Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Новосибирский государственный технический университет»

Кафедра вычислительной техники

**Отчет**

По лабораторной работе №7

По дисциплине: «Технологии и методы программирования»

Организация в Java хранения данных в базе данных

Вариант 8

Выполнили: Павлюк А.С. Преподаватель: Копылова О.А.

Казанцев К.О.

Группа: АВТ-008

Факультет: АВТ

Новосибирск, 2022

**Цель работы:**

Изучить особенности реализации хранения данных в базе данных в приложениях Java.

**Задание к лабораторной работе:**

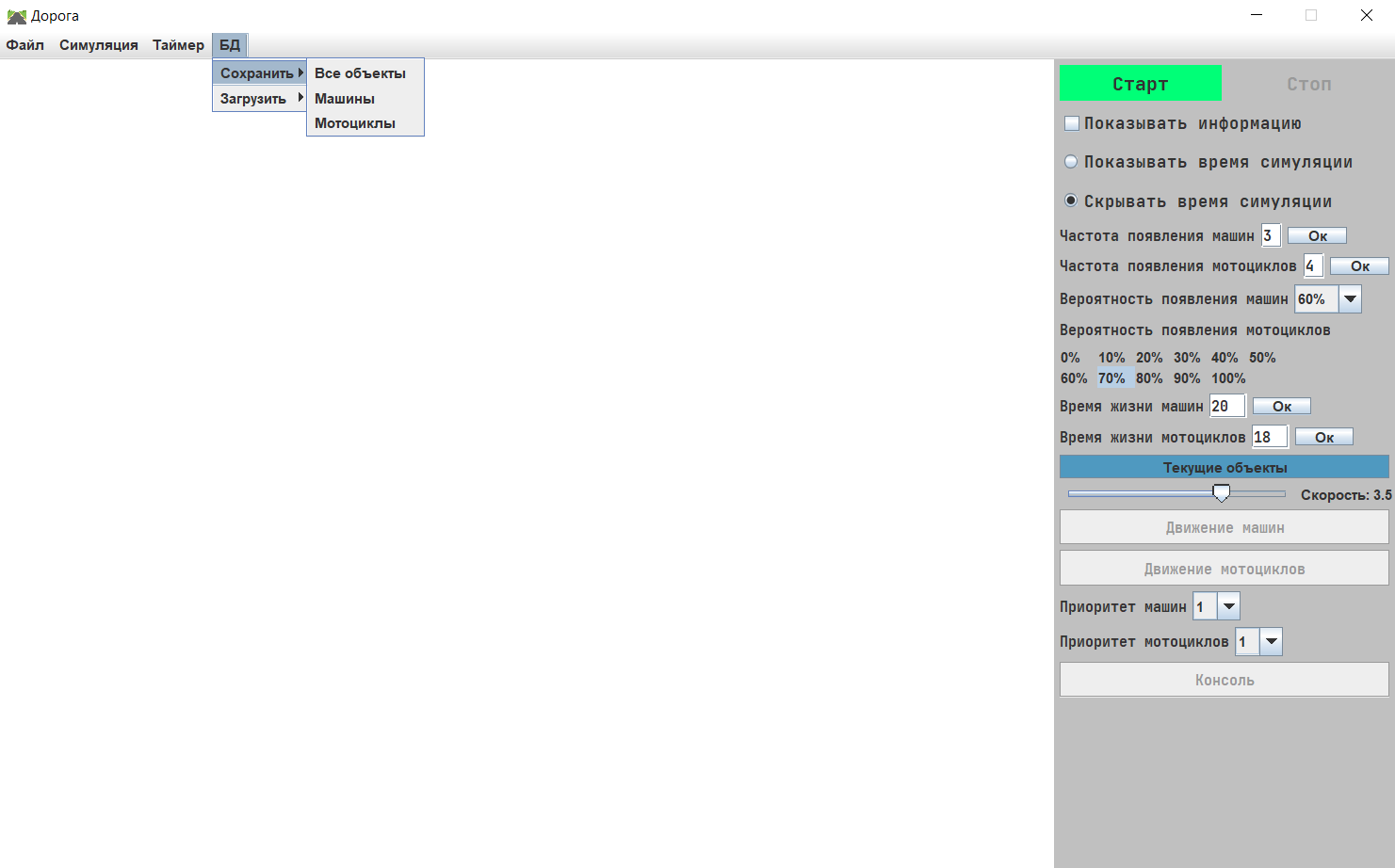
Доработать программу, созданную в лабораторной работу №6:

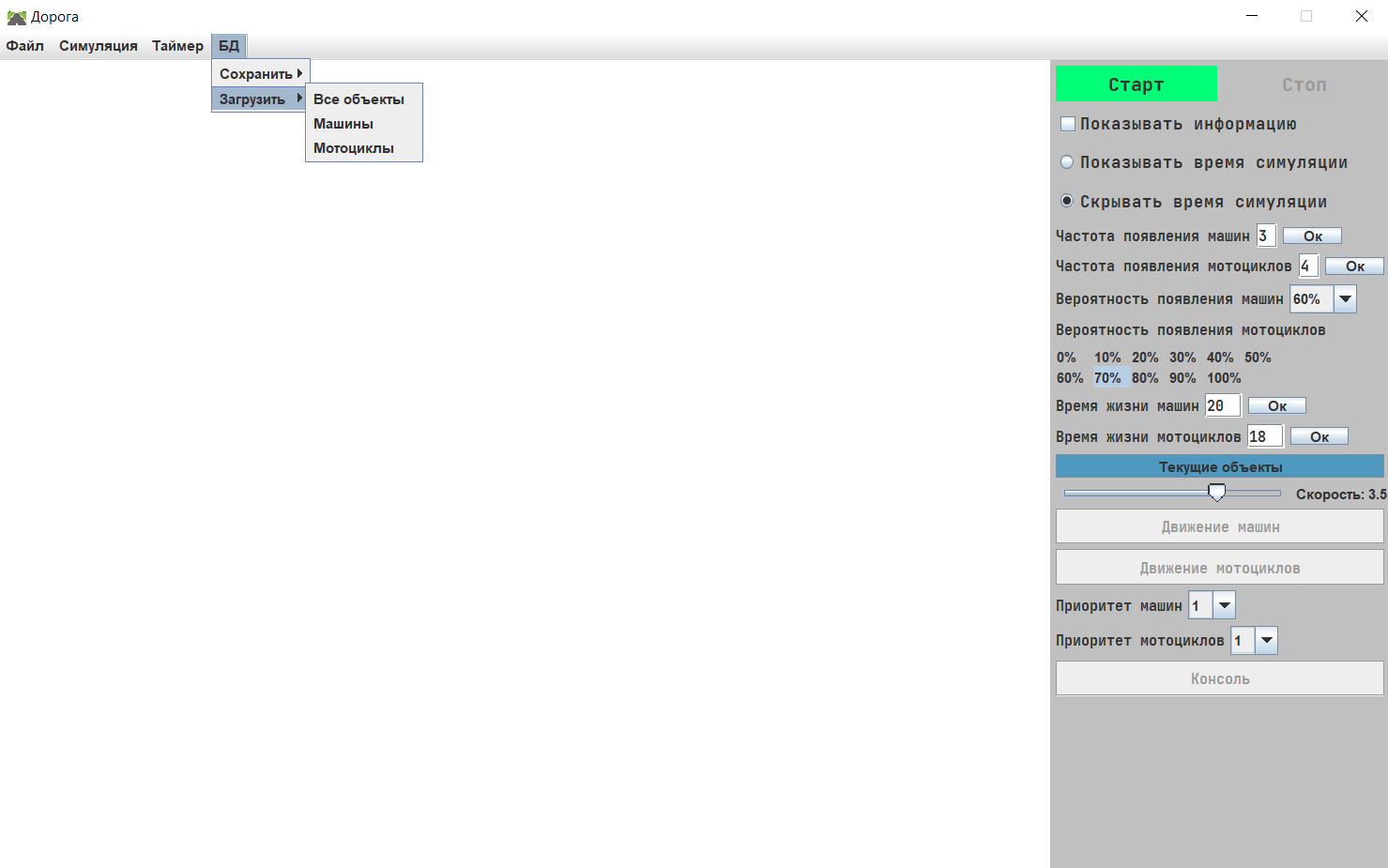
1. Добавить в меню пункты сохранение и загрузку объектов в базу данных. Можно использовать любую СУБД.
2. Загрузку и сохранение объектов только заданного типа.

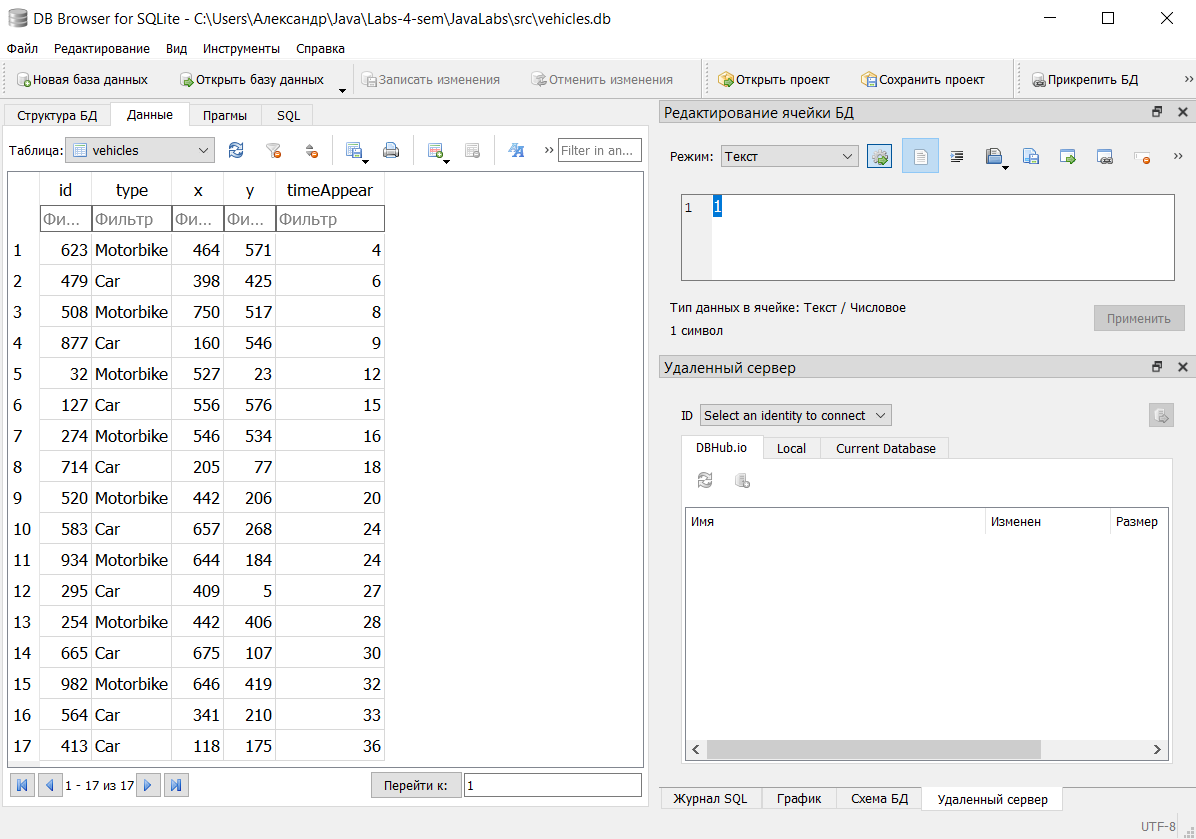
**Выполнение работы:**

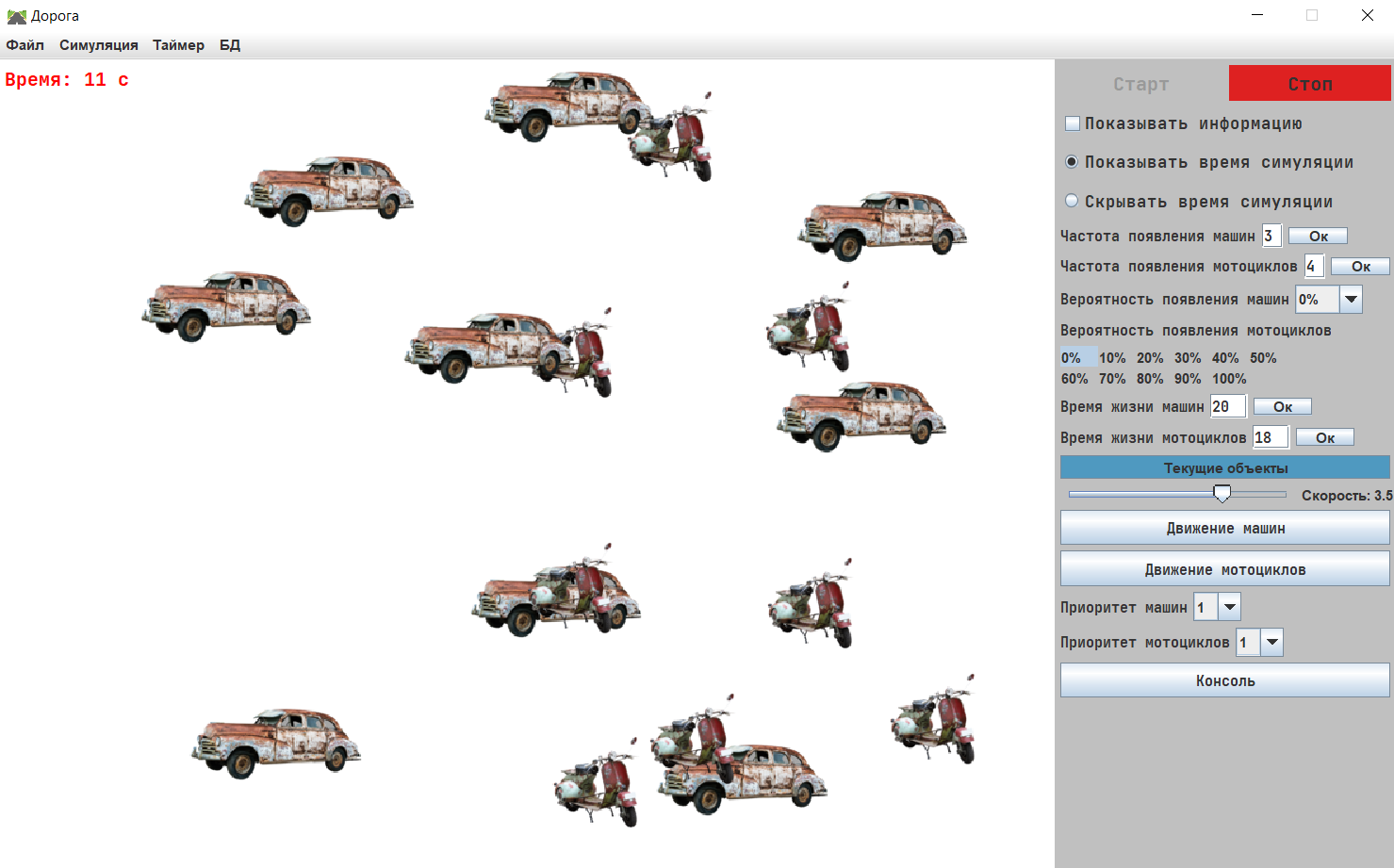
Для выполнения данной работы нами была выбрана СУБД Sqlite. Для работы с базами данных (далее «БД») в Java используется JDBC API, необходимо лишь скачать драйвер под выбранную СУБД и установить соединение. Далее была добавлена функция dropAndCreate, которая при каждом сохранении объектов в БД сначала удаляет таблицу, а затем создает ее снова. Строка меню пополнилась функциями работы с БД, они продемонстрированы в разделе «Выполнение работы». При загрузке объектов из БД их поле «время появления» заменяется на текущее время симуляции. Чтобы облегчить получение данных, в наследниках абстрактного класса Vehicle были добавлены конструкторы, которые принимают поля, загруженные из БД.

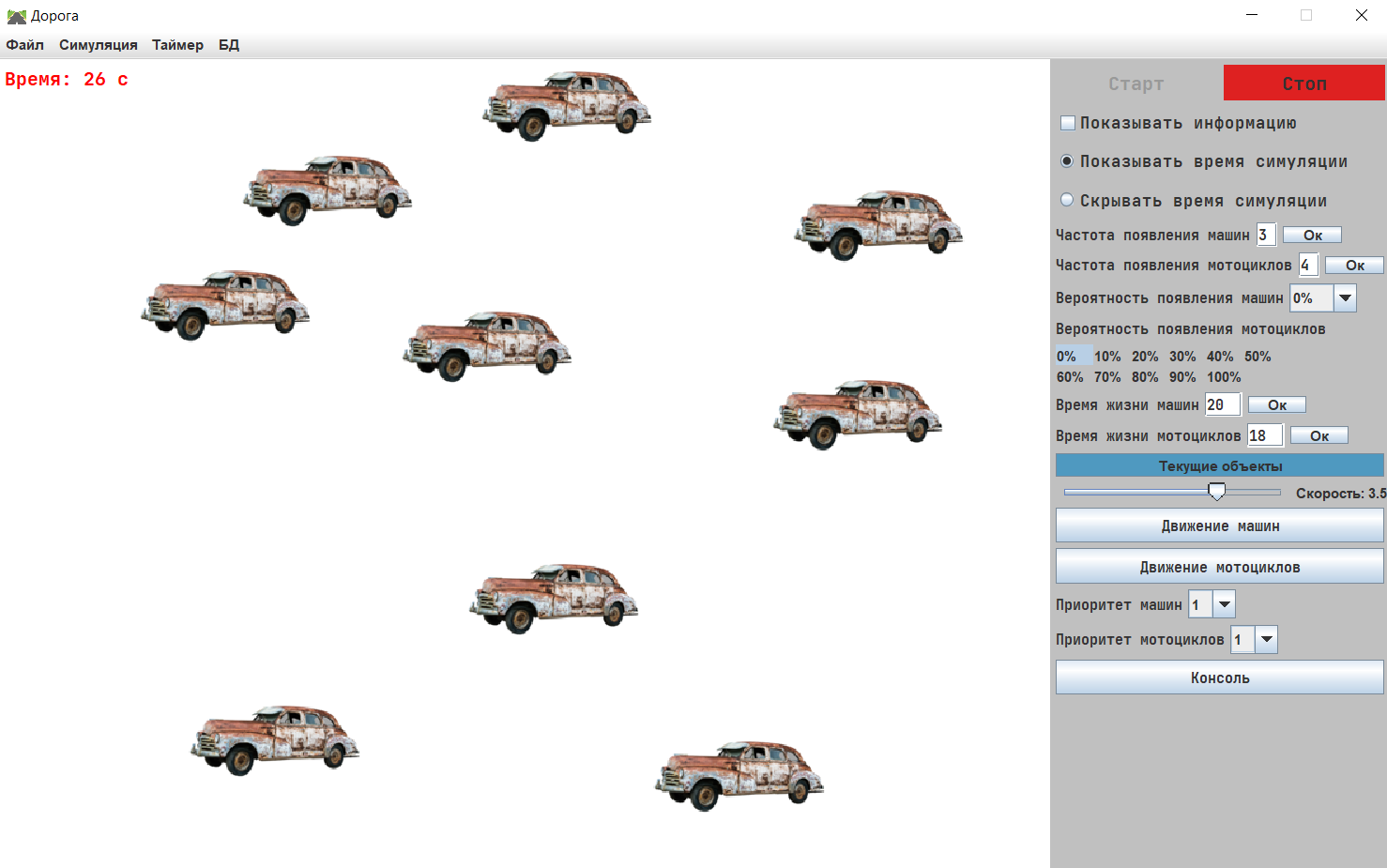
**Пример выполнения работы**

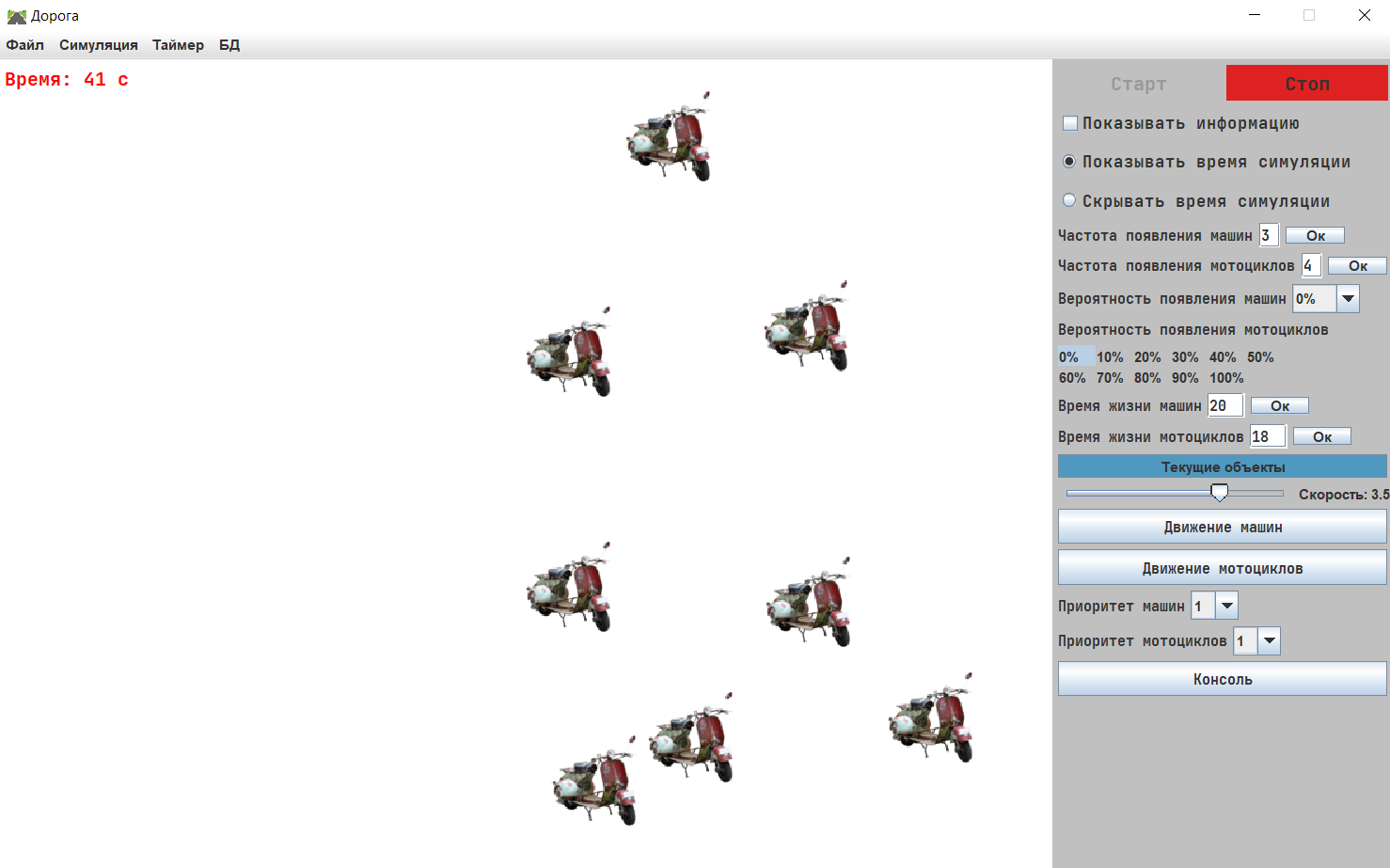
****

****









**Вывод:** в ходе выполнения лабораторной работы были изучены особенности реализации хранения данных в базе данных в приложениях Java. Получена практика написания простых SQL запросов.

**Изменения в классе Car:**

**public** Car(**int** x, **int** y, **int** id, **int** timeAppear) {  
 **this**.setX(x);  
 **this**.setY(y);  
 **this**.setId(id);  
 **this**.setTimeAppear(timeAppear);  
 **this**.setImage(**new** ImageIcon(**"C:\\Users\\Александр\\Java\\Labs-4-sem\\JavaLabs\\src\\nstu\\client\\imgs\\car.png"**));  
 Habitat.*ids*.add(id);  
}

**Изменения в классе Motorbike:**

**public** Motorbike(**int** x, **int** y, **int** id, **int** timeAppear) {  
 **this**.setX(x);  
 **this**.setY(y);  
 **this**.setId(id);  
 **this**.setTimeAppear(timeAppear);  
 **this**.setImage(**new** ImageIcon(**"C:\\Users\\Александр\\Java\\Labs-4-sem\\JavaLabs\\src\\nstu\\client\\imgs\\moto.png"**));  
 Habitat.*ids*.add(id);  
}

**Изменения в классе MyFrame:**

**public static final** String ***DB\_NAME*** = **"vehicles.db"**;  
**public static final** String ***CONNECTION\_STRING*** = **"jdbc:sqlite:C:\\Users\\Александр\\Java\\Labs-4-sem\\JavaLabs\\src\\"** + ***DB\_NAME***;  
**public static final** String ***VEHICLES\_TABLE*** = **"vehicles"**;  
**public static final** String ***CREATE\_VEHICLES\_TABLE*** = **"CREATE TABLE IF NOT EXISTS "** + ***VEHICLES\_TABLE*** +  
 **"(id INTEGER, type TEXT, x INTEGER, y INTEGER, timeAppear INTEGER)"**;  
**public static final** String ***DROP\_VEHICLES\_TABLE*** = **"DROP TABLE IF EXISTS "** + ***VEHICLES\_TABLE***;  
**public** Connection **con**;  
**public** Statement **statement**;  
**public** ResultSet **resultSet**;  
**public** JMenu **saveToDB**;  
**public** JMenuItem **saveToDBAllItem**;  
**public** JMenuItem **saveToDBOnlyCarsItem**;  
**public** JMenuItem **saveToDBOnlyMotosItem**;  
**public** JMenu **loadFromDB**;  
**public** JMenuItem **loadFromDBAllItem**;  
**public** JMenuItem **loadFromDBOnlyCarsItem**;  
**public** JMenuItem **loadFromDBOnlyMotosItem**;

**public void** dropAndCreate() {  
 **try** {  
 **con** = DriverManager.*getConnection*(***CONNECTION\_STRING***);  
 **statement** = **con**.createStatement();  
 **statement**.execute(***DROP\_VEHICLES\_TABLE***);  
 **statement**.execute(***CREATE\_VEHICLES\_TABLE***);  
 } **catch** (SQLException ex) {  
 ex.printStackTrace();  
 }  
}

JMenu dbMenu = **new** JMenu(**"БД"**);  
**saveToDB** = **new** JMenu(**"Сохранить"**);  
  
**saveToDBAllItem** = **new** JMenuItem(**"Все объекты"**);  
**saveToDBOnlyCarsItem** = **new** JMenuItem(**"Машины"**);  
**saveToDBOnlyMotosItem** = **new** JMenuItem(**"Мотоциклы"**);  
**saveToDBAllItem**.addActionListener(e -> {  
 dropAndCreate();  
 **try** {  
 **for** (Vehicle v : *vehicles*) {  
 **statement**.execute(**"INSERT INTO "** + ***VEHICLES\_TABLE*** + **" (id, type, x, y, timeAppear) "** +  
 **"VALUES("** + v.getId() + **", "** + **"\'"** + v.getClass().getName().substring(21) + **"\'"** + **", "** + (**int**) v.getX() + **", "** + (**int**) v.getY() + **", "** + (**int**) v.getTimeAppear() + **")"**);  
 }  
 } **catch** (SQLException ex) {  
 ex.printStackTrace();  
 }  
});  
**saveToDBOnlyCarsItem**.addActionListener(e -> {  
 dropAndCreate();  
 **try** {  
 **for** (Vehicle v : *vehicles*) {  
 **if** (v **instanceof** Car) {  
 **statement**.execute(**"INSERT INTO "** + ***VEHICLES\_TABLE*** + **" (id, type, x, y, timeAppear) "** +  
 **"VALUES("** + v.getId() + **", "** + **"\'"** + v.getClass().getName().substring(21) + **"\'"** + **", "** + (**int**) v.getX() + **", "** + (**int**) v.getY() + **", "** + (**int**) v.getTimeAppear() + **")"**);  
  
 }  
 }  
 } **catch** (SQLException ex) {  
 ex.printStackTrace();  
 }  
});  
**saveToDBOnlyMotosItem**.addActionListener(e -> {  
 dropAndCreate();  
 **try** {  
 **for** (Vehicle v : *vehicles*) {  
 **if** (v **instanceof** Motorbike) {  
 **statement**.execute(**"INSERT INTO "** + ***VEHICLES\_TABLE*** + **" (id, type, x, y, timeAppear) "** +  
 **"VALUES("** + v.getId() + **", "** + **"\'"** + v.getClass().getName().substring(21) + **"\'"** + **", "** + (**int**) v.getX() + **", "** + (**int**) v.getY() + **", "** + (**int**) v.getTimeAppear() + **")"**);  
 }  
 }  
 } **catch** (SQLException ex) {  
 ex.printStackTrace();  
 }  
});  
  
**loadFromDB** = **new** JMenu(**"Загрузить"**);  
**loadFromDBAllItem** = **new** JMenuItem(**"Все объекты"**);  
**loadFromDBOnlyCarsItem** = **new** JMenuItem(**"Машины"**);  
**loadFromDBOnlyMotosItem** = **new** JMenuItem(**"Мотоциклы"**);  
**loadFromDBAllItem**.addActionListener(e -> {  
 **h**.*vehicles*.clear();  
 **h**.**carCount** = 0;  
 **h**.**motoCount** = 0;  
 **try** {  
 **con** = DriverManager.*getConnection*(***CONNECTION\_STRING***);  
 **statement** = **con**.createStatement();  
 **resultSet** = **statement**.executeQuery(**"SELECT id, type, x, y FROM "** + ***VEHICLES\_TABLE***);  
 **while** (**resultSet**.next()) {  
 **int** id = **resultSet**.getInt(1);  
 String type = **resultSet**.getString(2);  
 **int** x = **resultSet**.getInt(3);  
 **int** y = **resultSet**.getInt(4);  
 **if** (type.equals(**"Car"**)) {  
 **h**.*vehicles*.add(**new** Car(x, y, id, (**int**) *time*));  
 **h**.**carCount**++;  
 } **else if** (type.equals(**"Motorbike"**)) {  
 **h**.*vehicles*.add(**new** Motorbike(x, y, id, (**int**) *time*));  
 **h**.**motoCount**++;  
 }  
 }  
 } **catch** (SQLException ex) {  
 ex.printStackTrace();  
 }  
});  
**loadFromDBOnlyCarsItem**.addActionListener(e -> {  
 **h**.*vehicles*.clear();  
 **h**.**carCount** = 0;  
 **h**.**motoCount** = 0;  
 **try** {  
 **con** = DriverManager.*getConnection*(***CONNECTION\_STRING***);  
 **statement** = **con**.createStatement();  
 **resultSet** = **statement**.executeQuery(**"SELECT id, x, y FROM "** + ***VEHICLES\_TABLE*** + **" WHERE type = 'Car'"**);  
 **while** (**resultSet**.next()) {  
 **int** id = **resultSet**.getInt(1);  
 **int** x = **resultSet**.getInt(2);  
 **int** y = **resultSet**.getInt(3);  
 **h**.*vehicles*.add(**new** Car(x, y, id, (**int**) *time*));  
 **h**.**carCount**++;  
 }  
 } **catch** (SQLException ex) {  
 ex.printStackTrace();  
 }  
});  
**loadFromDBOnlyMotosItem**.addActionListener(e -> {  
 **h**.*vehicles*.clear();  
 **h**.**carCount** = 0;  
 **h**.**motoCount** = 0;  
 **try** {  
 **con** = DriverManager.*getConnection*(***CONNECTION\_STRING***);  
 **statement** = **con**.createStatement();  
 **resultSet** = **statement**.executeQuery(**"SELECT id, x, y FROM "** + ***VEHICLES\_TABLE*** + **" WHERE type = 'Motorbike'"**);  
 **while** (**resultSet**.next()) {  
 **int** id = **resultSet**.getInt(1);  
 **int** x = **resultSet**.getInt(2);  
 **int** y = **resultSet**.getInt(3);  
 **h**.*vehicles*.add(**new** Motorbike(x, y, id, (**int**) *time*));  
 **h**.**motoCount**++;  
 }  
 } **catch** (SQLException ex) {  
 ex.printStackTrace();  
 }  
});  
dbMenu.add(**saveToDB**);  
dbMenu.add(**loadFromDB**);  
**saveToDB**.add(**saveToDBAllItem**);  
**saveToDB**.add(**saveToDBOnlyCarsItem**);  
**saveToDB**.add(**saveToDBOnlyMotosItem**);  
**loadFromDB**.add(**loadFromDBAllItem**);  
**loadFromDB**.add(**loadFromDBOnlyCarsItem**);  
**loadFromDB**.add(**loadFromDBOnlyMotosItem**);  
menu.add(fileMenu);  
menu.add(simulationMenu);  
menu.add(timerMenu);  
menu.add(dbMenu);