

# **Flight Registration Made Easy with tkinter and FlightsRegister**

**Introduction to Flights Register**

**Using tkinter for FlightsRegister**

**Creating Direct Flights**

**Creating Multi-Leg Flights**

**GUI Functionalities**

**TDD Approach**

**Future Developements**

**Conclusion**

# Introduction to Flights Register

The Flights Register is a vital component of the Flight Management System. It serves as a centralized database for storing and managing flight information, including flight numbers, departure and arrival times, airline information, and other relevant data. The Flights Register plays a crucial role in ensuring that all flights are properly tracked and managed.

With the Flights Register, users can easily search for specific flights, view flight schedules, and manage flight itineraries. The system also provides real-time updates on flight status, allowing users to stay informed about any changes or delays. Overall, the Flights Register is an essential tool for any airline or aviation company looking to streamline their flight management operations.



# Introduction to Flights Register App

The Flights Register App is a software application that enables users to manage flight information with ease. It provides an intuitive interface for creating and managing flights, including direct and multi-leg flights. The app also includes GUI functionalities that allow users to interact with the software in a user-friendly manner.

The app's intuitive interface and seamless integration with the Flights Register make it an essential tool for anyone involved in flight management operations.

The Flights Register App was developed using the Test-Driven Development (TDD) approach, which ensures that the software is thoroughly tested throughout the development process.





## Using tkinter for FlightsRegister

For simplicity reasons we decided to implement our application GUI using tkinter. This, thanks to his guide book and to its friendly use, helped us to save time and build up a pretty easy but efficient interface. Even if the result is very minimal we plan to renew it in future developments.

Another choice that we made to save time for our first implementation of the application was to not implement a full integrated database and use only simple ways to store the data of our application. For this reason the first step to move forward would be to implement a fully integrated database that will help to store more info about airports and save historical datas of the flight register.



NEW FLIGHT  
AIRPORT DESTINATION

FORM NUMBER AIRCRAFT

✈️ 4380000 \$1400

→ LON 11/11 > 25,222

→ LON 11/11 > 12,90

CONFIRM AVER?

# Creating Direct Flights

Direct flights are flights that go directly from the departure airport to the destination airport without any stops in between. To create a direct flight using the `DirectFlight` class, follow these steps:

1. Insert the 5 digits code of the Flight
2. Insert the departure and destination airport acronym
3. Insert the duration of the flight in minutes
4. Use tkinter widget button to create an instance of the `DirectFlight` class for the registration system
5. Add the direct flight to the Flights Register using the `add_flight()` function and pressing the appropriate button

With these steps, you can create a flight within our flight registration system that will help you manage your flight data with ease.



# Creating Multi-Leg Flights

Multi-leg flights are flights that have one or more stops between the departure airport and the destination airport. To create a multi-leg flight using the MultiLegFlight class, follow these steps:

1. Insert the number of legs in the appropriate box and check the appropriate box to enable the insertion of MultiLeg flights
2. Fill all the fields taking into consideration that the correct order of the airport is departure - legs airport - destination airport
3. Create an instance of the MultiLegFlight class with the flight number, departure airport, list of stop airports, destination airport, and total duration in minutes
4. Add the multi-leg flight to the Flights Register using the add\_flight() function

With these simple steps, you can create multi-leg flights.

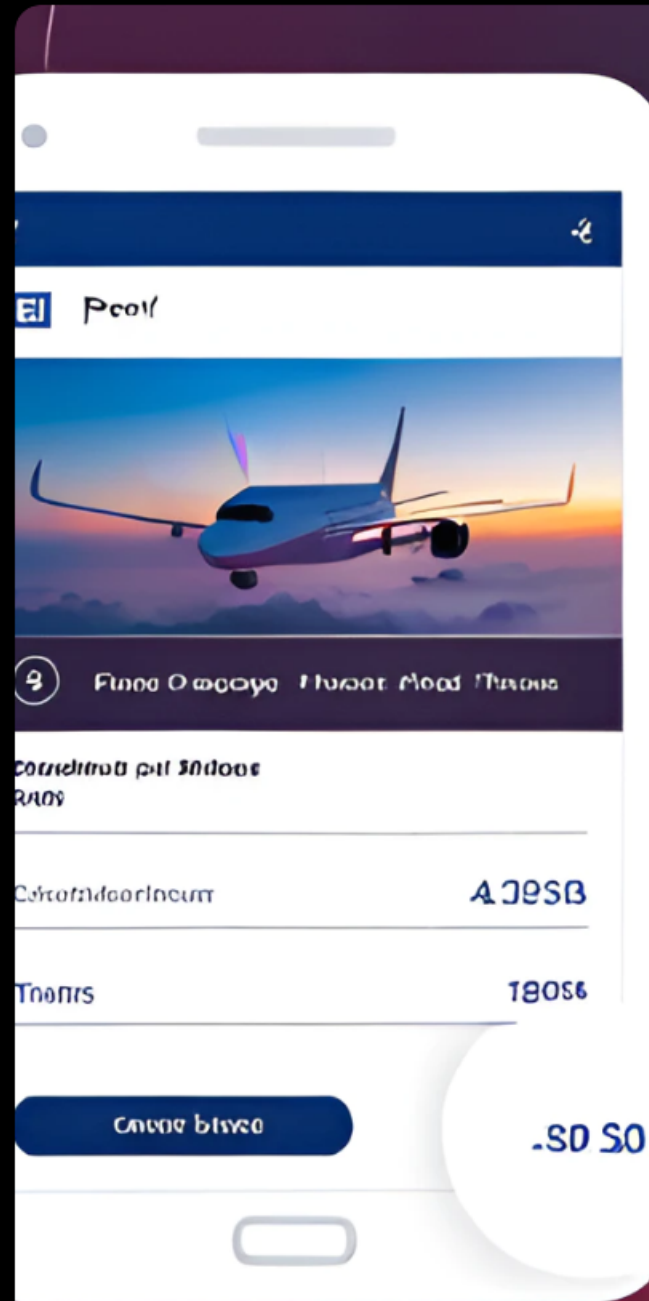




## Print the Flight Register report on file

The GUI application for managing civil air flights has a feature to print the flight register report on file. This report contains all the necessary information about the flights, including the flight number, departure and arrival airport, duration of the flight. The report is generated in a TXT format, which can be easily shared with other stakeholders such as airline staff, airport authorities, and regulatory bodies.

Generating the flight register report on file is a simple process that requires just a few clicks, the application generates the report and saves it with the specified name. The report can then be accessed and printed at any time, making it a convenient way to keep track of the flights and their details.



Print report on file



## Flight Code Search

The GUI application for managing civil air flights features a flight code search function that allows users to quickly locate specific flights within the system. This feature is particularly useful for airline staff who need to access information about a particular flight, such as its departure and arrival airport, flight duration.

To use the flight code search function, simply enter the flight code into the designated search field and click the search button. The application will then display all relevant information about the flight. This feature saves time and streamlines the process of accessing important flight information.



## Show All Flight Codes

The GUI application for managing civil air flights offers a feature that allows users to view all the flight codes in the system. This function is particularly useful for airline staff who need to access a complete list of flight codes for reference purposes.

To use this feature, simply navigate to the 'View All Flight Codes' option and click on it. The application will then display a comprehensive list of all the flight codes currently stored in the system. This feature saves time and provides easy access to important flight information.



# A Test-Driven Development Project

One of the fundamental aspects of this project is the Test-Driven Development methodology, or TDD for short. TDD involves writing tests before implementing the actual code. This approach offers numerous advantages, such as improved code quality, better test coverage, and easier maintenance.

By writing tests first, we ensured that the code we wrote meet the requirements and specifications set out for it. This helped to prevent bugs and errors from creeping into the codebase, which can be costly and time-consuming to fix later on. Additionally, TDD encouraged us to think about edge cases and potential issues that may arise, leading to more robust and resilient code.





## Future Developments

As we move forward with the development of our flight management application, there are several exciting features that we plan to implement. First and foremost, we aim to integrate real-time flight tracking data into our system. By doing so, users will be able to view the exact location of their flight in real-time, providing them with up-to-date information on arrival times and any potential delays.

In addition, we plan to introduce more info about passengers and crew members. This will streamline the booking process for users and provide them with a more seamless experience.

Another fundamental feature to introduce is a restyling of the GUI that for time deadlines and shortage resource reasons was made very minimal and with few functionalities. We also plan to integrate and expand our airport database to include more international airports, making our application accessible to users all over the world.





## Bibliography

The development of our Python project was inspired by the growing need for efficient management of civil air flights. In designing the 5 Flight Classes GUI application, we drew from various sources to ensure that our app is both user-friendly and effective. Our primary references include works on object-oriented programming, software engineering, and graphical user interface design. We also consulted several aviation databases and industry reports to ensure that our app meets the needs of modern-day air travel.

We are confident that our Python project will be a valuable asset to anyone who needs to manage civil air flights, and we hope that it will continue to evolve and improve over time.



## Conclusion

Flights Register is a powerful tool for managing flight data in the aviation industry. With its user-friendly interface and comprehensive features, Flights Register makes it easy to create and manage flight schedules, track flights in real-time, and generate reports on flight data.

Whether you're an airline operator, a travel agent, or an aviation enthusiast, Flights Register has something to offer. So why wait? Start using Flights Register today and take your flight management to the next level!

