**LAPORAN PRAKTIKUM**

**TEKNIK PEMROGRAMAN**

**CLEAN CODE SONARQUBE TEST**

MINGGU KE-10



NAMA: FAUZI ISMAIL

NIM: 241524042

KELAS: D4-1B

**PROGRAM STUDI SARJANA TERAPAN**

**TEKNIK INFORMATIKA**

**POLITEKNIK NEGERI BANDUNG**

**2025**

## DAFTAR ISI

[DAFTAR ISI 2](#_Toc196137394)

[https://github.com/mailvlous/teknikPemrograman/tree/main/Week8 3](#_Toc196137395)

[1. Studi Kasus, Post Sosial Media 3](#_Toc196137396)

[1. Text Post 3](#_Toc196137397)

[2. Image Post 3](#_Toc196137398)

[3. Polling Post(Generic Class) 4](#_Toc196137399)

[Wildcard 5](#_Toc196137400)

[Main Function 6](#_Toc196137401)

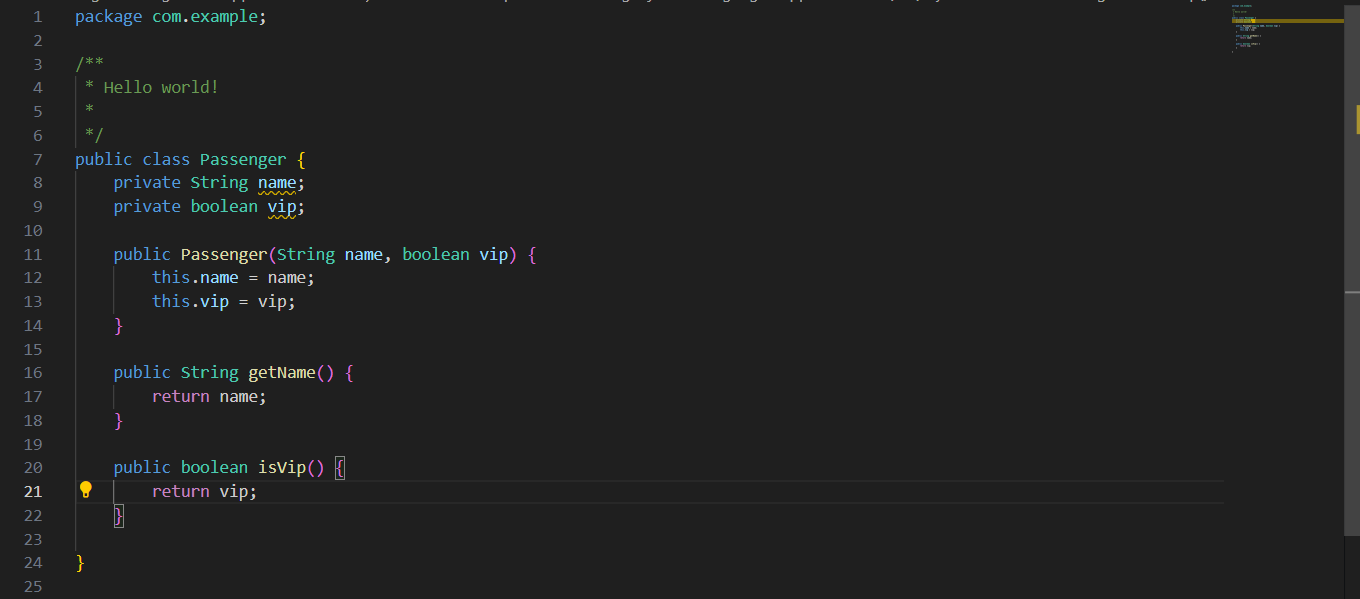
## <https://github.com/mailvlous/teknikPemrograman/tree/main/Week9/flight-management-app>

## JUNIT TEST DRIVEN DEVELOPMENT FLIGHT MANAGEMENT APPLICATION

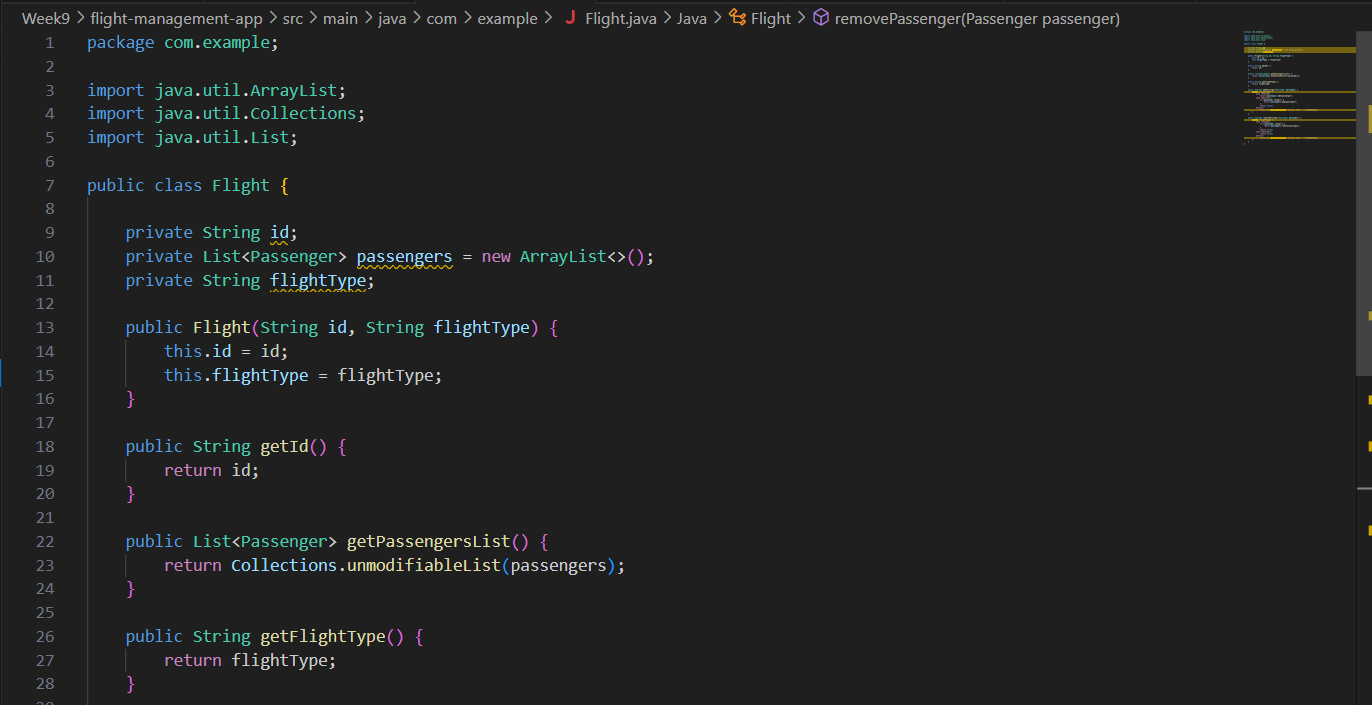
Pada program Flight Management App terdapat Test Driven Development dalam file AirportTest.java.

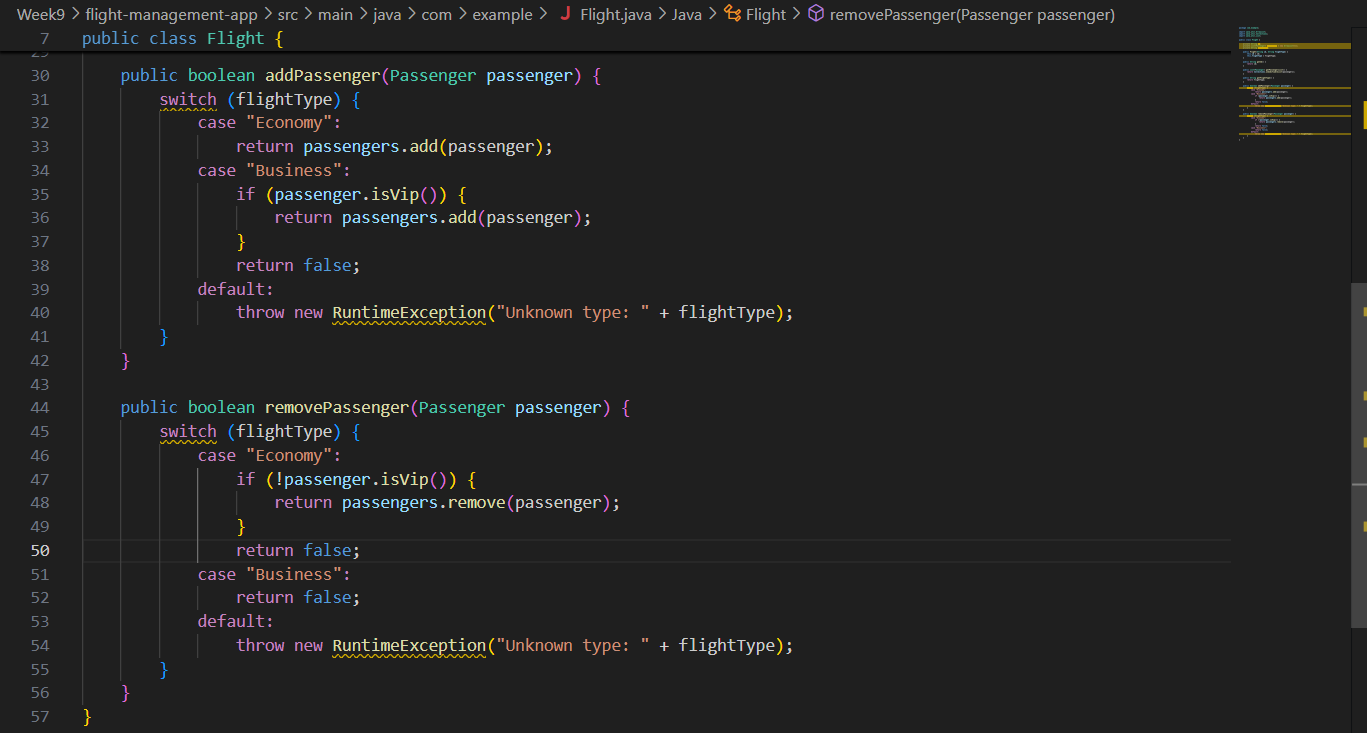
Terdapat 3 Class utama dalam Flight Management Application ini :

1. Passenger

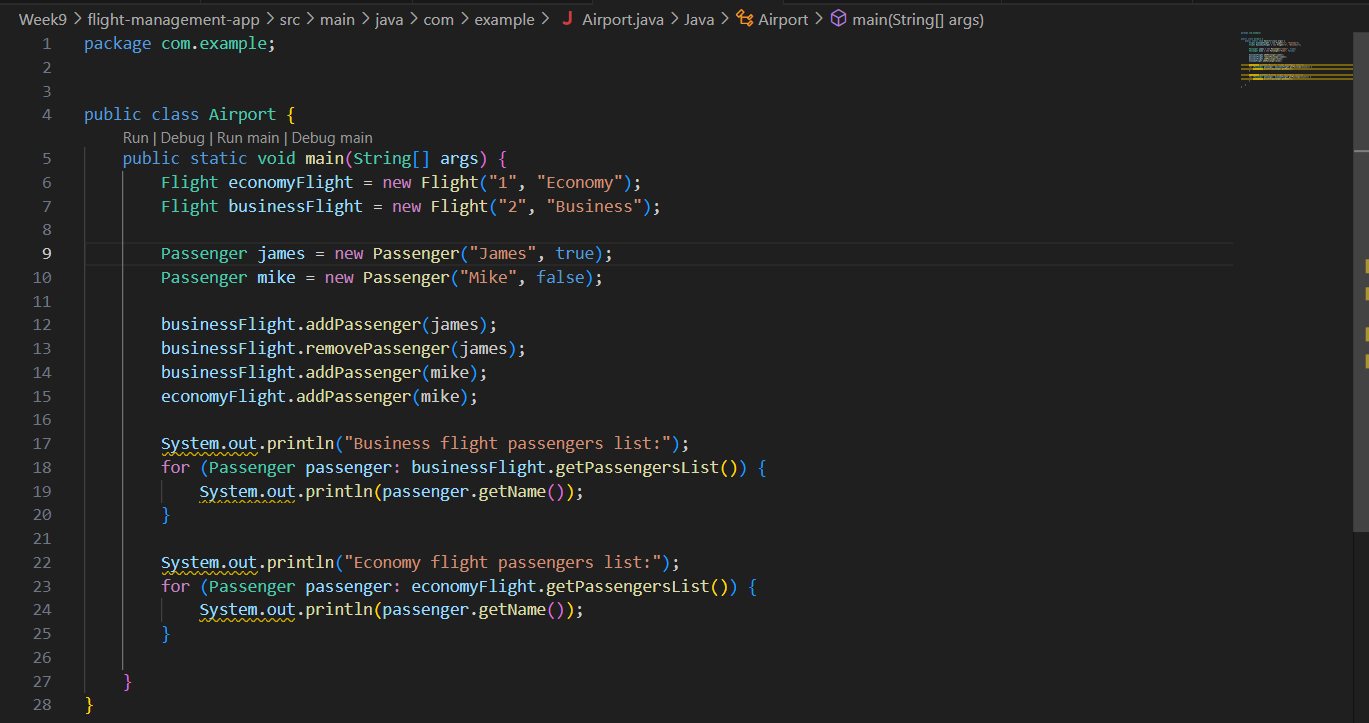


2. Flight



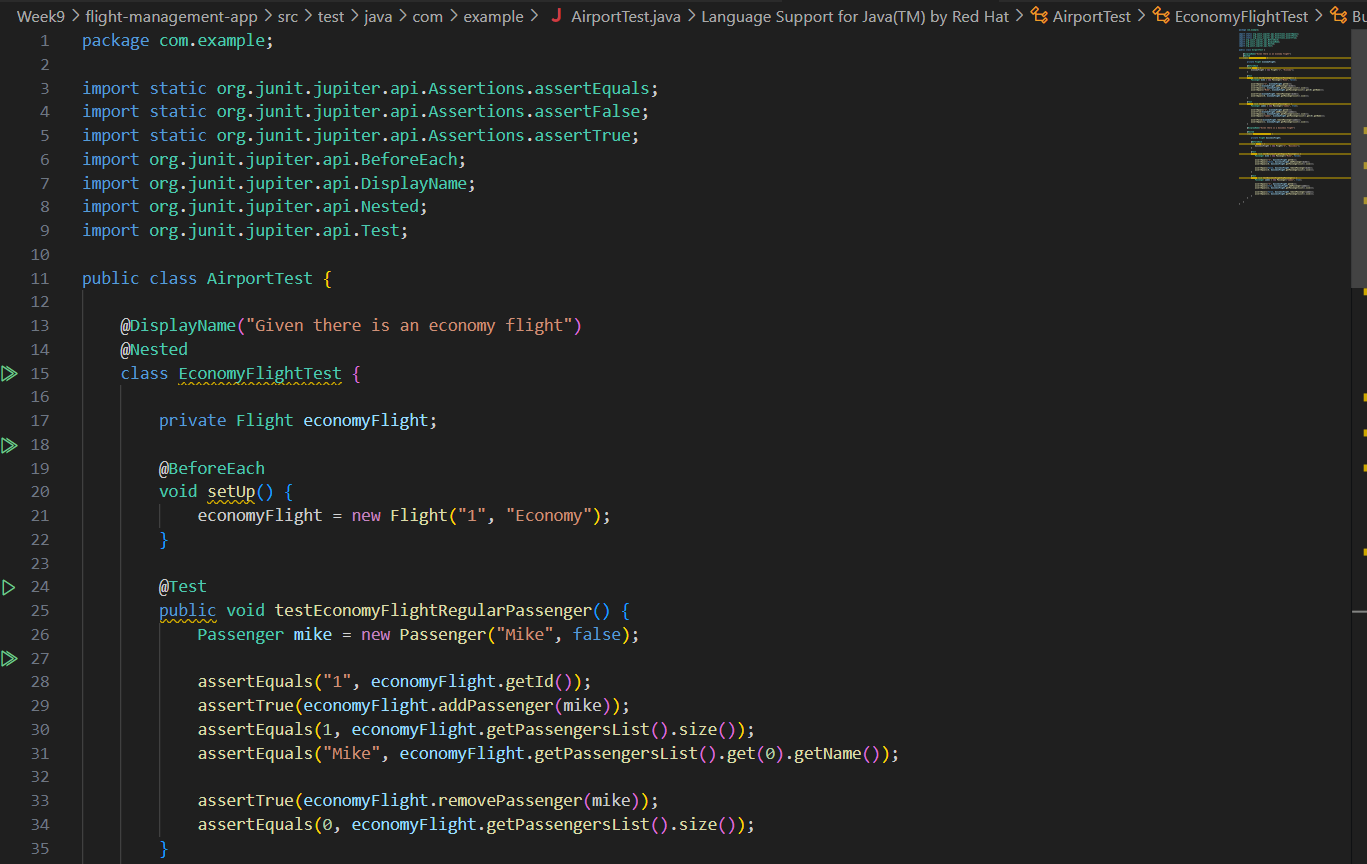


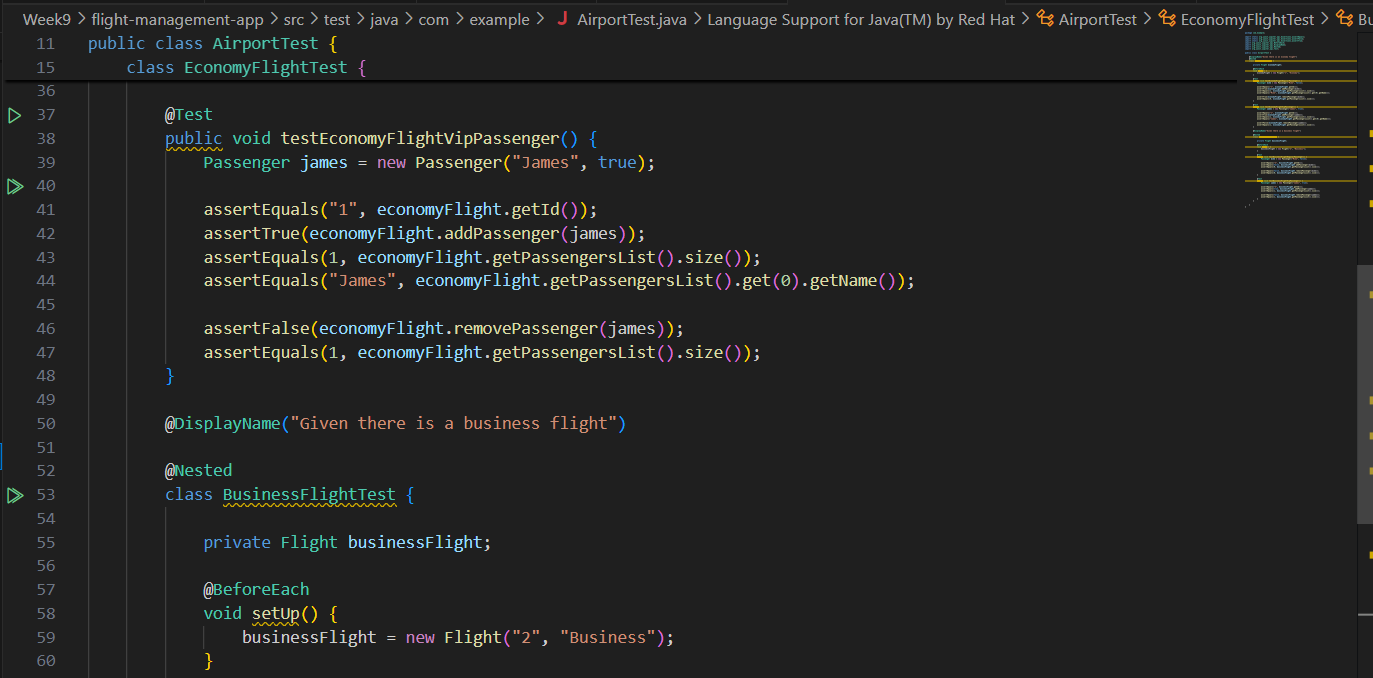
3. Airport/Main

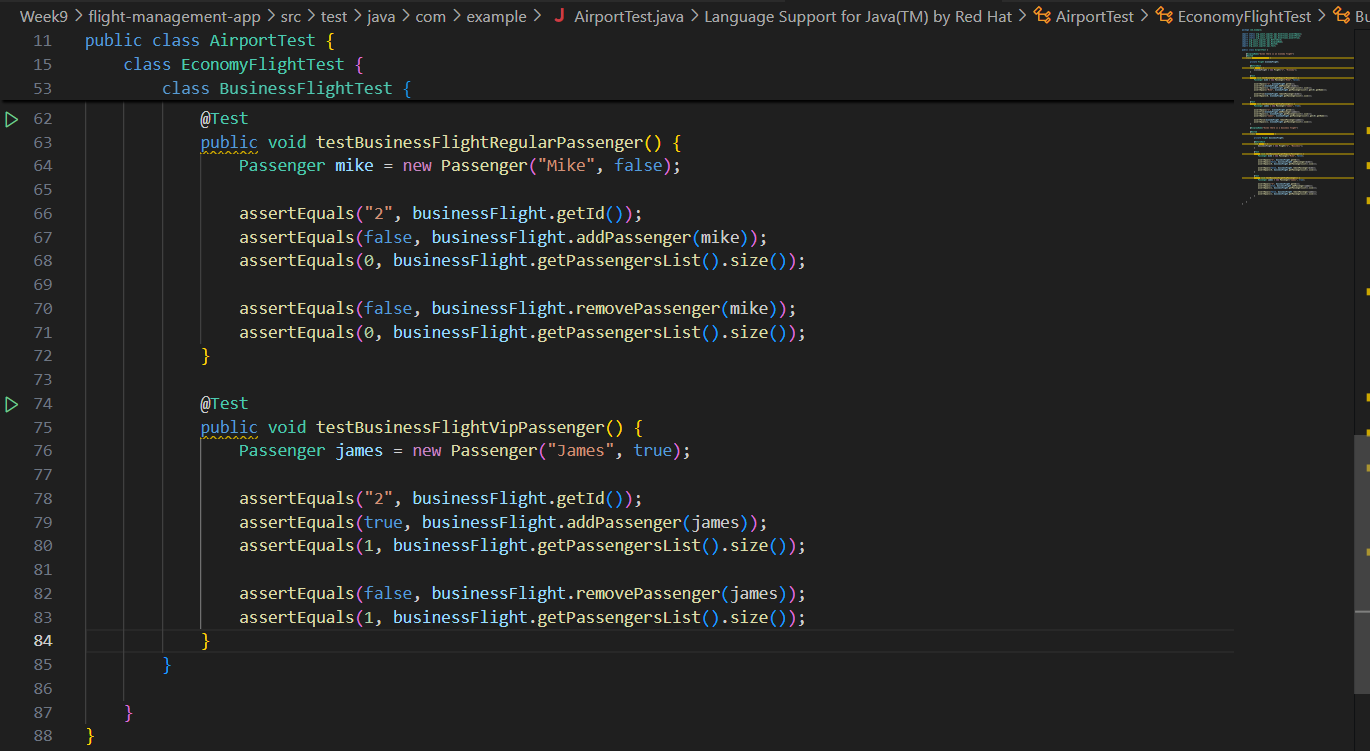


Class class tadi akan di tes di dalam

AirportTest.java

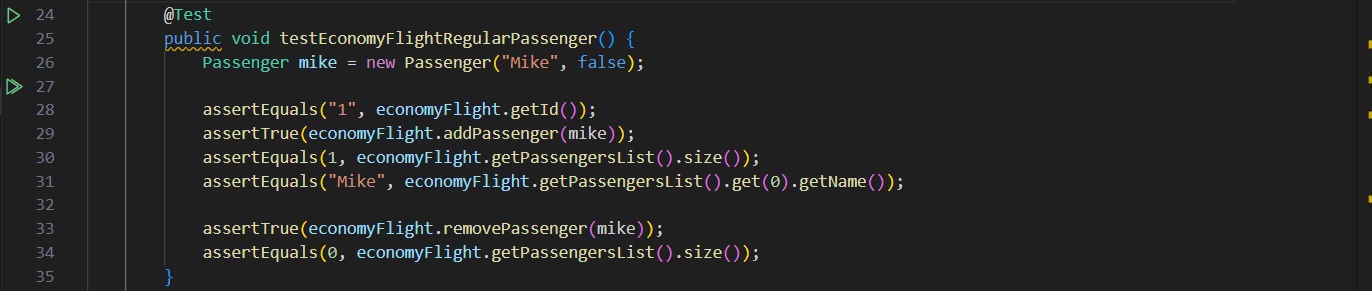






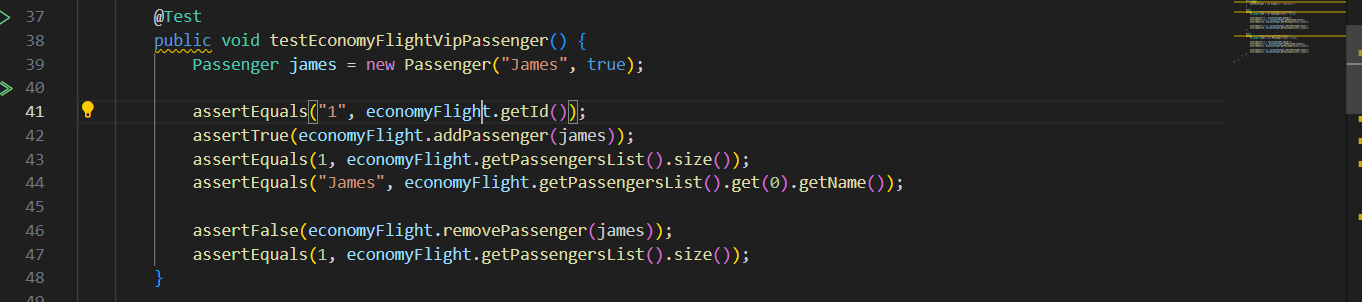
Dalam file AirportTest.java akan mengecek 4 Test yaitu:

1.



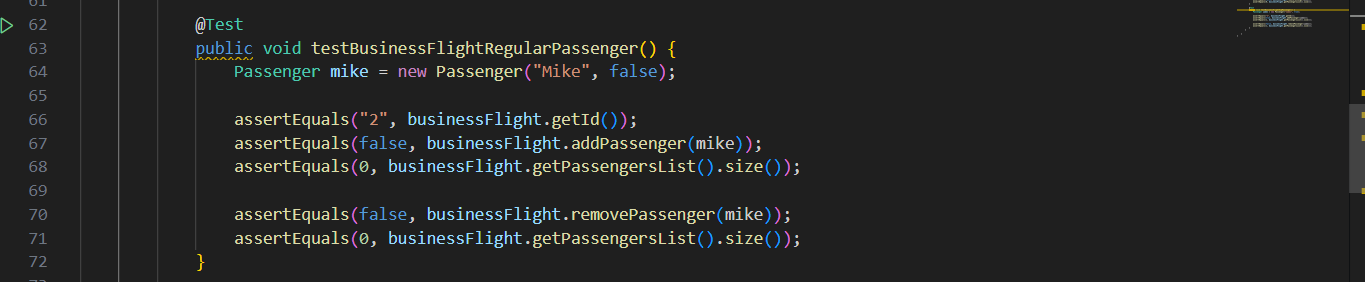
Dalam test ini akan mengecek passenger dalam penerbangan regular di kelas economy, dimana tes ini expected:

* Mengecek bahwa object mike dibuat dengan nama “Mike”, dan isVip() false
* Memastikan bahwa ID economyFlight adalah "1".
* Mengecek bahwa menambahkan Mike ke penerbangan ekonomi berhasil (bernilai true).
* Memastikan bahwa sekarang jumlah penumpang di flight menjadi **1** setelah Mike ditambahkan.
* Memastikan bahwa penumpang pertama (index 0) di daftar penumpang adalah **"Mike"**.
* Mengecek bahwa **menghapus Mike dari flight berhasil** (bernilai true).
* Setelah Mike dihapus, daftar penumpang harus kosong lagi (jumlah = 0).

2. 

Dalam test ini akan mengecek passenger dalam penerbangan vip di kelas economy, dimana tes ini expected:

* Mengecek bahwa object james dibuat dengan nama "James", dan isVip() true.
* Memastikan bahwa ID economyFlight adalah "1".
* Mengecek bahwa menambahkan james ke penerbangan ekonomi berhasil (bernilai true).
* Memastikan bahwa sekarang jumlah penumpang di flight menjadi 1 setelah james ditambahkan.
* Memastikan bahwa penumpang pertama (index 0) di daftar penumpang adalah "James".
* Mengecek bahwa menghapus james dari flight gagal (bernilai false).
* Setelah mencoba menghapus james, daftar penumpang tetap berisi satu orang (jumlah = 1).

3. 

Dalam test ini akan mengecek passenger dalam penerbangan bisnis namun bukan vip, dimana tes ini:

* Mengecek bahwa object mike dibuat dengan nama "Mike" dan isVip() bernilai false.
* Memastikan bahwa ID businessFlight adalah "2".
* Mengecek bahwa menambahkan mike ke businessFlight gagal (bernilai false).
* Memastikan bahwa setelah mencoba menambahkan mike, jumlah penumpang di flight tetap 0.
* Mengecek bahwa menghapus mike dari businessFlight juga gagal (bernilai false).
* Setelah mencoba menghapus mike, jumlah penumpang di flight tetap 0.

4. 

Dalam test ini akan mengecek vip passenger dalam kelas business, dimana tes ini:

* Mengecek bahwa object james dibuat dengan nama "James" dan isVip() bernilai true.
* Memastikan bahwa ID businessFlight adalah "2".
* Mengecek bahwa menambahkan james ke businessFlight berhasil (bernilai true).
* Memastikan bahwa setelah james ditambahkan, jumlah penumpang di flight menjadi 1.
* Mengecek bahwa menghapus james dari businessFlight gagal (bernilai false).
* Setelah mencoba menghapus james, daftar penumpang tetap berisi satu orang (jumlah = 1).