Travel

User
«abstract»
- email: String
+ User(String email)

Client
- personalAddress: String
- creditCardNumber: String
- expiry: Date
- firstName: String
- lastName: String
+ Client(email: String, personalAddress: String, creditCardNumber: String, expiryString: String, firstName: String, lastName: String)
+ getters/setters

Administrator
+ Administrator()

Flight
- flightNumber: String
- airline: String
- cost: double
- departureDate: Date
- arrivalDate: Date
- origin: String
- destination: String
- numSeats: int
+ Flight(flightNumber: String, airline: String, cost: double, departureDate: Date, arrivalDate: Date, origin: String, destination: String)
+ getTravelTime(): String
+ getters/setters

Itinerary
- client: Client
- flights: Flight[]
- origin: String
- destination: String
- totalCost: double
- travelTime: String
+ Itinerary(Client client, Flight[] flights)
+ sumFlights(): double
+ sumTravel(): String
+ getters/setters

Controller

Storage
- path: String
+ Storage()
+ Storage(String path)
+ loadClients(path: String): void
+ loadFlights(path: String): void
+ getItineraries(email: String): Itinerary[]
+ getClients(): Client[]
+ getFlights(): Flight[]
+ setPassword(userEmail: String, password: String): void
+ getPassword(userEmail: String): String
+ getAllItineraries(): Itinerary[]
+ addItinerary(itinerary: Itinerary): void
+ addOrUpdateFlight(flight: Flight): void
+ getClient(email: String): Client
+ addOrUpdateClient(client: Client): void

Backend
- user: User
- flights: Flight[]
- data: Storage
+ Backend()
+ Backend(path: String)
+ getItineraries(departureDate: Date, origin: String, destination: String): Itinerary[]
+ sortItinerariesByTravelTime(itineraries: Itinerary[]): void
+ sortItinerariesByCost(itineraries: Itinerary[]): void
+ searchFlight(departureDate: String, origin: String, destination: String): String[]
+ searchClient(email: String): String
+ addFlight(path: String): void
+ addFlight(flightNumber: String, airline: String, cost: String, departDate: String, arrivalDate: String, origin: String, destination: String, numSeats: int): void
+ addClient(path: String): void
+ addClient(email: String, personalAddress: String, creditCardNumber: String, expiryString: String, firstName: String, lastName: String): void
+ addOrUpdateClient(client: Client): void
+ getClient(String email): Client
+ bookItinerary(itinerary: Itinerary): void
+ getAllClients(): Client[]
+ setUser(): void
+ login(user: String, Password: String): boolean
+ setUserPassword(user: String, pass: String): void

StorageException
+ StorageException()
+ StorageException(message: String)

CompareItinerairesByCost
+ compare(it: Itinerary, it2: Itinerary): int
+ compareTo(it: Itinerary, it2: Itinerary): int

CompareItinerairesByTravelTime
+ compare(it: Itinerary, it2: Itinerary): int
+ compareTo(it: Itinerary, it2: Itinerary): int