### BPP Business School Coursework Cover Sheet

Please use this document as the cover sheet of for the 1st page of your assessment.

Please complete the below table – the grey columns

|  |  |
| --- | --- |
| Module Name | **Applied Modelling and Visualization** |
| Student Reference Number  (SRN) | **BP0274581** |
| Assessment Title | **Skywards International Airlines Consultancy Report** |

Please complete the yellow sections in the below declaration:

**BP0274581**

Declaration of Original Work:

I hereby declare that I have read and understood BPP’s regulations on plagiarism and that this is my original work, researched, undertaken, completed and submitted in accordance with the requirements of BPP School of Business and Technology.

The word count, excluding contents table, bibliography and appendices, is **993** words.

Student Reference Number:

Date: 25-01-2024

By submitting this coursework you agree to all rules and regulations of BPP regarding assessments and awards for programs.

Please note that by submitting this assessment you are declaring that you are fit to sit this assessment.

BPP University reserves the right to use all submitted work for educational purposes and may request that work be published for a wider audience.

**Skywards International Airlines Consultancy Report**

**Table of Contents**

1. Introduction 5
   1. Background 5
   2. Purpose Statement 5
   3. Scope 5
2. SDLC for Skywards Airlines Project 5
   1. Requirements 5
      1. Non-functional Requirements 6
         1. Why Google Collab? 6
         2. Why Python? 6
      2. Functional Requirements 6
   2. Design 6
      1. Flowchart 7
      2. Pseudocode 8
   3. Implementation 8
      1. Libraries 8
         1. Pandas 9
         2. Matplotlib.pyplot 9
         3. Pygwalker 11
         4. Seaborn 12
      2. Import, Clean and Prepare Data 12
         1. Import 12
         2. Clean 13
            1. Handling Missing Values 13
            2. Drop Records with Missing Data 14
            3. Duplicate Rows 16
            4. Remove Outliers 17
      3. Exploratory Data Analysis 20
      4. Visualization 21
   4. Verification 22
      1. Unit Testing 23
      2. Functional Testing 23
   5. Maintenance 23
3. Appendix 1 – Code 24
4. References 25
5. **Introduction**
   1. **Background**

The Skywards International Airlines Corporation holds a large number of data about passengers as a large airline company. This data includes different parameters including the satisfaction of the customers after experiencing the Skywards Airlines services.

* 1. **Purpose Statement**

This Data Analytics Consultancy Report aims to analyze the data provided by Skywards Airlines Corporation in the SKYWARDS\_DATA\_CW3.csv file. Specifically, the data about the satisfaction of the passengers will be cleansed, transformed, and loaded, and then Exploratory Data Analysis on the data will be performed. Lastly, visualization and data modelling techniques will be used to analyze the current situation of data leading to future predictions.

* 1. **Scope**

The scope of the report includes discussing the following aspects of programming solutions to analyze the provided passenger data:

* Python code that can assist in uploading CSV data file
* Extraction of data
* Cleaning the data
* Exploratory Data Analysis
* Visualization for analyzing trends in airline data
* Analytical Model

1. **SDLC for Skywards Airlines Project**
   1. **Requirements**

The first stage allows the collection of the requirements by reviewing the problem statement. There are two types of requirements as below (White, 2010):

* + 1. **Non-functional Requirements**

The platform suggested to be used for this project is Google Collab. Besides, the suggested language is Python.

* + - 1. **Why Google Collab?**

Google Collab is recommended as it (Pessoa et al., 2018):

* Allows a fully configured environment
* Free of charge
* Works in the same way on different hardware systems
* Easy to detect errors while running the code
* No dependency on the hardware resources
  + - 1. **Why Python?**

The Python language is being used in this project as it has many benefits (Nitnaware, 2019):

* Easily readable
* User-friendly and simple language
* Open source/free
* Compatible with many platforms
* Supports a large number of libraries
  + 1. **Functional Requirements**
* Design a solution to upload Excel file
* Extract data from the Excel file
* Cleaning the Data frame
* Exploratory data analysis
* Visualization through graphs
  1. **Design**

The programming or coding is being done in this phase (Memon et al., 2018).

* + 1. **Flowchart**

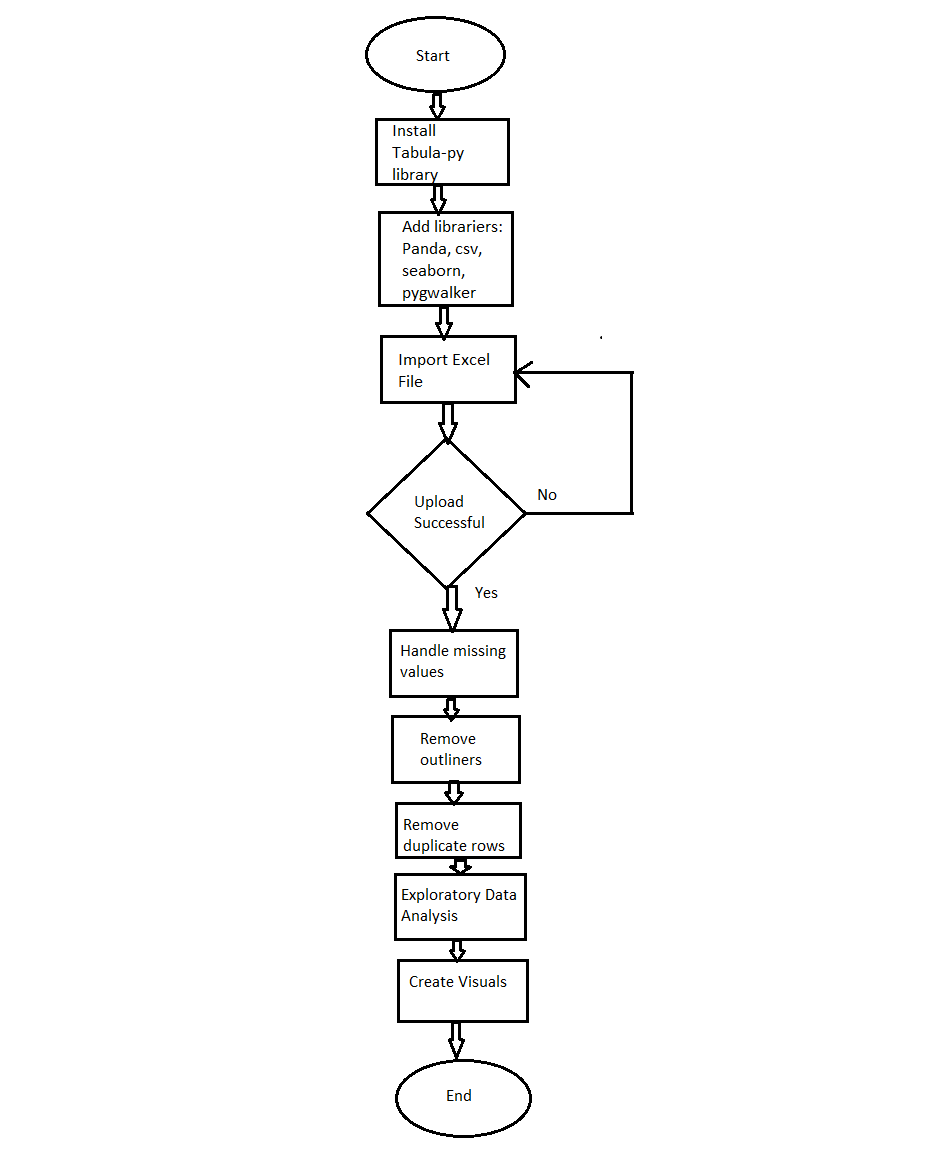


Figure 1.

* + 1. **Pseudocode**

1. Start
2. Install libraries
3. Import libraries
4. Import CSV Data File as a data set
5. If the Upload is successful, extract the Data set from the Excel file.
6. If not, import the Excel file again.
7. Handle missing values
8. Drop the duplicate values
9. Remove outliers
10. Exploratory data analysis to calculate average and percentage values
11. Create Visuals of Exploratory data analysis results
12. Analytical Modeling
13. End
    1. **Implementation**
       1. **Libraries**

Firstly, the installation of tabula and pygwalker is done on Google Collab notebook by using the pip command as in Figure 2:

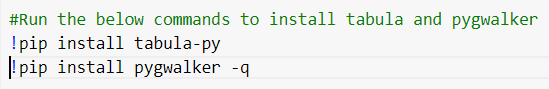


Figure 2.

Afterwards, import the libraries below:

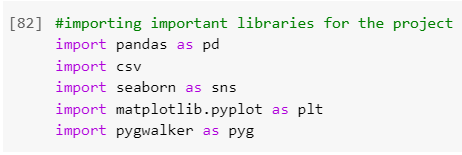


Figure 3.

* + - 1. **Pandas**

This library helps perform analysis of data in Python by manipulating the data. Pandas render data structures/functions for data analysis. The main purposes for using Pandas in this project are (Nitnaware, 2019):

* Data Cleaning
* Data Transformation
* Data Analysis by using various mathematical functions
* Data input/output
* Data visualization
  + - 1. **Matplotlib.pyplot**

This Python library is used to make visualizations such as below:

* Graphs
* Plots
* Charts

In the current project, ‘matplotlib.pyplot’ has been used for the following purposes:

* **Data Visualization:** Various types of exploratory data analysis pie charts have been created in this report to visualize the results of analysis by using this library. For instance, Figure 4 shows the pie chart:



Figure 4.

* **Customization:** By using the ‘matplotlib.pyplot’ library, customized plots/charts have been drawn in different colors by using the desired labels as shown in Figure 5.

**Input:**

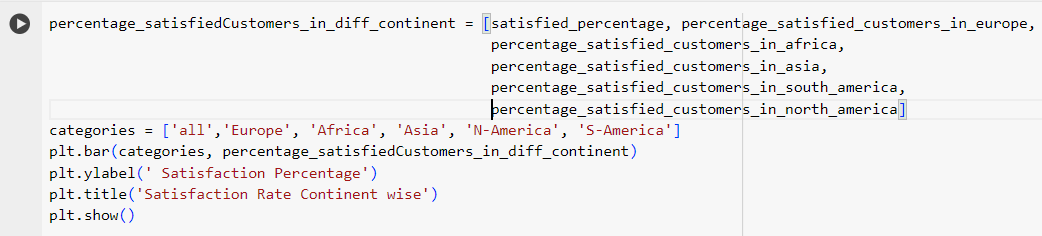


Figure 5.

**Output:**

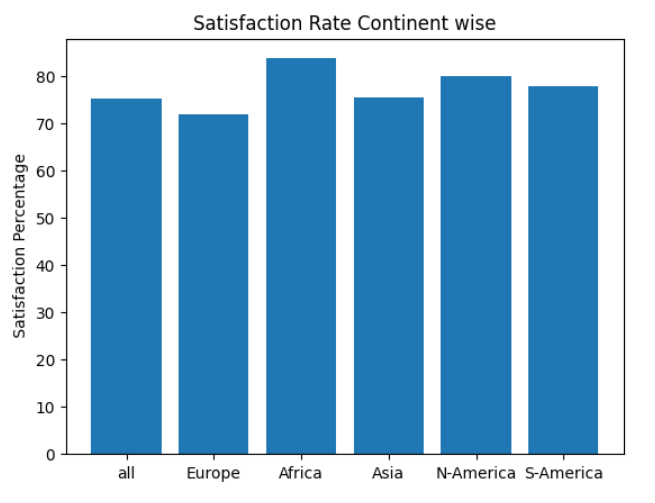


Figure 6.

* + - 1. **Pygwalker**

This library allows one to view the raw data in a Data frame. In the current report, the data frame extracted from Excel file can be visualized by using pygwalker functions.

**Input:**

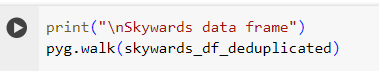


Figure 7.

**Output:**

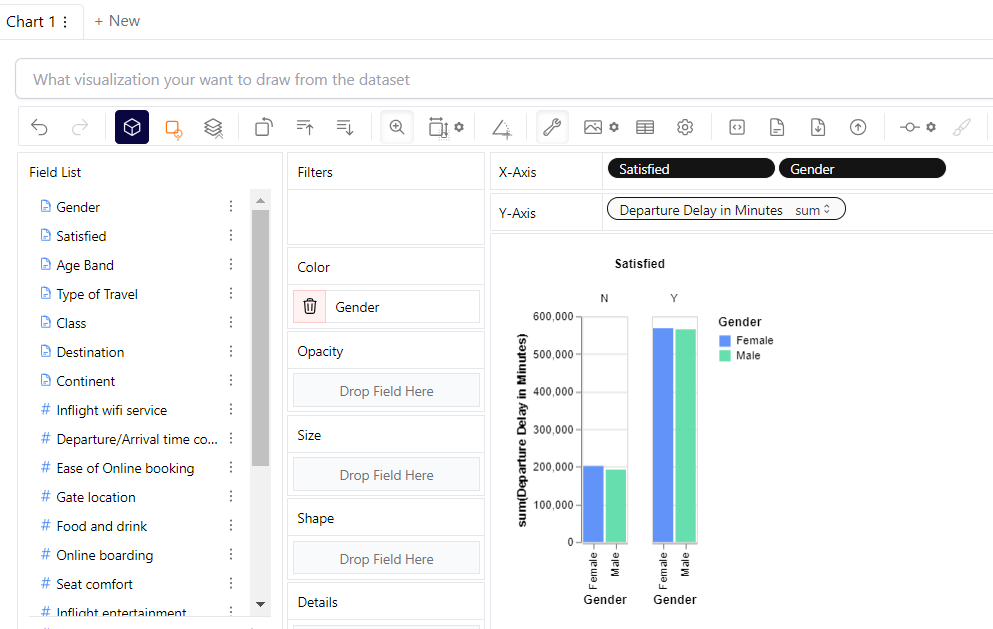


Figure 8.

* + - 1. **Seaborn**

Seaborn is a popular Python data visualization library based on Matplotlib. It provides a high-level interface for creating informative and attractive statistical graphics. Seaborn is particularly useful for creating complex visualizations with concise syntax (Waskom, 2021). Figure 29 and 30 represents the use of seaborn library in this report.

* + 1. **Import, Clean and Prepare Data**
       1. **Import**

Firstly, the read\_csv function has been used in the project to extract data from the excel file as shown in Figure 4.

**Input:**

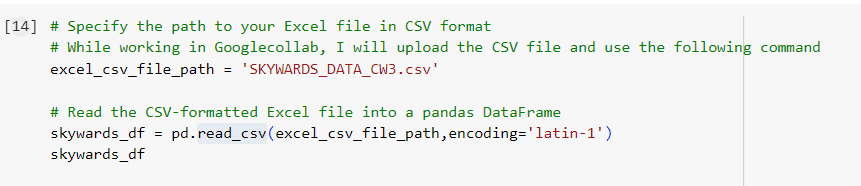


Figure 9.

**Output:**

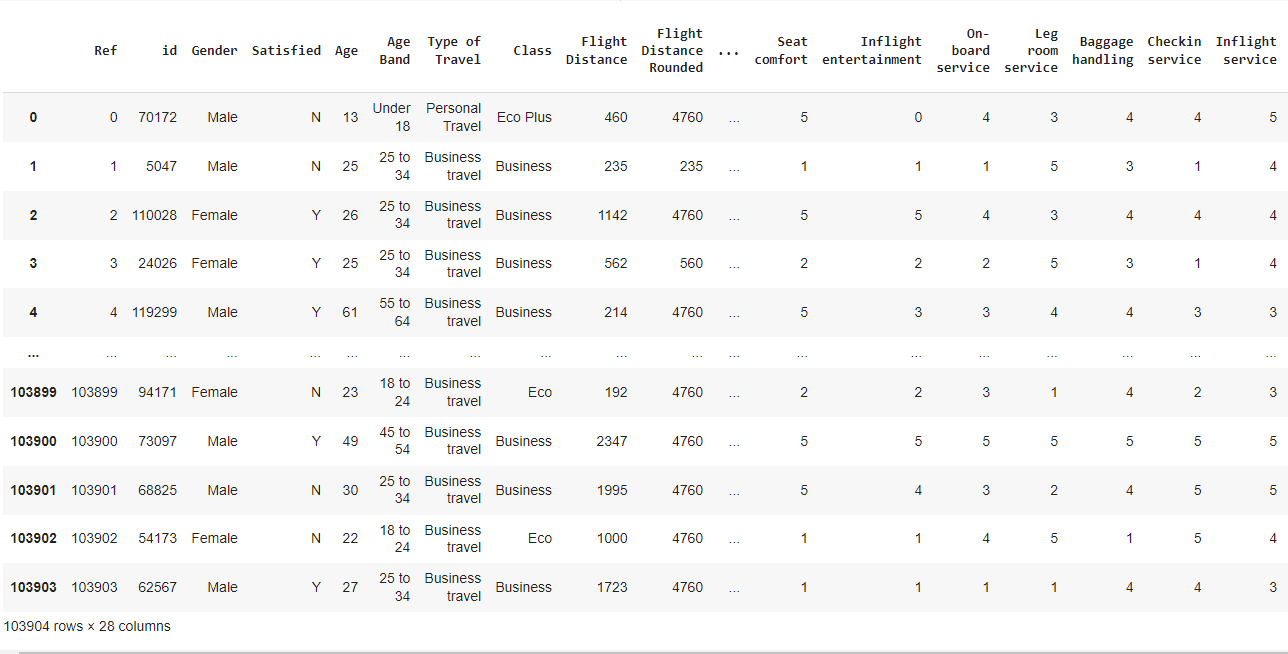


Figure 10.

* + - 1. **Clean**
         1. **Handling Missing Values**

**Input:**



Figure 11.

**Output:**

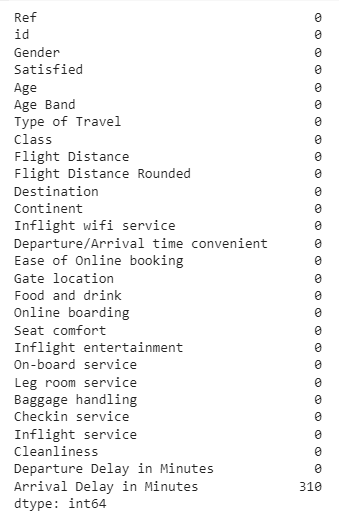


Figure 12.

* + - * 1. **Drop records with missing data**

**Input:**

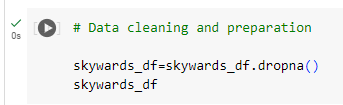


Figure 13.

**Output:**



Figure 14.

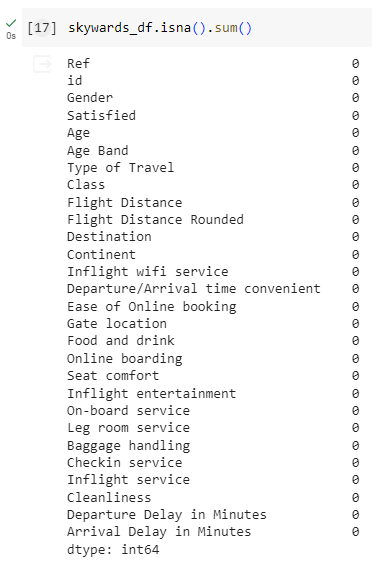


Figure 15.

* + - * 1. **Duplicate Rows**

**Input:**

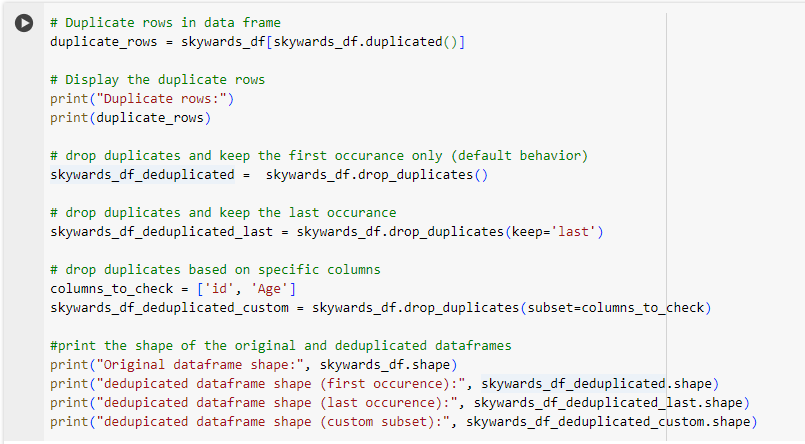


Figure 16.

**Output:**

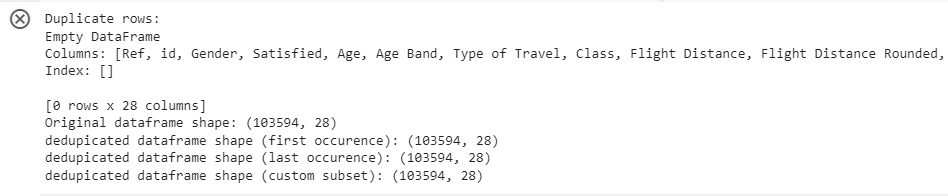


Figure 17.

* + - * 1. **Remove Outliers**

Firstly, the data frame is described before removing the outliers:

**Input:**

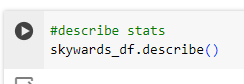


Figure 18.

**Output:**

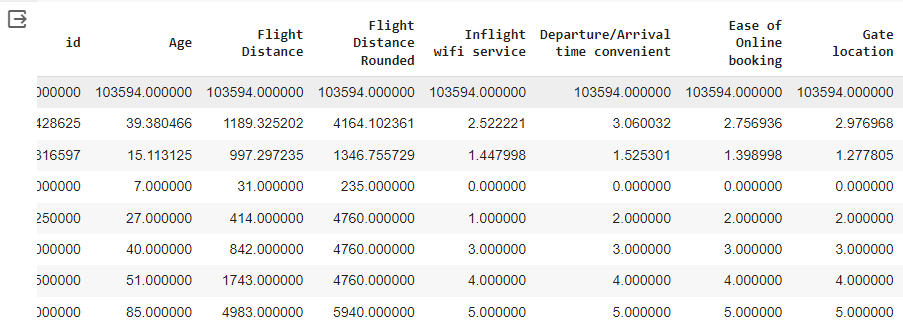


Figure 19.

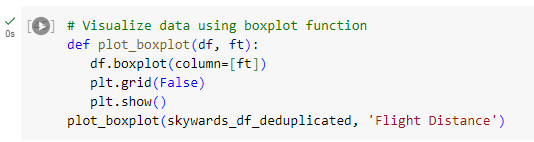
**Input:**  


Figure 20.

**Output:**

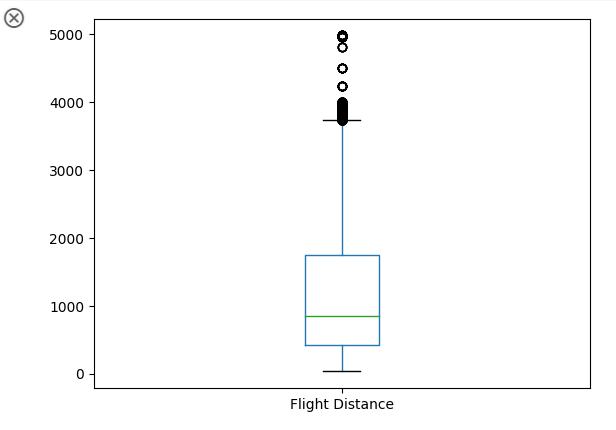


Figure 22.

The outliers can be removed as follows:

**Input:**

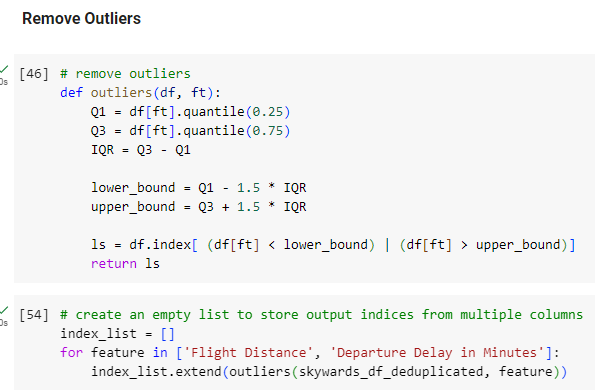


Figure 23.

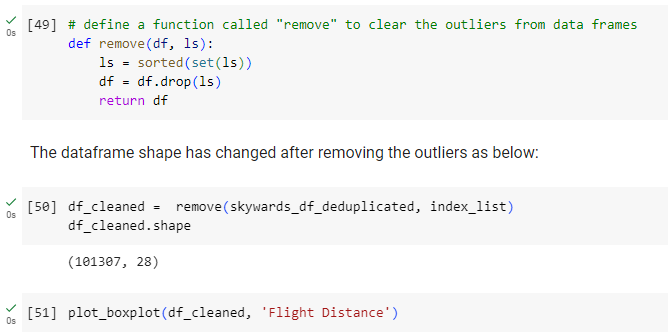


Figure 24.

The visualization after removing the outliers:

**Output:**

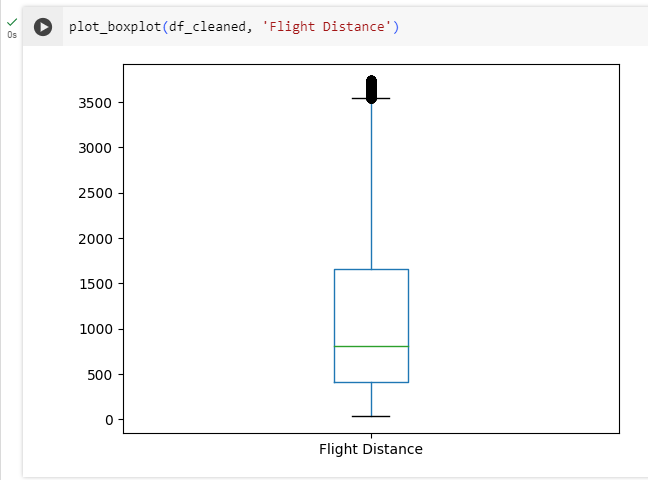


Figure 25.

* + 1. **Exploratory Data Analysis**

The satisfied percentage of different male, female, total passengers’ data and the passengers from different continents has been calculated as below:

**Input:**

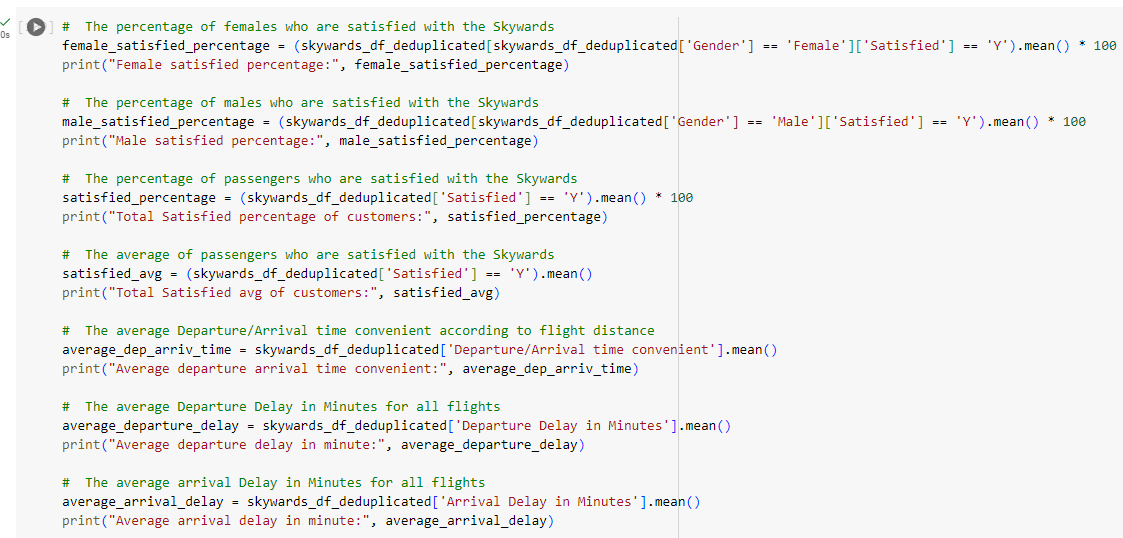


Figure 26.



Figure 27.

**Output:**

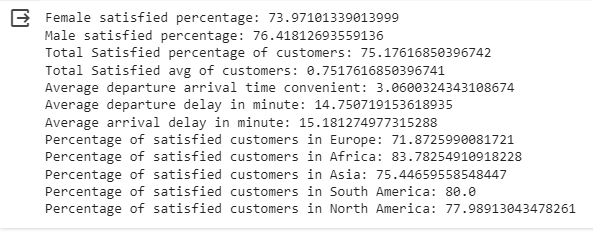


Figure 28.

* + 1. **Visualization**

The visualization with the help of heat map has been done to see the hot and cold point of the passengers and airlines data as follows:

**Input:**

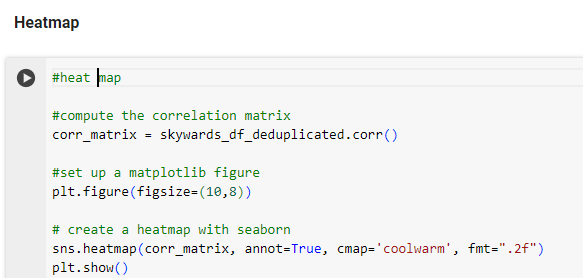


Figure 29.

**Output:**

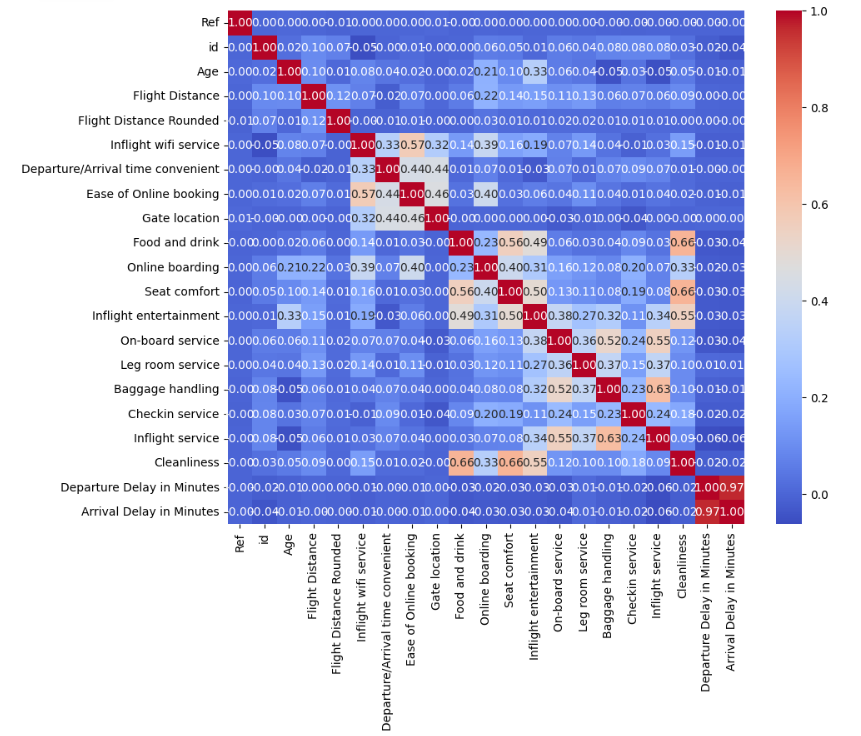


Figure 30.

* 1. **Verification**

The quality assurance engineer verifies the code in this stage. Besides, the developer performs unit testing in the environment (JiravaÚstav, 2004).

* + 1. **Unit testing**

The developer performs this testing at his end after finishing the coding (Ostan, 2003).

**Test case 1:**

Verify there is no error when the code is executed.

* **Steps to perform:** Run the programming file with the .ipynb extension.
* **Expected Result:** There shouldn’t be any error when the program is executed.
* **Actual Result:** There is no error in executing the program.
  + 1. **Functional Testing**

This type of testing ensures that the functional requirements of the code are met successfully (Lawanna, 2012).

* 1. **Maintenance**

This involves focusing on the following aspects in the future (Memon et al., 2018):

* Bug fixing
* Upgrade
* Enhancement with time

1. **Appendix 1 – Code**

The programming solution is as follows:

****

1. **References**

* JiravaÚstav, P. (2004) System Development Life Cycle. Available at: <https://www.researchgate.net/publication/44982277_System_development_life_cycle>
* Lawanna, A. (2012) ‘The Theory of Software Testing’, *ITS Journal - Intelligent Transportation Systems Journal.*
* Memon, S., Maham, U. (2018) Importance of Software Life Cycle in Development of Quality Software.
* Nitnaware, R. (2019) ‘Basic Fundamental of Python Programming Language and The Bright Future’, *A Peer-Reviewed Journal About.*
* Ostan, M. (2003) Unit testing: Test early, test often.
* Pessoa, T., Medeiros, R., Nepomuceno, T., Filho, P. (2018) Performance Analysis of Google Colaboratory as a Tool for Accelerating Deep Learning Applications.
* Waskom, M. (2021) ‘Seaborn: statistical data visualization’, *The Journal of Open Source Software.*
* White, A. (2010) ‘A review of the impact of requirements on software project development using a control theoretic model’, *Journal of Software Engineering and Applications*, 03(09), pp. 852–857. doi:10.4236/jsea.2010.39099.