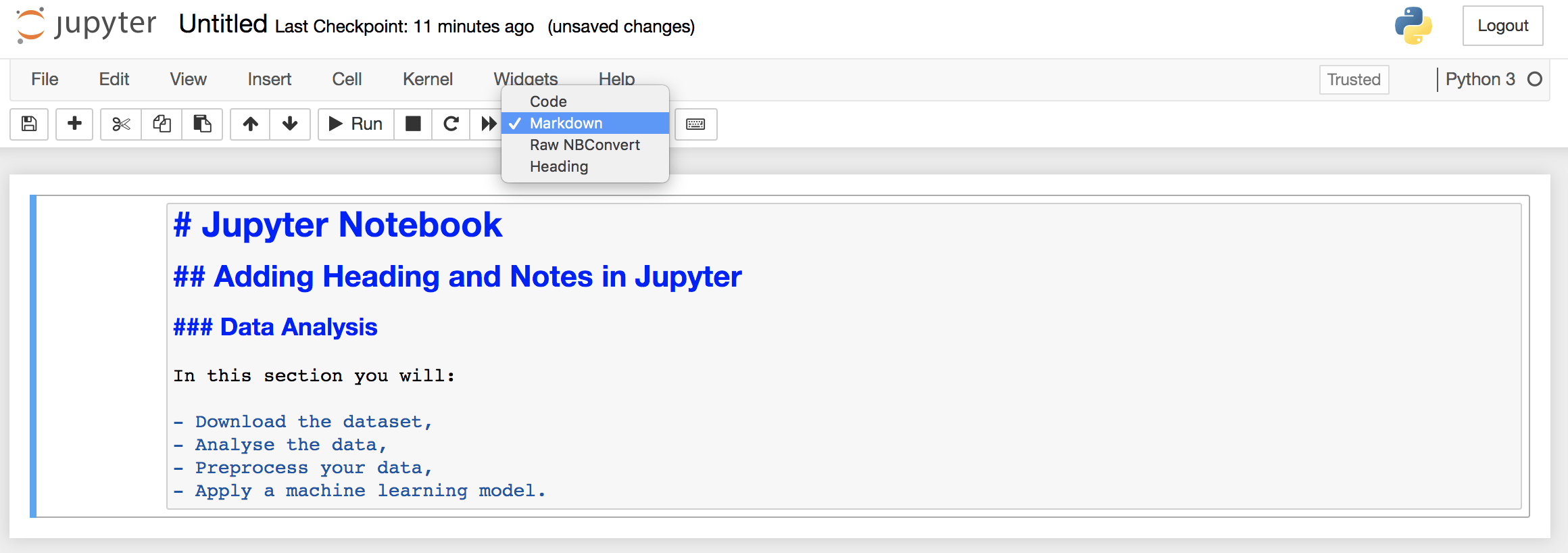


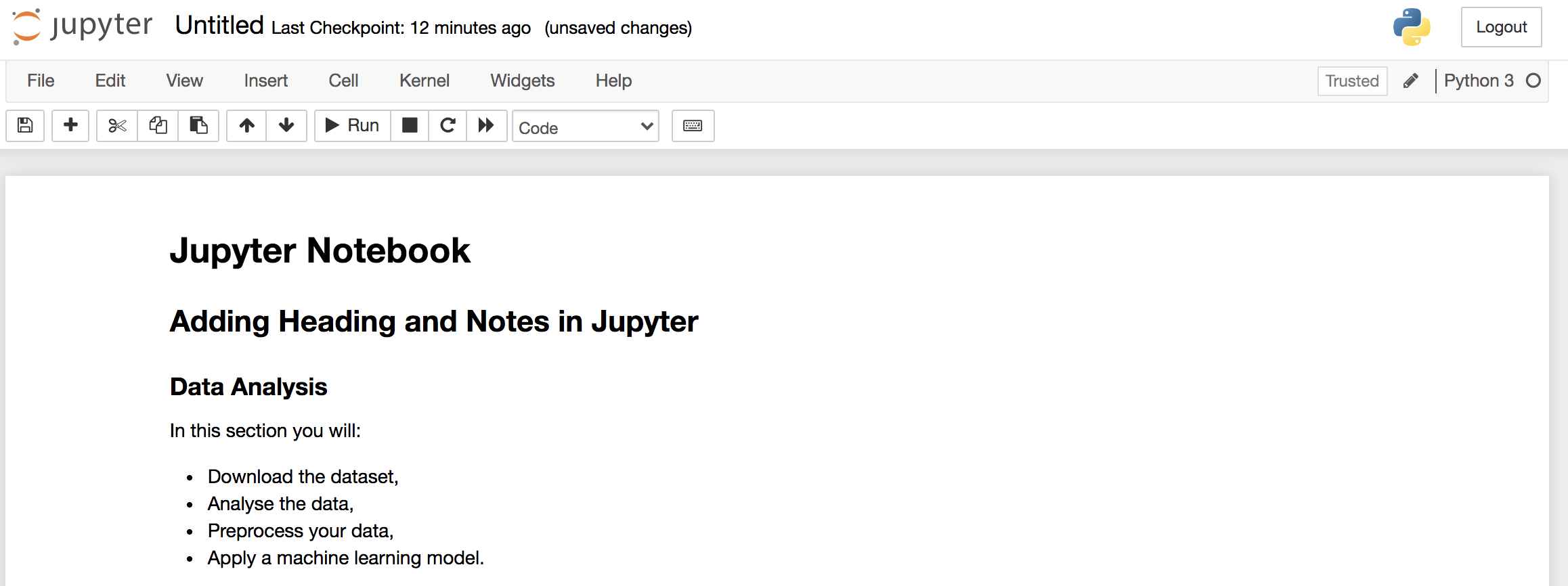
**pip3 install jupyter**











**II. IMPLEMENTATION**

3.1 Gathering Requirements and Defining Problem Statement

This is the first step wherein the requirements are collected from the clients to understand the deliverables and goals to be achieved after which a problem statement is defined which has to be adhered to while development of the project.

3.2 **Obtain and review raw data**

Data collection is a systematic approach for gathering and measuring information from a variety of sources in order to obtain a complete and accurate picture of an interest area. It helps an individual or organization to address specific questions, determine outcomes and forecast future probabilities and patterns.

The data Fitness Data analysis has been collected through various GitHub repositories and can also be obtained from the google docs through a particular link

Data importing is referred to as uploading the required data into the coding environment from internal sources (computer) or external sources (online websites and data repositories). This data can then be manipulated, aggregated, filtered according to the requirements and needs of the project.

Packages Used:

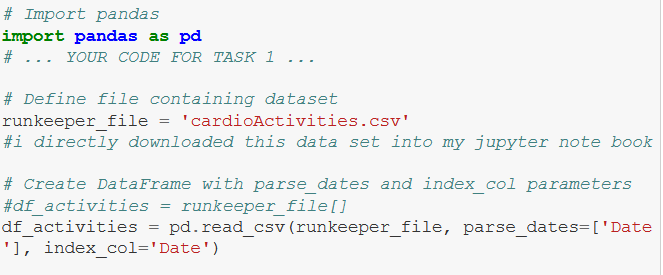
**Read**: The goal of read is to provide a fast and friendly way to read rectangular data (like csv, tsv, and fwf). It is designed to flexibly parse many types of data found in the wild, while still cleanly failing when data unexpectedly changes. To accurately read a rectangular dataset with readr, one needs to combine two pieces: a function that parses the overall file, and a column specification.

Functions Used:

**Read\_csv ()**: It is a wrapper function for [read.table() that](https://renenyffenegger.ch/notes/development/languages/R/functions/read/table) mandates a comma as separator and uses the input file's first line as header that specifies the table's column names. Thus, it is an ideal candidate to read [CSV files.](https://renenyffenegger.ch/notes/development/Data/exchange/formats/CSV/index) It has an additional parameter of url() which is used to pull live data directly from GitHub repository.

**read\_excel ():** It calls excel\_format() to determine if path is xls or xlsx, based on

the file extension and the file itself, in that order



3.4 **Data preprocessing**

***“Quality data beats fancy algorithms”***

Data is the most imperative aspect of Analytics and Machine Learning. Everywhere in computing or business, data is required. But many a times, the data may be incomplete, inconsistent or may contain missing values when it comes to the real world. If the data is corrupted then the process may be impeded or inaccurate results may be provided. Hence, Data cleaning is considered a foundational element of the basic data science.

Data Cleaning means the process by which the incorrect, incomplete, inaccurate, irrelevant or missing part of the data is identified and then modified, replaced or deleted as needed.

With reference to the cardioActivities dataset, it may contain many null values or incorrect value simply because of inconsistency in reporting cases and testing statistics by countries and states. Hence various functions are used to clean this data.

Packages Used:

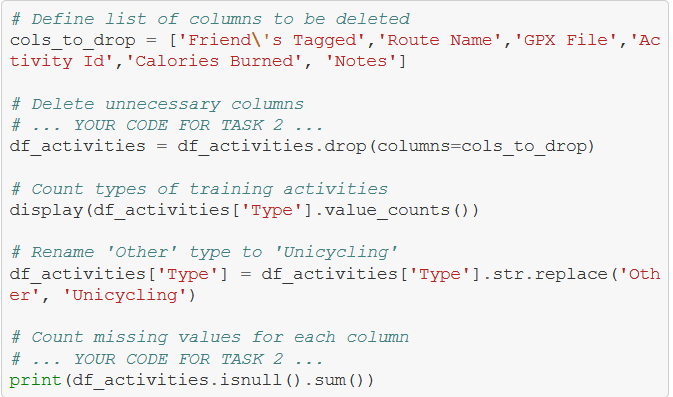
Pandas and Numpy

Functions Used:

**Is.na():** In Python, missing values are represented by the symbol **NA** (not available). Impossible values (e.g., dividing by zero) are represented by the symbol **NaN** (not a number). This function is used to check if a dataset contains NA values

Or not

* Dropping unnecessary columns in a DataFrame
* Changing the index of a DataFrame
* Using .str() methods to clean columns
* Using the DataFrame.applymap() function to clean the entire dataset, element-wise
* Renaming columns to a more recognizable set of labels
* Skipping unnecessary rows in a CSV file



3.5 **Dealing with missing values**

Data filtering is the method of choosing a smaller portion of the data set and using that subset to view, analyze and evaluate data. Generally, filtering is temporary – the entire data set is retained, but only part of it is used for calculation. It is also called subsetting or drill down data wherein data is extracted with respect to certain defined logical conditions. Filtering is used for the following tasks:

 Analyzing results for a particular period of time.

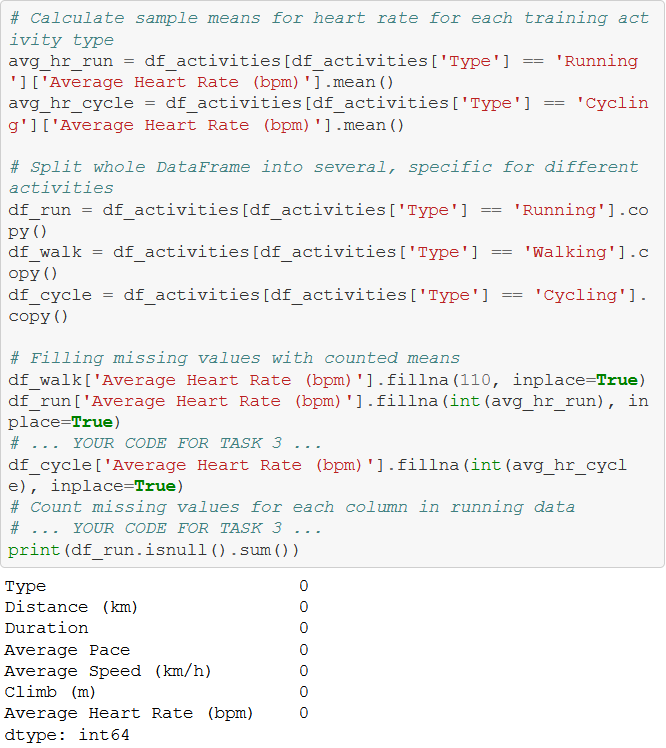
 Calculating results for particular groups of interest.

 Exclude erroneous or "bad" observations from an analysis.

 Train and validate statistical models.

As we can see from the last output, there are 214 missing entries for my average heart rate.

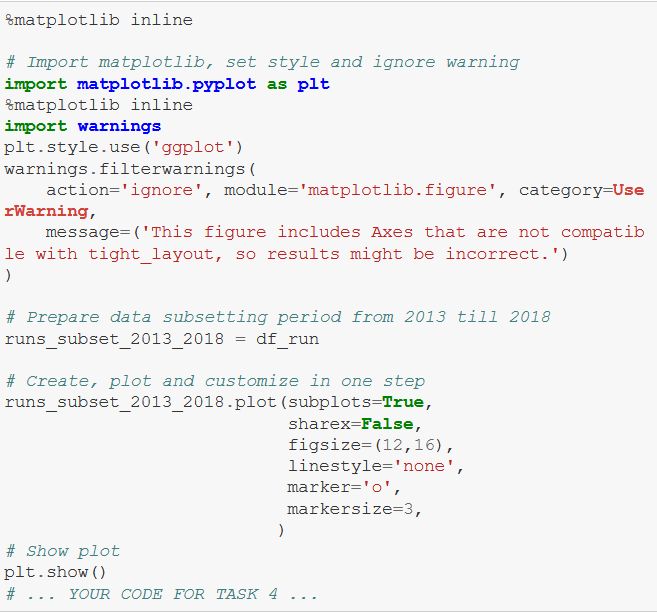
We can't go back in time to get those data, but we can fill in the missing values with an average value. This process is called *mean imputation*. When imputing the mean to fill in missing data, we need to consider that the average heart rate varies for different activities (e.g., walking vs. running). We'll filter the DataFrames by activity type (Type) and calculate each activity's mean heart rate, then fill in the missing values with those means.



3.6 **Plot running data**

A prototype is an early version, model, or release of a product that is constructedto test a design or process. It is generally used by system analysts and users to assess a new design to enhance precision. Prototyping serves to specify a real, working system rather than a theoretical one. Creation of a prototype in some design workflow models is the step between formalizing and testing an idea.

The screenshots of the prototype are as follows:



3.7 **Running statistics**

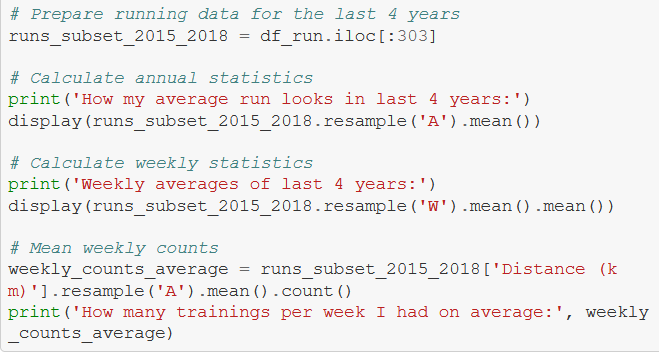
No doubt, running helps people stay mentally and physically healthy and productive at any age. And it is great fun! When runners talk to each other about their hobby, we not only discuss our results, but we also discuss different training strategies.

You'll know you're with a group of runners if you commonly hear questions like:

* What is your average distance?
* How fast do you run?
* Do you measure your heart rate?
* How often do you train?

Let's find the answers to these questions in my data. If you look back at plots in Task 4, you can see the answer to, *Do you measure your heart rate?* Before 2015: no. To look at the averages, let's only use the data from 2015 through 2018.

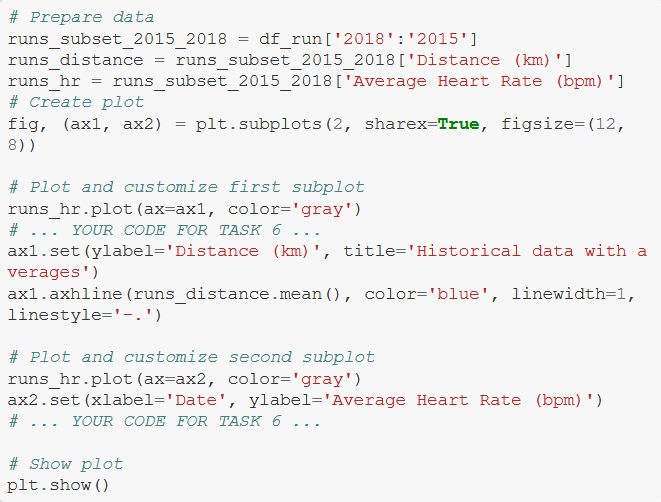
In pandas, the resample() method is similar to the groupby() method - with resample() you group by a specific time span. We'll use resample() to group the time series data by a sampling period and apply several methods to each sampling period. In our case, we'll resample annually and weekly.



**3.7 Visualization with averages**

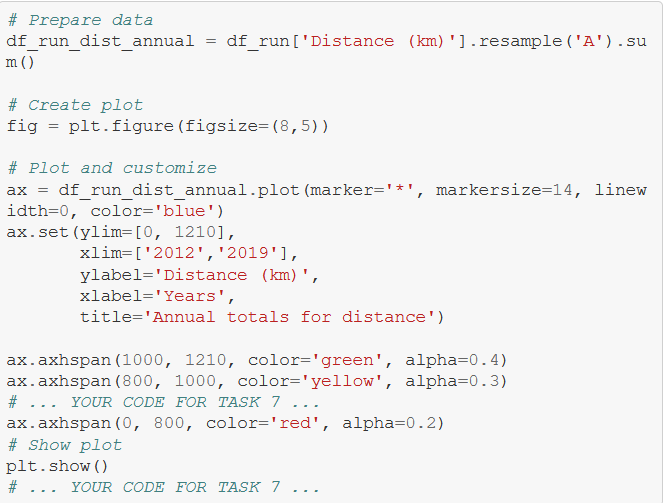
Let's plot the long term averages of my distance run and my heart rate with their raw data to visually compare the averages to each training session. Again, we'll use the data from 2015 through 2018.

In this task, we will use matplotlib functionality for plot creation and customization.

****

**3.8 Did I reach my goals?**

To motivate myself to run regularly, I set a target goal of running 1000 km per year. Let's visualize my annual running distance (km) from 2013 through 2018 to see if I reached my goal each year. Only stars in the green region indicate success.

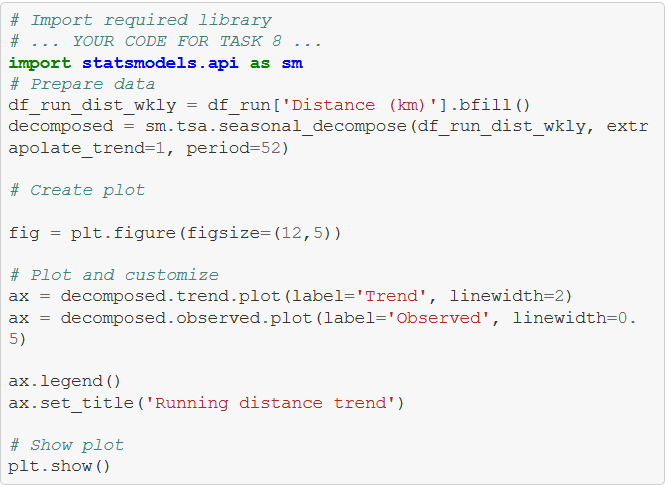
****

**3.9 Am I progressing?**

Let's dive a little deeper into the data to answer a tricky question: am I progressing in terms of my running skills?

To answer this question, we'll decompose my weekly distance run and visually compare it to the raw data. A red trend line will represent the weekly distance run.

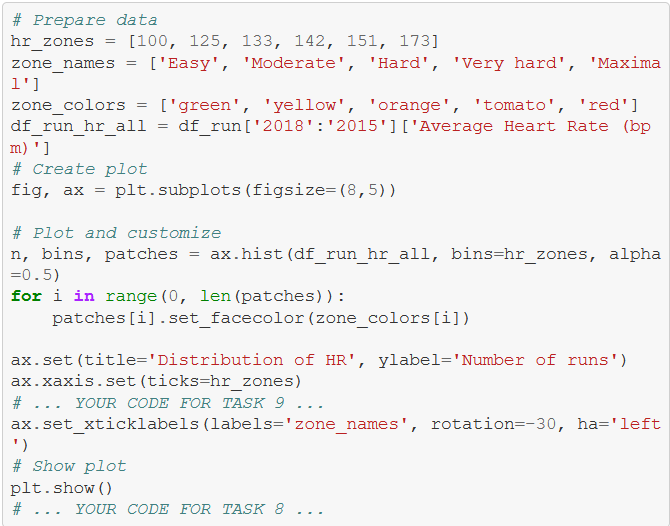
We are going to use statsmodels library to decompose the weekly trend.



**3.10 Training intensity**

Heart rate is a popular metric used to measure training intensity. Depending on age and fitness level, heart rates are grouped into different zones that people can target depending on training goals. A target heart rate during moderate-intensity activities is about 50-70% of maximum heart rate, while during vigorous physical activity it’s about 70-85% of maximum.

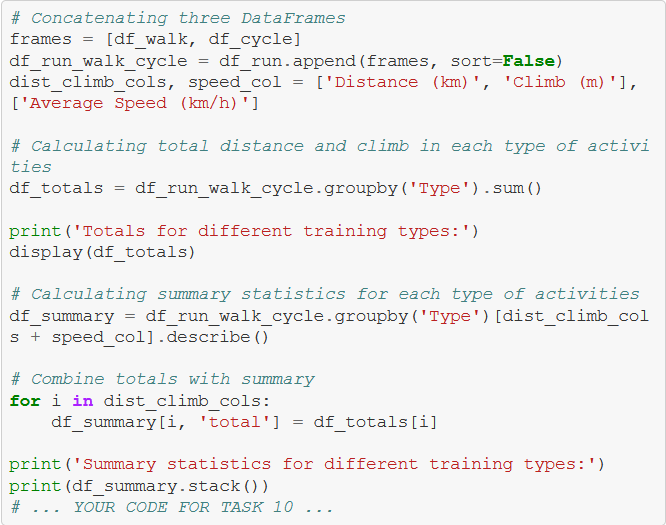
We'll create a distribution plot of my heart rate data by training intensity. It will be a visual presentation for the number of activities from predefined training zones.



**3.11 Detailed summary report**

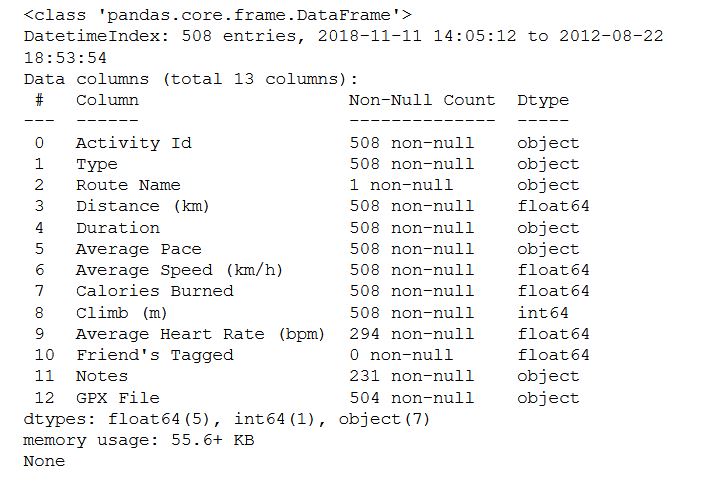
With all this data cleaning, analysis, and visualization, let's create detailed summary tables of my training.

To do this, we'll create two tables. The first table will be a summary of the distance (km) and climb (m) variables for each training activity. The second table will list the summary statistics for the average speed (km/hr), climb (m), and distance (km) variables for each training activity.

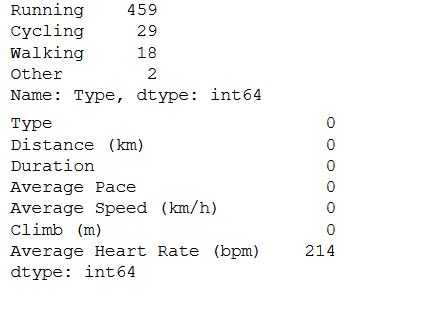
****

**IV. SAMPLE SCREENSHOTS AND OBSERVATIONS**

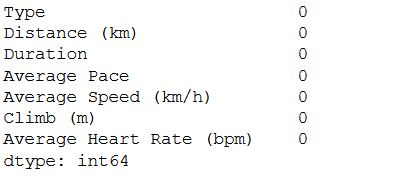
**4.1 Obtain and review raw data**



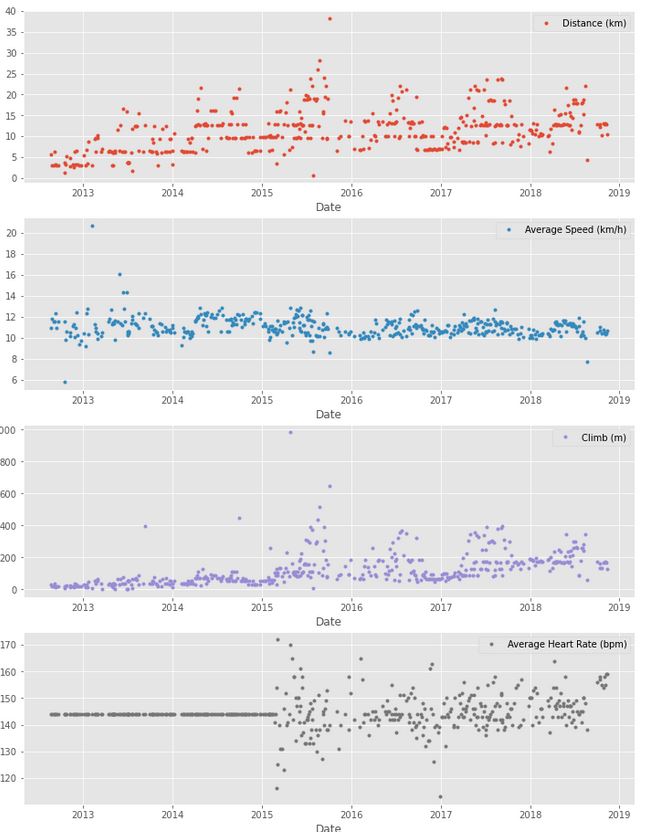
**4.2 Data preprocessing**

****

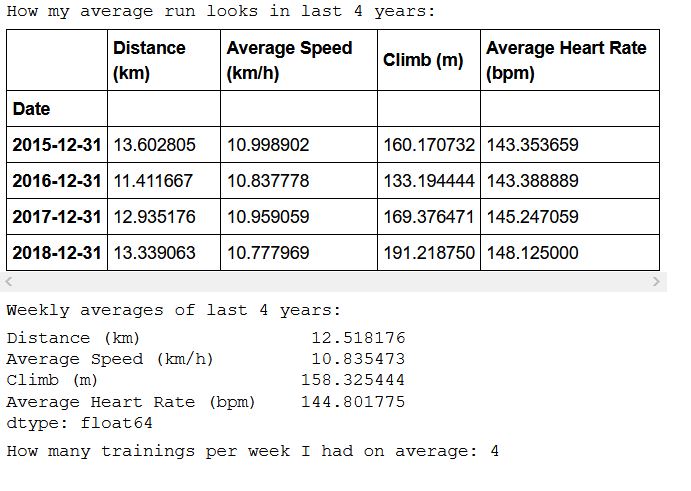
**4.3 Dealing with missing values**

****

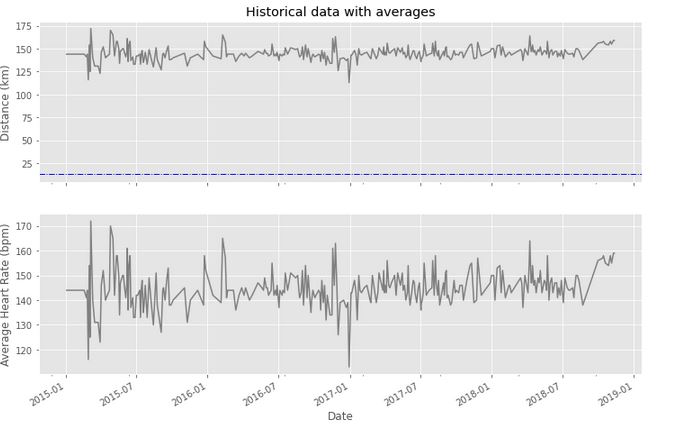
## 4.4 Plot running data

****

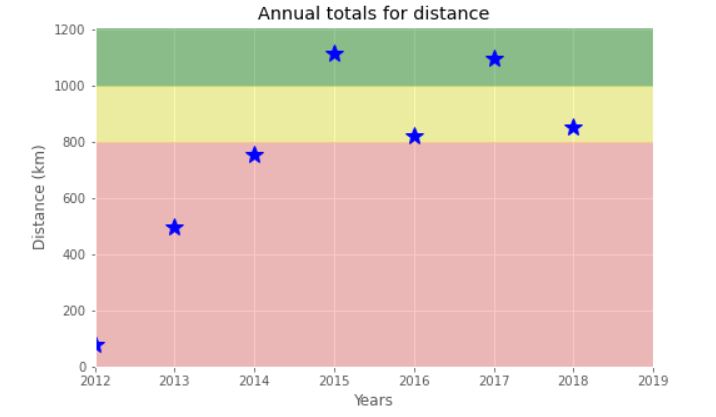
**4.5 Running statistics**

****

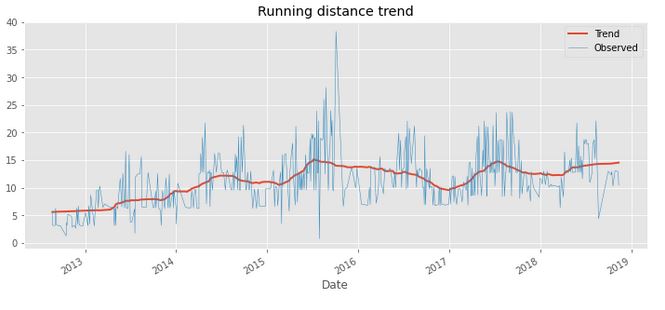
**4.6 Visualization with averages**



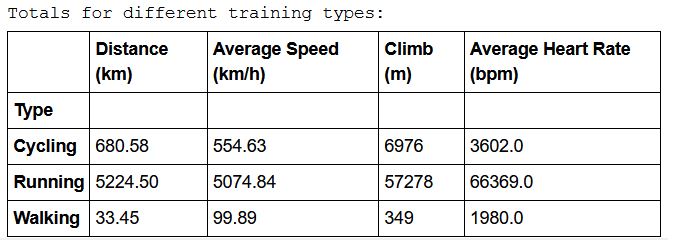
**4.7 Did I reach my goals?**

****

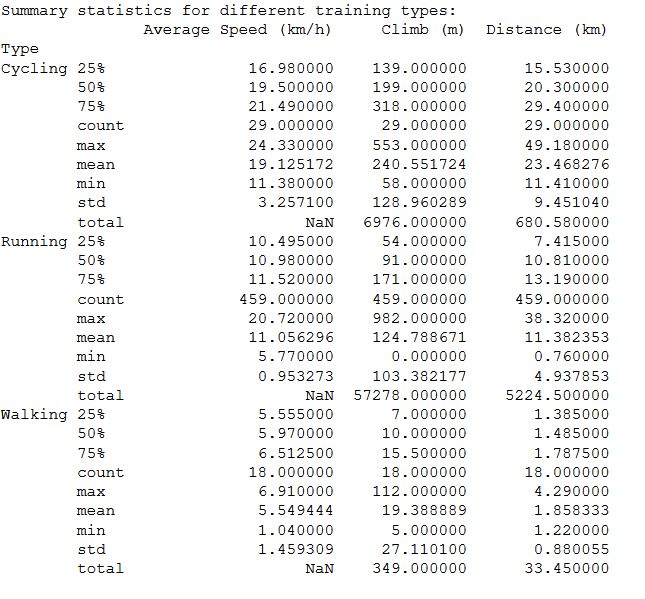
**4.8 Am I progressing?**

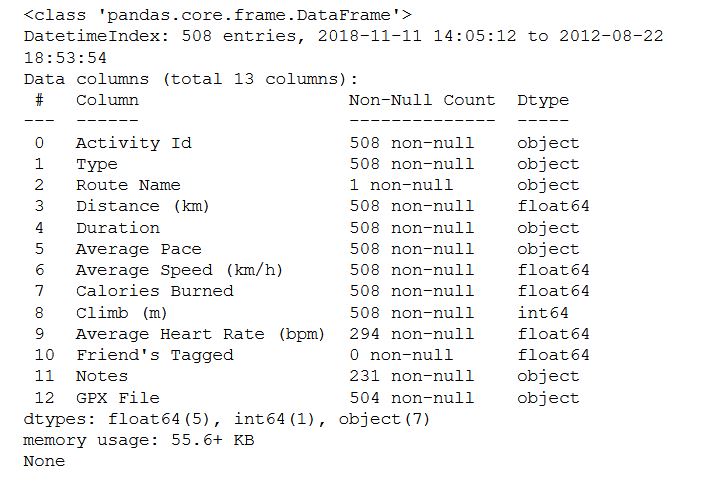
****

**4.9 Training intensity**

****

**4.10 Detailed summary report**

****



**V. CONCLUSION AND FUTURE SCOPE**

To wrap up, let’s pick some fun facts out of the summary tables and solve the last exercise.

These data (my running history) represent 6 years, 2 months and 21 days. And I remember how many running shoes I went through–7.

FUN FACTS

- Average distance: 11.38 km

- Longest distance: 38.32 km

- Highest climb: 982 m

- Total climb: 57,278 m

- Total number of km run: 5,224 km

- Total runs: 459

- Number of running shoes gone through: 7 pairs

The story of Forrest Gump is well known–the man, who for no particular reason decided to go for a "little run." His epic run duration was 3 years, 2 months and 14 days (1169 days). In the picture you can see Forrest’s route of 24,700 km.

FORREST RUN FACTS

- Average distance: 21.13 km

- Total number of km run: 24,700 km

- Total runs: 1169

- Number of running shoes gone through: ...

Assuming Forest and I go through running shoes at the same rate, figure out how many pairs of shoes Forrest needed for his run.

**V. REFERENCES**

**Data Collection**

The following website has been referred to obtain the input data and statistics:

<https://github.com/aravindpatel/cardioActivities/blob/main/cardioActivities.csv>

**Programming References**

The following websites have been referred for R coding and Shiny tutorials:

a. <https://datascienceplus.com/category/programming>

b.  [https://docs.python.org/3/](%20https:/docs.python.org/3/)

c. <https://bookdown.org/yihui/rmarkdown/document-templates.html>

d. <https://datascienceplus.com/map-visualization-of-covid19-across-world/>

e. <https://bookdown.org/yihui/rmarkdown/dashboards.html>

f. <https://rmarkdown.rstudio.com/lesson-12.html>

g. <https://bookdown.org/yihui/rmarkdown/cheat-sheets.html>

h. <http://www.htmlwidgets.org/showcase_leaflet.html>

i. <http://jeffgoldsmith.com/p8105_f2017/shiny.html>

j. <https://rmarkdown.rstudio.com/flexdashboard/using.html#page_icons>