

OOD - Individual assignment: Transportation hub

You are tasked to create a Windows form application for a transportation hub. Both passenger and freight transport pass through the hub.

With this application it should be possible to manage the vehicles and associated rides. In addition, saving and loading the vehicle info in the application as a file is also required.

Disclaimer

This is an individual assignment to give you better insight about your proficiency related to the covered concepts in OOD and the learning outcomes. Your teacher will give you a formative indication and feedback to help you understand your proficiency.



Note that this assignment is individual and is used by your teacher to help you get to the expected proficiency for OOD. For this reason, it must be your own code and you are not allowed to copy any part of it from your fellow peers. Your teacher will give you feedback about what you submit and we assume that it is your own code and that you understand it fully.

The end-user of the application must at least be able to do the following:

- **FR-V1: Add a new vehicle**
This must adhere to the data constraints about [vehicles](#). You should be able to add [Cars](#), [Vans](#) and [Trucks](#).
- **FR-V2: Modify an existing vehicle**
This must adhere to the data constraints about a [vehicle](#).
- **FR-S1: Sort**
Vehicles should be sortable on license plate.
Rides should be sortable on starting date first, and then total kilometres (when completed).
- **FR-R1: Reserve a ride at the hub**
This must adhere to the data constraints about a [ride](#). A ride contains a vehicle and passengers OR a vehicle and freight (volume and weight). It also should contain the kilometres that need to be travelled. An available vehicle is used for the ride. The ride is then added to the list of rides in the transportation hub dashboard. The ride status is then in progress. The vehicle is no longer available for rides until the ride is finished. If no vehicle is available a notification should be sent to the user. So, User input is KM to travel, AND {number of persons} OR {volume AND weight of the cargo}. The result is either a created ride or a message indicating no vehicle could be found.
- **FR-R2: Finish a ride**
This finishes a ride and returns the vehicle to the hub. The price for the ride is calculated based on the kilometres travelled. The ride status is changed to completed.
- **FR-R3: See an overview of the rides**
It must be possible to view all rides.
- **FR-P1: Persistent vehicle data**
It must be possible to save and load vehicle data to and from a file.
We do not accept solutions that communicate with a database.
- **FR-P2: Persistent ride data**
It must be possible to save and load ride data to and from a file.
We do not accept solutions that communicate with a database.

You will have to work with data related to the transportation hub entities for some functional requirements. The descriptions of these entities can be found below.

Transportation hub

The transportation hub is the central communication point for the user. It lists all vehicles, current rides and past rides. If needed, you may want to extend this with functions or properties.

As a minimum, the following data should be stored:

	Description	Constraints
<i>Rides</i>	All rides currently in progress.	It should be possible to finish a ride. It should be possible to list the vehicles that are on a specific ride. When creating a new ride, a number of passengers, or a weight and volume for freight should be added.
<i>Completed rides</i>	All rides currently completed.	It should be possible to retrieve the final price for a completed ride.
<i>Vehicles</i>	The vehicles currently available for rides.	
<i>Vehicles on a ride</i>	The vehicles currently not available for a ride.	

Ride

The ride is the core business of the transportation hub. As a minimum the following data should be stored:

	Description	Constraints
<i>Vehicle</i>	The associated vehicle	Required data.
<i>Amount of persons</i>	Not when hauling freight	Optional data.
<i>Volume of the cargo in m³</i>	Not when transporting passengers	Optional data.
<i>Weight of the cargo in kg</i>	Not when transporting passengers	Optional data.
<i>Price of the ride</i>	Final price of the ride, calculated by (price of the vehicle * number of kilometres) + starting price for a ride.	Required data. Price in Decimal, with 2 decimal places.
<i>Starting price</i>	The starting price for a ride. This contains the starting price for the ride.	Optional data. Price in Decimal, with 2 decimal places.
<i>Kilometres</i>	The amount of kilometres expected to be travelled for this ride.	Required data.
<i>Starting time</i>	Starting time of the ride	Required data. Date and time.
<i>End time</i>	End time of the ride	Required data. Date and time. End time must be bigger than start time.

Vehicle

A vehicle contains the passengers or freight.

	Description	Constraints
<i>Make and model</i>	The vehicles car make and model, e.g. "Mercedes Transporter 2015"	Optional data.
<i>License plate</i>	The license plate of the vehicle, e.g. "V-DL-003"	Required data.
<i>Gas usage litres per km</i>	Has the gas milage of the vehicle per Km.	Required data.
<i>Price per kilometre</i>	Has the prices in euro's per km. Based on 100/Gas usage litres per kilometre.	Required data.
<i>Calculate consumed fuel</i>	Calculates the fuel consumed by the vehicle using Total KM * Gas usage per km.	Required data. Total consumed in Decimal, with 2 decimal places.
<i>Total km</i>	Total kilometres driven with the vehicle until this point. It is updated after every completed ride.	Required data.

Freighter

A freighter is a vehicle type that can transport cargo.

	Description	Constraints
<i>Maximum weight</i>	The freighter can carry this weight in kilometres as a max.	Required data.
<i>Maximum volume</i>	The freighter can carry this volume in M ³ as a max.	Required data.

Passenger transport

A passenger transport is a vehicle type that can transport passengers.

	Description	Constraints
<i>Maximum passengers</i>	The vehicle can carry this amount of passengers as a maximum.	Required data.

Car

A car is a passenger transport.

Van

A van is both a passenger transport and a freighter, and so can be used for both purposes, but not at the same time.

Truck

A truck is a freighter.

The assignment

First analyse the case described above and design the classes for your application; make sure all functional requirements are supported by your design.

Note that it is expected that you use the programming concepts covered during semester 1 and OOD, especially abstraction and interfaces.

After you have determined what the classes and its members are, you can start implementing the code in a Windows Form application. You may decide yourself what controls to add and how to show the rides and vehicles, but just be sure to keep the user in mind (e.g. proper information shown, monkey proofing, user friendly layout, etc.).

Be sure to submit the solution of this assignment before the deadline on Canvas; both the implemented functionalities and code quality will be reviewed.