

Operation Contracts – Iteration 2

Traveller Class

Contract CO8: add_reservation

Operation: add_reservation(Reservation reservation)

Cross Reference: None.

Preconditions: There is an existing record for a Reservation object that is being added to a Traveller's object.

Postconditions: A Reservation object has been associated with Traveller (association formed).

Traveller Class

Contract CO9: list_trips

Operation: add_trips()

Cross Reference: None.

Preconditions: There is an existing Traveller record in the system.

Postconditions: None. The system is not modified after the listing of trips.

Travellers Class

Contract CO10: get_or_create

Operation: get_or_create(String firstName, String lastName, String age, Integer id)

Cross Reference: None.

Preconditions: There is a record for Traveller that is being created or looked for.

Postconditions: A new Traveller object has been created (instance creation).
A new Traveller object has been associated with Travellers (association formed).

Travellers Class

Contract CO11: find_by_id

Operation: find_by_id(String id)

Cross Reference: None.

Preconditions: None.

Postconditions: None. The system is not modified after fetching for a Traveller record.

Trips Class

Contract CO12: create_full_trip

Operation: create_full_trip(Connection connection, [Traveller] travellers, Reservations reservation_registry)

Cross Reference: There is an existing record for the provided Travellers, there is an existing connection, and there is an object Reservations that is being created.

Preconditions: None.

Postconditions: An object Trip is created (instance creation).

The Trip object is associated with a Reservation object (association formed).

The Reservation object is associated with the Reservations object (association formed).

Reservations Class

Contract CO13: add

Operation: add(Reservation reservation)

Cross Reference: None.

Preconditions: There is an existing Reservation record.

Postconditions: The Reservation record is associated with a Reservations object (association formed).

BookingSystem class

Contract CO14: save_trips

Operation: save_trips(String filepath)

Cross Reference: Client Views booked trips.

Preconditions: A BookingSystem object exists with Travellers and Trips collections.

Postconditions: A Trip object is created and returned. A Reservation object has been created for each traveller. Each traveller is appended to the list.

BookingSystem class

Contract CO15: load_trips

Operation: load_trips(String filepath)

Cross Reference: Client Views booked trips.

Preconditions: The RailNetwork connections have been loaded. A JSON file exists at the specified filepath.

Postconditions: Traveller objects have been created from persisted data (instance creation). Traveller objects have been associated with the Travellers registry (association formed). Trip objects have been recreated from persisted data (instance creation). Trip objects have been associated with the Trips registry (association formed). Reservation objects have been recreated and associated with Trips and Travellers (association formed). Ticket objects have been recreated and associated with Reservations (association formed).

BookingSystem class

Contract CO16: book_trip

Operation: book_trips(Connection connection, list traveller_data)

Cross Reference: Client Books Trip.

Preconditions: A BookingSystem object exists with Travellers and Connections collections.

Postconditions: All Traveller data has been serialized to JSON format. All trip data has been serialized to JSON format. All reservation and ticket associations have been serialized. A JSON file "trip_data.json" has been created or updated on disk.