Java Enterprise Edition

Fall Semester 2020

Instructor: Zhuanyshev Ilyas

HOME TASK 2

Problem 1

Your aim is to develop a simple web application for registering transports according to their types.

As a database, you can use simple **ArrayList** of **Transport** interface, inside of **DBManager.java** class, where you must create all methods to manage with them.

Transports.java

+ String getTransportName();
+ String getTransportDescription();
+ int getTransportPrice();

Trucks.java implements **Transports** interface

- Long id;
- String name;
- double liftingCapacity;
- int price;
- int trailerPrice;

Bus.java implements **Transports** interface

- Long id;
- String name;
- String model;
- int passengerPlaces;
- int price;

Cars.java implements **Transports** interface

- Long id;
- String name;
- String model;
- String carcaseType; // sedan, hatchback, cabriolet etc..
- int maxSpeed;
- double engineVolume;
- int price;
- int year;

Implement all methods of **Transport.java** interface in your classes.

Create constructors, getters and setters for parameters.

Inside of **DBManager.java** class, create static **ArrayList** of **Transport** interface.

Also, create static methods to manage with transports:

void addTransport(Transports transport); **//this method adds new transport into list**

Transports getTransport (Long id); **//this method returns an object of transport by id**

ArrayList<Transports> getAllTransports(); **//this method returns a list of all transports**

void deleteTransport(Long id); **//this method deletes transport from list by id**

In main page, you will have a list of all transports shown as in example:

All Transports Page or Home Page

KAMAZ-5490 NEO

85000 USD

This truck can lift 25000 KG, with trailer price 15000 USD

Read More

Toyota Camry 70

25000 USD

Sedan, from 2018 year, with max speed: 240km/h, 2.4 liters

Read More

Bus Mercedes - eCitaro 2020

85000 USD

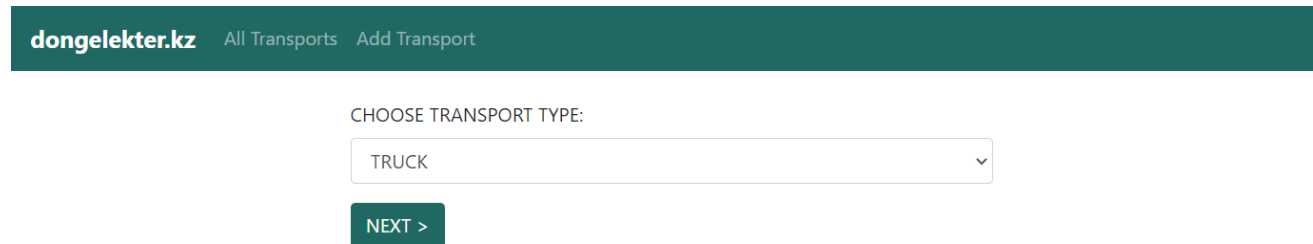This bus can transport 56 passengers

Read More

Use JSP pages as a views.

Use Bootstrap 4 Library for CSS and Design.

After choosing transport type, you must open a form with input fields, according to chosen transport type.

All forms pages of adding transport must be different and with different theme.

Add Transport Page

So, if you choose **Truck**, you will create an object of **Truck.java** class and complete its parameters from a form.

If you choose **Bus**, you will create an object of **Bus.java** class and complete its parameters from a form.

If you choose **Car**, you will create an object of **Car.java** class and complete its parameters from a form.

All these objects must be added into an **ArrayList<Transport>** in **DBManager.java** class.

Here, below, you will see an approximately designed form for each types of transports.

Add Transport Form Page if you choose Truck

dongelekter.kz   All Transports   Add Transport

TRUCK NAME:

Insert truck name

LIFTING CAPACITY (KG):

0

TRUCK PRICE:

0

TRAILER PRICE:

0

+ ADD NEW TRUCK

Add Transport Form Page if you choose Bus

dongelekter.kz   All Transports   Add Transport

BUS NAME:

Insert bus name

BUS MODEL:

Insert bus model

BUS PASSENGER PLACES:

0

BUS PRICE:

0

+ ADD NEW BUS

Add Transport Form Page if you choose Car

CAR NAME:

Insert car name

CAR MODEL:

Insert car model

CAR CARCASE TYPE:

SEDAN

CAR MAX SPEED (km/h):

0

CAR ENGINE VOLUME (liters):

0

CAR PRICE:

0

CAR YEAR:

2020

+ ADD NEW CAR