Question 1 [4 points]

We call a vehicle **VIP** if its identifier ends with a special sequence of characters (given as parameter **"viptag"** -- see below). For example, if this sequence of characters is **"_vip"**, then vehicles with identifiers like **"v1_vip"** and **"car_vip"** are **VIP** vehicles.

Add a ***Dequeuing Strategy*** to the simulator that advances at most **"limit"** (a number -- see below) vehicles giving priority to **VIP** vehicles in the order in which they appear in the queue, and then to other vehicles in the order that they appear in the queue. For example, if we have a queue with vehicles **[v1_vip,c1,c2,v2_vip,c3]** (in this order) and **"limit"** is 3, then vehicles **[v1_vip,v2_vip,c1]** are advanced (in this order). If **"limit"** is 7, then vehicles **[v1_vip,v2_vip,c1,c2,c3]** are advanced (in this order).

The **JSON** input corresponding to this strategy has the following syntax:

```
{
  "type" : "vip_dqs",
  "data" : { "viptag" : "_vip", "limit" : 5 }
}
```

where

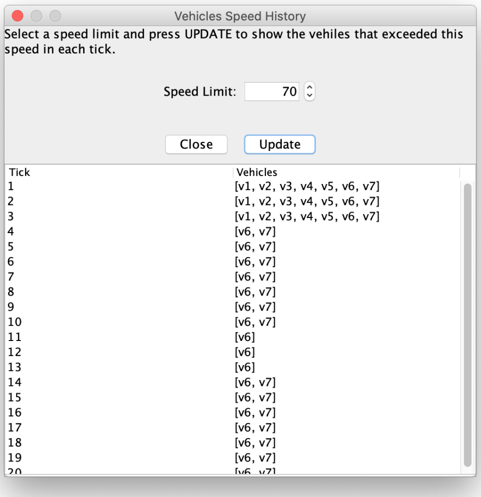
1. The value of **"viptag"** is the special sequence of characters that identify **VIP** vehicles. It is mandatory, and an exception should be thrown if it does not appear.
2. The value of **"limit"** represents the maximum number of vehicles that can be advanced in a single tick. It is optional, with default value 1.

Together with this **PDF**, you have a **JSON** file that you can use to test your implementation (it corresponds to the example above). You can use `s1.endsWith(s2)` to check if `String s1` ends with a `String s2`.

In your solution, you can add new classes but you are not allowed to modify existing ones except the `Main` class. A solution that does not respect this requirement will be given 0 points.

Question 2 [6 points]

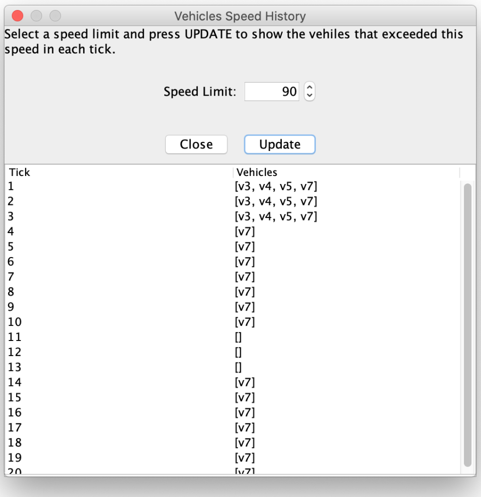
Add a new button to the **`ControlPanel`**, such that when clicked it opens a Dialog with the following aspect:



Select a speed limit and press UPDATE to show the vehiles that exceeded this speed in each tick.

Speed Limit: 70

Tick	Vehicles
1	[v1, v2, v3, v4, v5, v6, v7]
2	[v1, v2, v3, v4, v5, v6, v7]
3	[v1, v2, v3, v4, v5, v6, v7]
4	[v6, v7]
5	[v6, v7]
6	[v6, v7]
7	[v6, v7]
8	[v6, v7]
9	[v6, v7]
10	[v6, v7]
11	[v6]
12	[v6]
13	[v6]
14	[v6, v7]
15	[v6, v7]
16	[v6, v7]
17	[v6, v7]
18	[v6, v7]
19	[v6, v7]
20	[v6, v7]



Select a speed limit and press UPDATE to show the vehiles that exceeded this speed in each tick.

Speed Limit: 90

Tick	Vehicles
1	[v3, v4, v5, v7]
2	[v3, v4, v5, v7]
3	[v3, v4, v5, v7]
4	[v7]
5	[v7]
6	[v7]
7	[v7]
8	[v7]
9	[v7]
10	[v7]
11	[]
12	[]
13	[]
14	[v7]
15	[v7]
16	[v7]
17	[v7]
18	[v7]
19	[v7]
20	[v7]

where the user can select a **Speed Limit** (between 0 and `Integer.MAX_VALUE`), and when **`UPDATE`** is clicked it shows in each row of the table: (2) a tick, i.e., a time; and (2) the identifiers of vehicles that exceeded the speed limit at that time (at the end of the corresponding simulation step). For example, suppose we have executed the simulation `20` ticks (using the **`JSON`** file provided with this question):

1. If we choose `70` as a speed limit and click on **`UPDATE`**, the table will be as the one shown on the left, which indicates that: during the first `3` ticks all vehicles exceeded the speed limit; from tick `4` to `10` only vehicles `v6` and `v7` exceeded the speed limit; from tick `11` to `13` only vehicle `v6` exceeded the speed limit; and from tick `14` to `20` only vehicles `v6` and `v7` exceeded the speed limit.
2. If we choose `90` as a limit and click on **`UPDATE`**, the table will the one on the right.

When the dialog window opens, the table can be empty or already filled with information for the selected limit (do not worry about this, it depends much on how you decided to implement this question, both solutions are valid).

The new button in the **`ControlPanel`** should be disabled during the simulation and enabled again when the simulation is over or stopped. Together with this **`PDF`**, you have an icon that should be used for the new button, and a **`JSON`** file that you can use to try your implementation (it corresponds to the tables above).

- **When you implement this question, think well of the information you need to store in order to generate the table. Be careful, as if you do not think of this well you will end up showing the information of the last tick in all rows.**
- **You are not allowed to modify or add any class the model, you can only modify or add classes to the view. A solution that does not respect this requirement will be given 0 points.**