Taxi Company

Develop an application to manage a taxi company.
All the classes must belong to package **taxi**. Use the class **MainClass** as an example.

R1. Companies and Taxi

A taxi company is represented by the class **TaxiCompany**.
Taxis are represented by the class **Taxi**, that provides a constructor accepting the unique id of the taxi.
The method **toString**() from class **Taxi** returns the identifier of the taxi.
The method **addTaxi** (**String** id) from class **TaxiCompany** allows adding a new item to the taxi list of a company; if the identifier is already present, it raises an **InvalidTaxiName** exception.
When a new taxi is added to a company, it is inserted at the end of the available taxis queue.
The method **getAvailable**() returns the free taxis queue.

R2. Places and Passengers

The class **Place** represents a real address and provides a constructor accepting as arguments the actual address and the relative district (or quarter) name. (E.b.: `"Corso Duca Abruzzi 24", "crocetta"`).
The method **toString**() returns the address of the place.
The class **Passenger** represents a customer of the taxi company, it provides a constructor accepting as argument the place where the customer is located currently, i.e. where he/she should be picked up by a taxi
The method **getPlace**() returns the place where the passenger is currently located.

R3. Taxi Management

Taxi management will be handled with a First-In-First-Out strategy by means of a queue containing the available taxis.
The method **callTaxi**(**Passenger** p) assigns the first taxi in the available queue to the passenger passed as argument and returns the taxi itself; if no taxi is available then the call is lost.
The method **getLostTrips**() returns the total number of lost calls.
The class **Taxi** provides the methods for initiating and terminating a taxi trip:

- a taxi, once assigned to a passenger, can initiate its trip by means of method **beginTrip**(**Place** dest), that accepts the destination as argument, while the departure place is represented by the current passenger position.
- the trip terminates with the method **terminateTrip()** that assigns to the passenger the destination place and puts the taxi at the end of the available taxis queue.

R4. Trips

The class **Trip** represents a trip completed by a taxi. The method **toString**() returns a string containing the departure and arrival addresses, separated by a comma (`','`).
The method **getTrips**(**String** id) of the class TaxiCompany returns a list containing all the trips completed by the taxi identified by the argument in chronological order; if the identifier is not present in the list of taxis working for the company then it raises an **InvalidTaxiName** exception.

R5. Statistics

Class **TaxiCompany** provides two methods used to retrieve statistics:

- the method **statsTaxi()** returns the list of taxis (through the interface **InfoI**) sorted by decreasing number of completed trips (those for which the terminateTrip() method has been invoked); in case of tie use the alphabetic order of the id.
- the method **statsDistricts()** returns the list of districts that have been destination of at least a trip (through the interface InfoI), sorted by decreasing number of trips having that district as the destination; in case of ties use the alphabetic order of the district.
