Operation contracts for methods in sequence diagram

Contract #1: viewFullInfo()
Operation: viewFlightInfo(source,destination)
Cross References: Use case 1: view flights given a source-destination pair
Preconditions: The flight must exist in the system
The user must be a registered user.
Postconditions: The flight object is found and returned to the caller.

Contract #2: viewBasicInfo()
Operation:viewBasicInfo(source,destination)
Cross References: Use case 1: view flights given a source-destination pair
Preconditions: The flight must exist in the system
The user must be a non-registered user.
Postconditions: The flight object is found and returned to the caller.

Contract #3: viewPrivateInfo()
Operation:viewPrivateInfo(source,destination)
Cross References: Use case 1: view flights given a source-destination pair
Preconditions: The flight must exist in the system
The user must be an airport administrator of either the source or destination airport.
Postconditions: The flight object is found and returned to the caller.

Contract #4: registerPrivateFlight()
Operation: registerPrivateFlight(airportCode)
Cross References: Use case 2: Register a flight into the flights database.
Preconditions: The airport must exist in the system
Postconditions: The flight object is created. The flight object is added to the database and. The flight object is associated with the airport that handles it. The flight object is assigned a source and destination airport.

Contract #5: registerNonPrivateFlight()
Operation: registerNonPrivateFlight(airportCode)
Cross References: Use case 2: Register a flight into the flights database.
Preconditions: The airport must exist in the system
Postconditions: The flight object is created. The flight object is added to the database and associated with one of the airline's aircrafts. The flight object is assigned a source and destination airport.