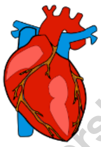
Wondershare
PDFelement

Architecture Design

Heart Disease Diagnostic Analysis



Wondershare
PDFelement

Revision Number - 1.1

Last Date of Revision - 05/02/2023

Arunesh Shukla

Wondershare
PDFelement

Document Control

Date	Version	Description	Author
30/01/2023	1.0	Introduction, Architecture, Deployment	Arunesh Shukla
05/02/2023	1.1	Final Revision	Arunesh Shukla

Wondershare
PDFelement

Wondershare PDFelement

Contents

Wondershare PDFelement

Wondershare PDFelement

1. Introduction

1.1 What is Architecture Design Document?

Any software needs the architectural design to represent the design of the software. IEEE defines architectural design as “the process of defining a collection of hardware and software components and their interfaces to establish the framework for the development of a computer system.” The software that is built for computer-based systems can exhibit one of these many architectures.
Each style will describe a system category that consists of:

- A set of components (eg: a database, computational modules) that will perform a function required by the system.
- The set of connectors will help in coordination, communication, and cooperation between the components.
- Conditions that how components can be integrated to form the system.
- Semantic models help the designer to understand the overall properties of the system.

Wondershare PDFelement

1.2 What is Scope?

Architecture Design Document (ADD) is an architectural design process that follows a step-by-step refinement process. The process can be used for designing data structures, required software architecture, source code and ultimately, performance algorithms. Overall, the design principles may be defined during requirement analysis and then refined during architectural design work.

Wondershare PDFelement

2. Architecture

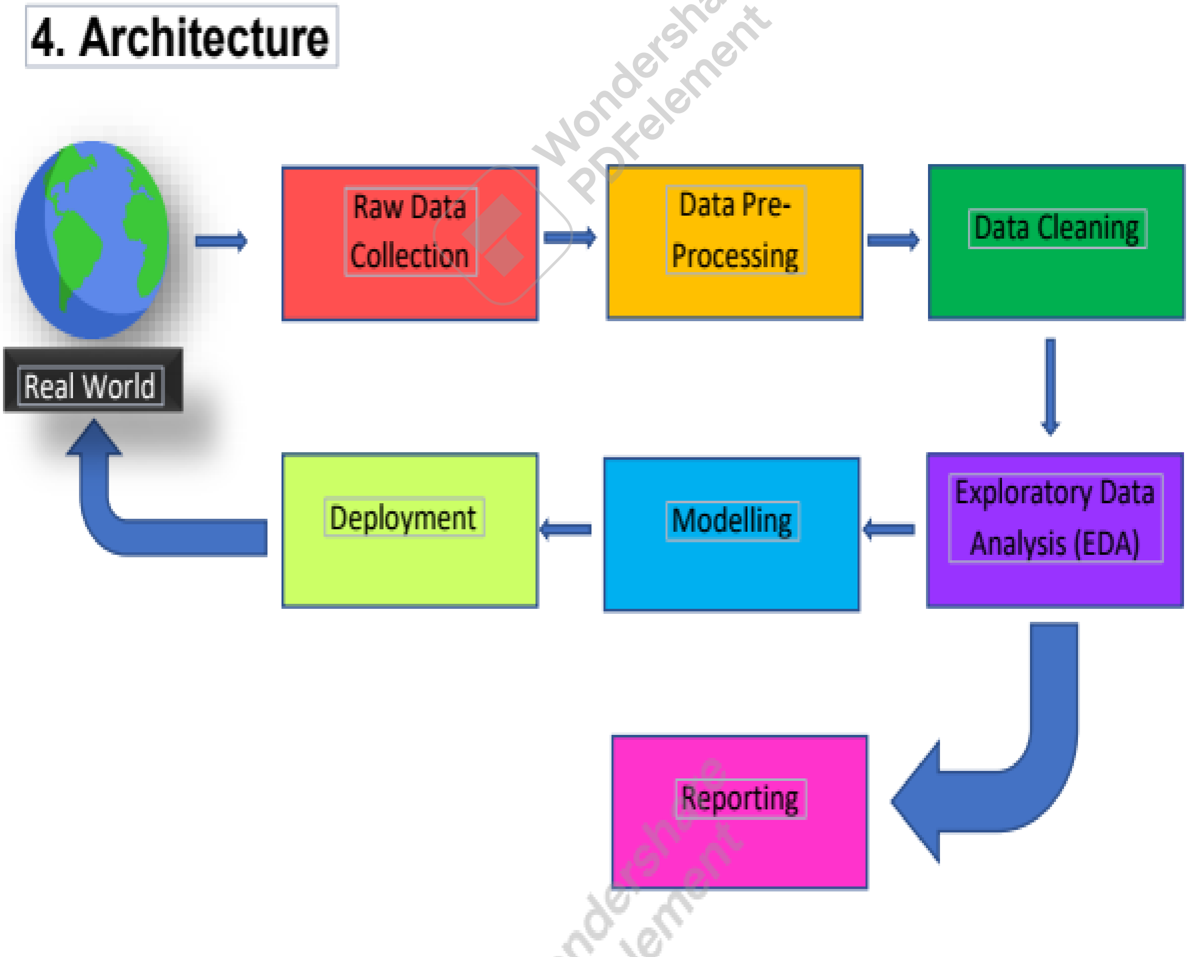
2.1 Pandas

Pandas (https://pandas.pydata.org/) is an open-source library that provides high-performance, memory-efficient, easy-to-use data structures, as well as support for data manipulation and analysis for Python.

The core pandas' data structures are Series and DataFrame objects

Pandas is built on top of two core Python libraries—matplotlib for data visualization and NumPy for mathematical operations.

Pandas acts as a wrapper over these libraries, allowing you to access many of matplotlib's and NumPy's methods with less code.



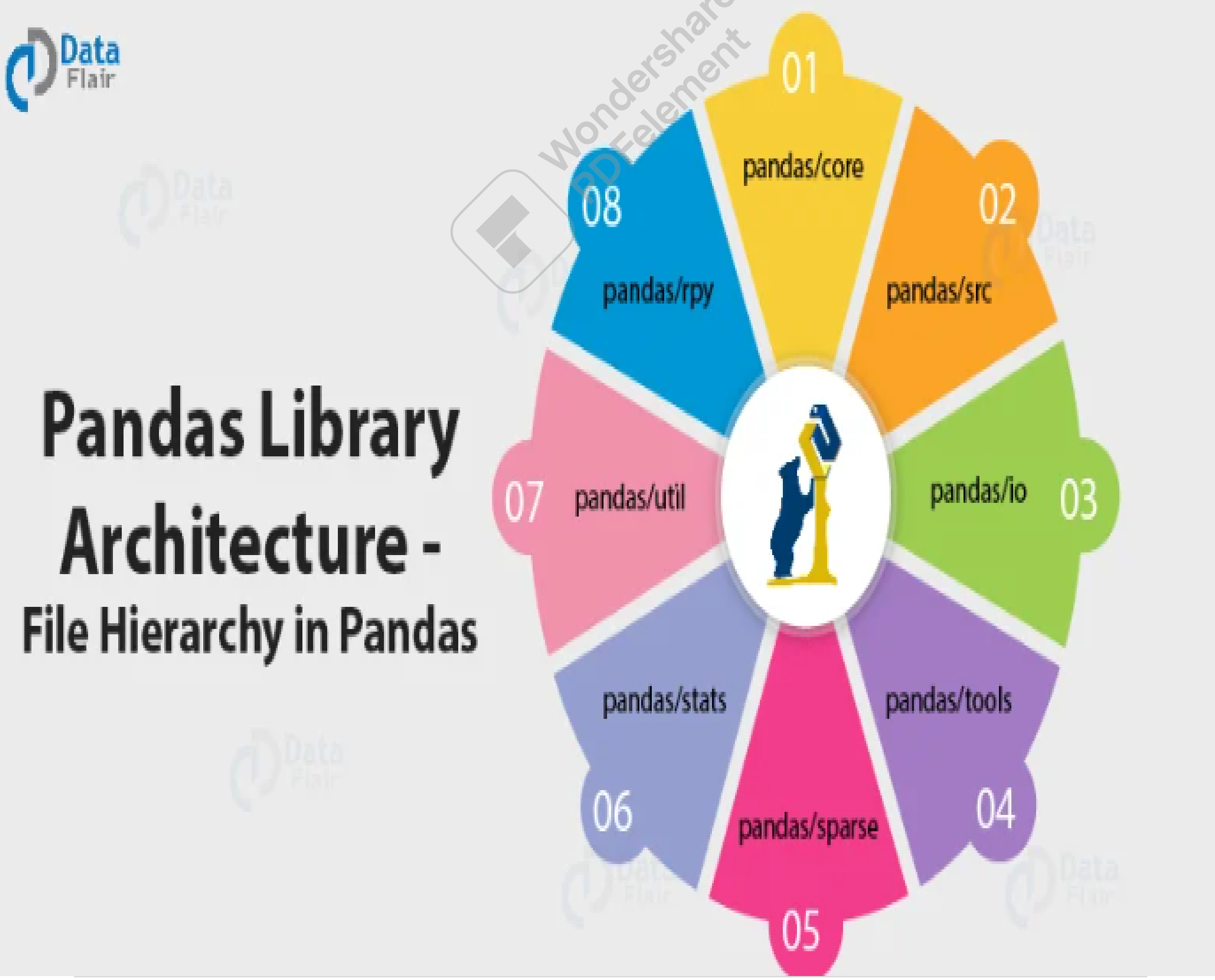
Wondershare PDFelement

Wondershare PDFelement

2.2 Components of Pandas Architecture

pandas/core
In Pandas library architecture, this part consists of basic files about the data structures present within the library. For examples, data structures – Series and DataFrames. There are various Python files within the core. The most important of them being:

api.py: Important key modules which will be used later are imported using these files.
base.py: This will provides the base for all the other classes present, like PandasObject and StringMIxin.
common.py: It controls the common utility methods which help in handling various data structures.
config.py: This helps to handle configurable objects found throughout the package.



Wondershare PDFelement

Wondershare
PDFelement

3. Deployment

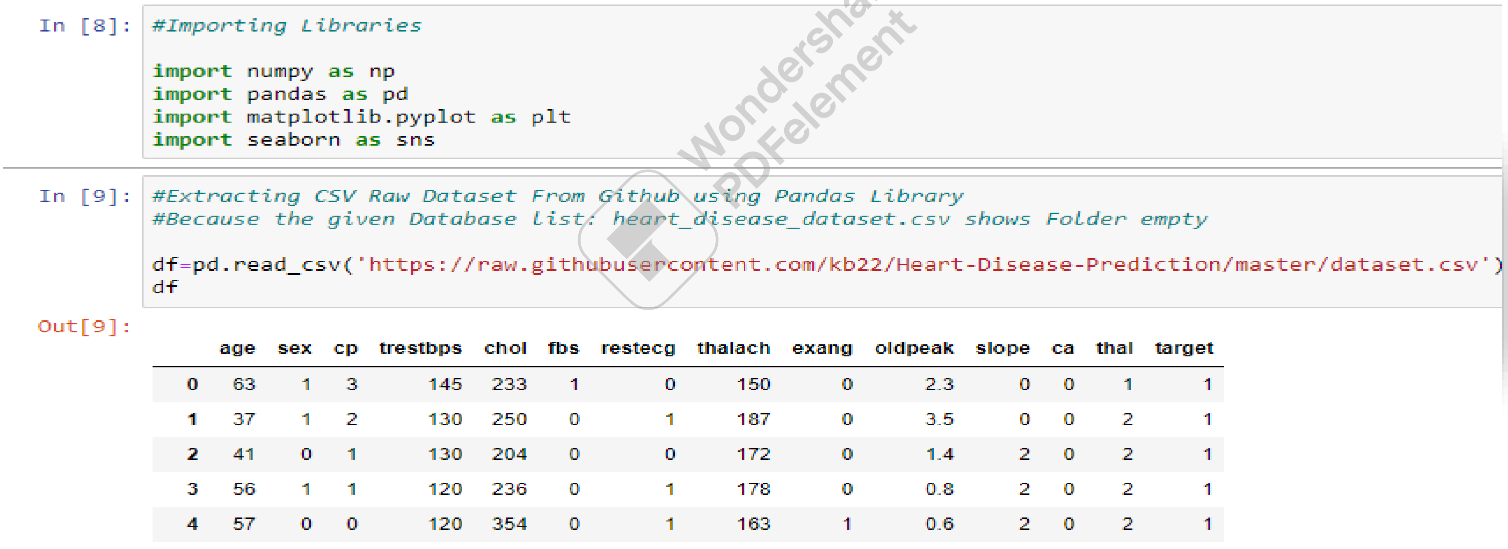
3.1 Pandas Deployment

Pandas is a fast, powerful, flexible, and easy to use open-source data analysis and manipulation tool. It is built on top of the Python programming language. Pandas is usually used in conjunction with Jupyter notebooks, making it more powerful and efficient for exploratory data analysis.

3.2 Publish datasets and reports from Pandas

Pandas stores the read data in a data structure called a Data Frame. According to the official documentation, a data frame is a two-dimensional, size-mutable, potentially heterogeneous tabular data structure that also contains labeled axes (rows and columns).It is similar to a 2D array in Python. By “size-mutable”, we mean that we can modify the size of the data frame at any time. By “heterogeneous”, we mean that it can have data of different types.

Wondershare
PDFelement

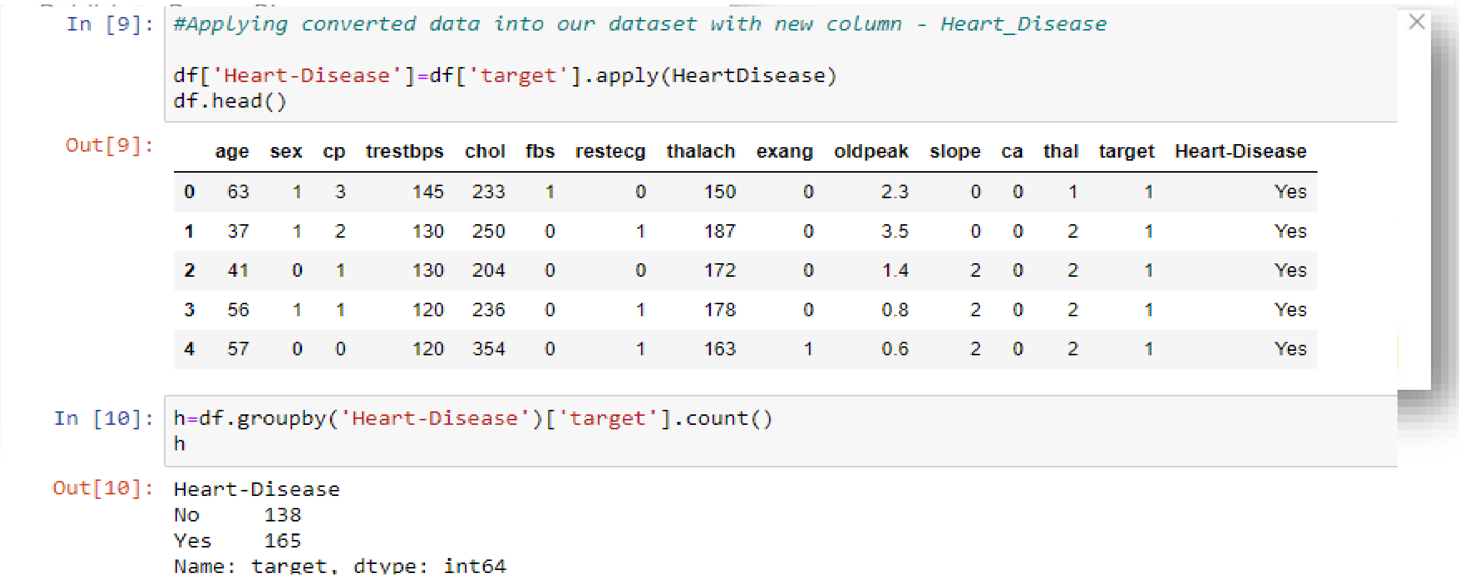


```
In [8]: #Importing Libraries

        import numpy as np
        import pandas as pd
        import matplotlib.pyplot as plt
        import seaborn as sns
```

```
In [9]: #Extracting CSV Raw Dataset From Github using Pandas Library
        #Because the given Database List: heart_disease_dataset.csv shows Folder empty

        df=pd.read_csv('https://raw.githubusercontent.com/kb22/Heart-Disease-Prediction/master/dataset.csv')
        df
```

Out[9]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1



```
In [9]: #Applying converted data into our dataset with new column - Heart_Disease

        df['Heart-Disease']=df['target'].apply(HeartDisease)
        df.head()
```

Out[9]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target	Heart-Disease
0	63	1	3	145	233	1	0	150	0	2.3	0	0	1	1	Yes
1	37	1	2	130	250	0	1	187	0	3.5	0	0	2	1	Yes
2	41	0	1	130	204	0	0	172	0	1.4	2	0	2	1	Yes
3	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1	Yes
4	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1	Yes

```
In [10]: h=df.groupby('Heart-Disease')['target'].count()
         h
```

```
Out[10]: Heart-Disease
         No      138
         Yes     165
         Name: target, dtype: int64
```