

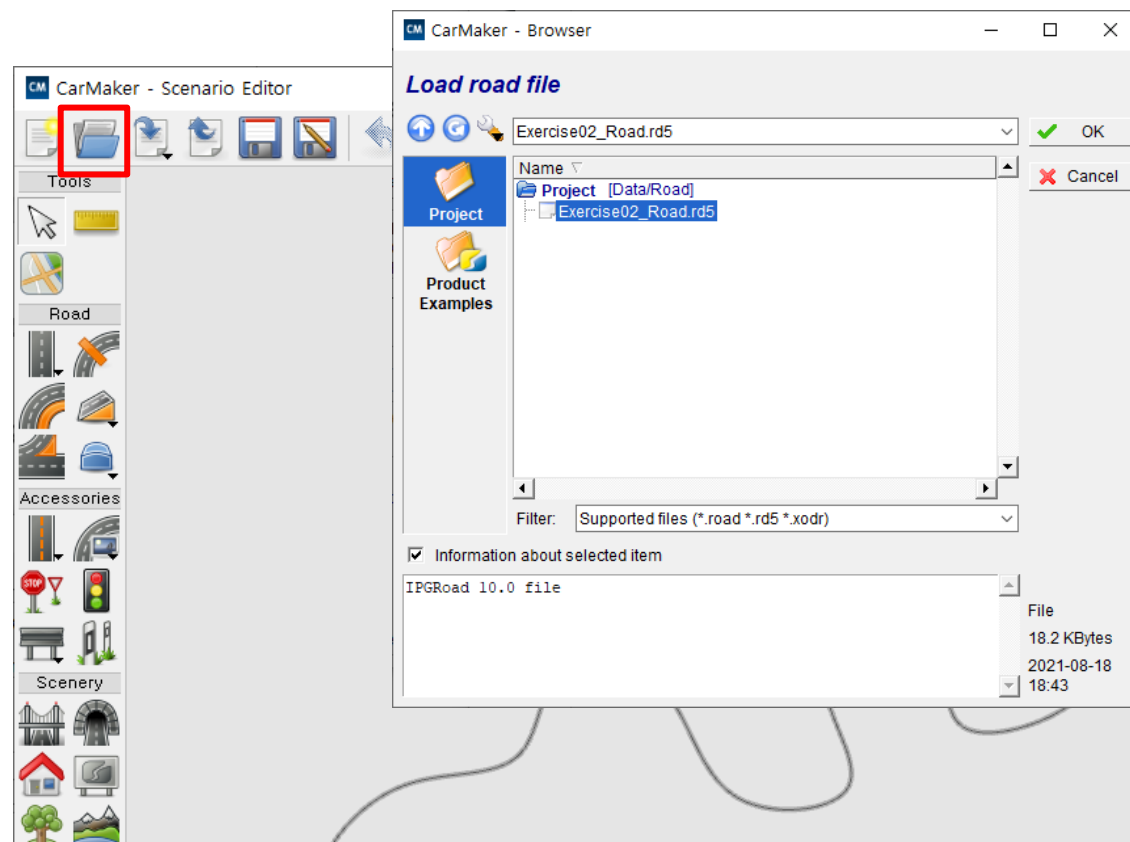
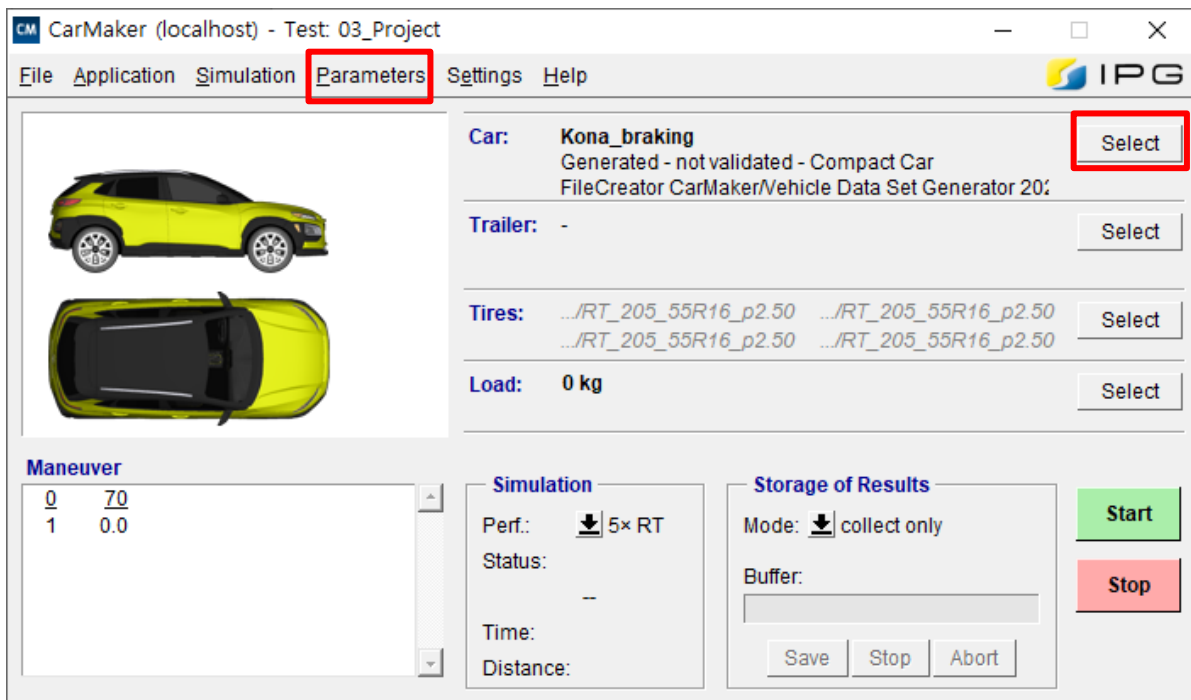
# 인천 대학교 자율주행 소프트웨어 챌린지

3차 소과제 – Minimanuever command 모범 답안

# 소과제 #3

## TestRun 구성

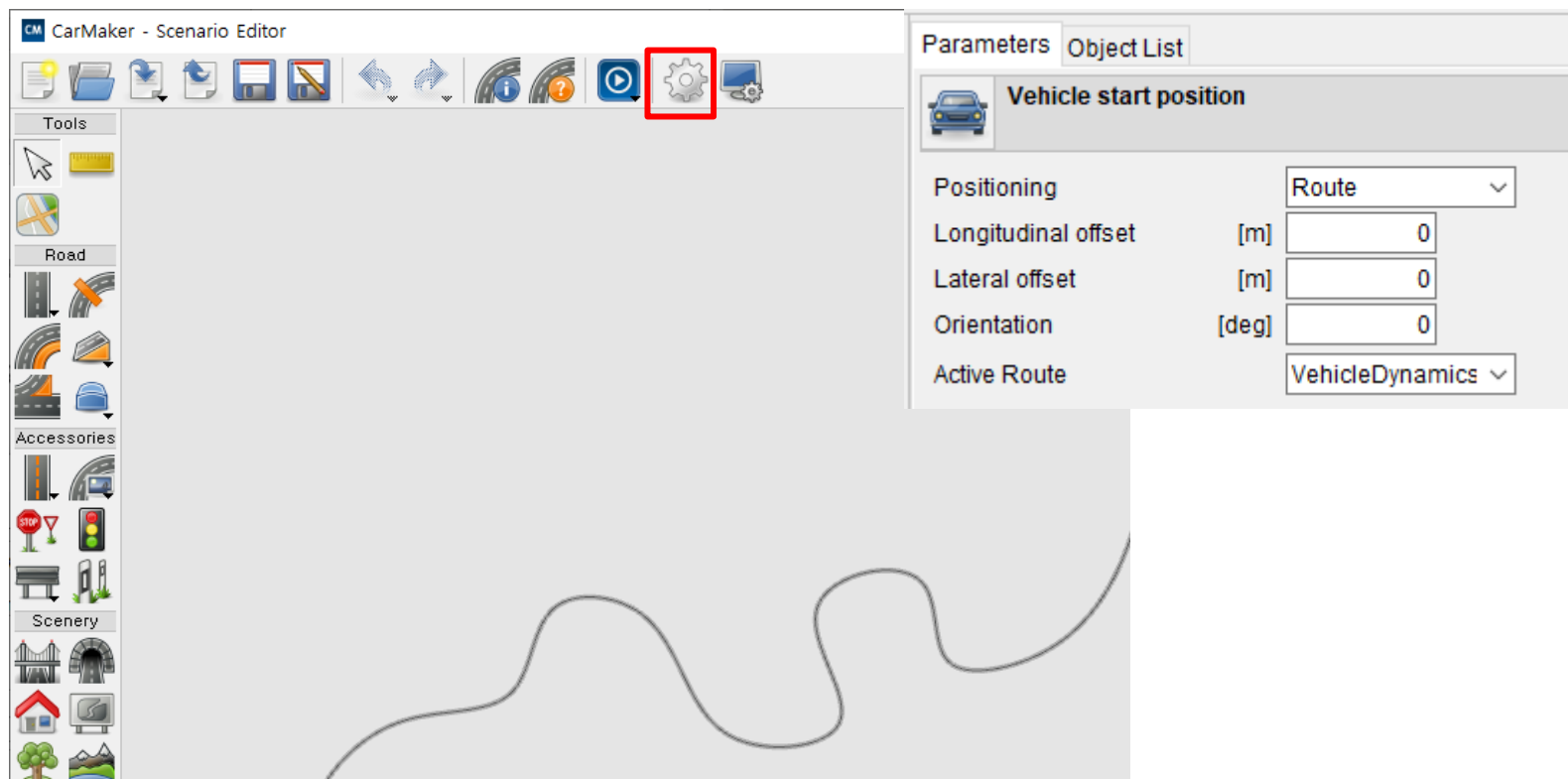
- CarMaker Main GUI > File > New 버튼을 이용해 새로운 TestRun 생성
- CarMaker Main GUI > Car select 버튼을 통해 1차 소과제 결과물인 Kona\_braking Load
- CarMaker Main GUI > Parameters > Scenario/Road > Load Road file을 통해 2차 소과제 Road file Open



# 소과제 #3

## TestRun 구성

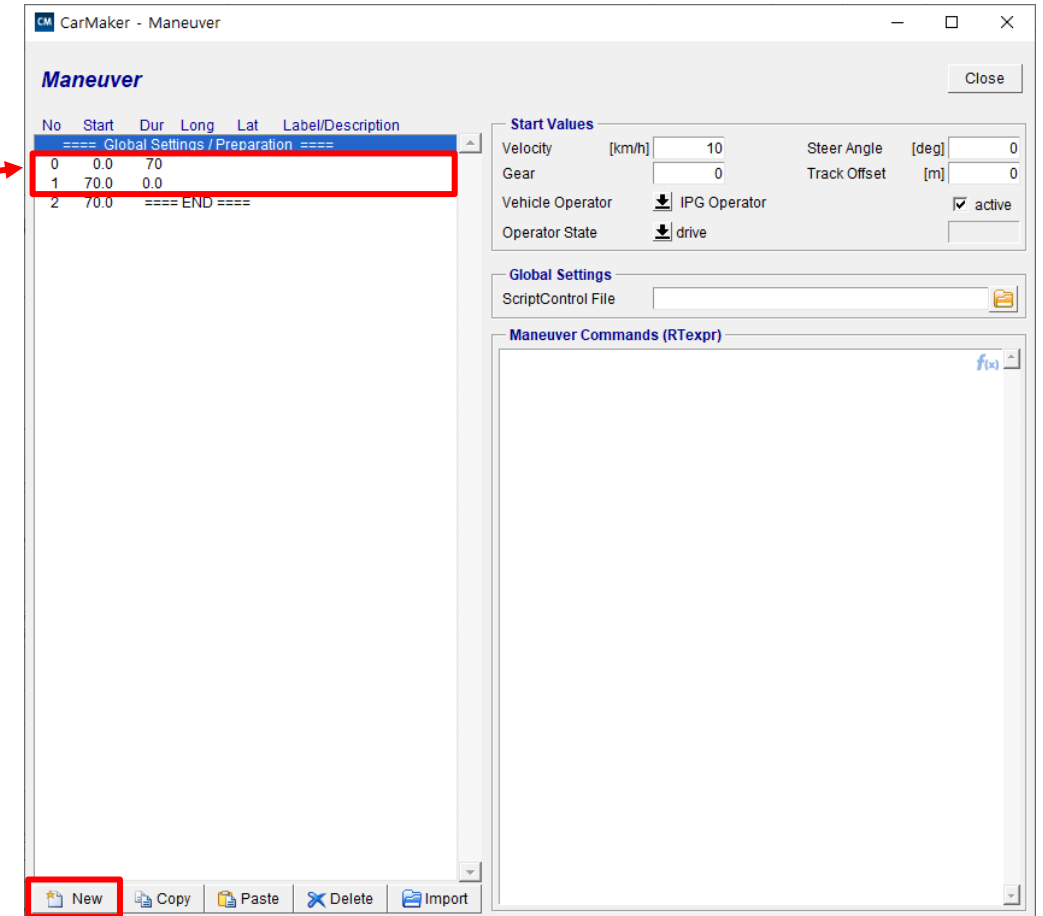
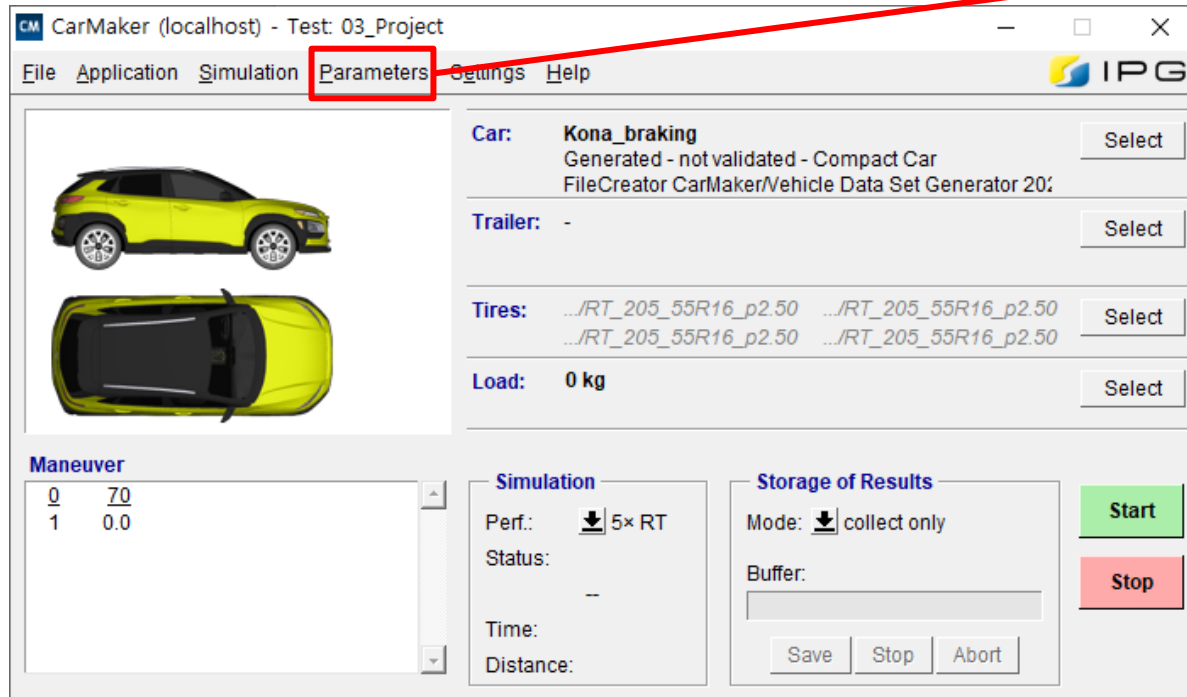
- Scenario Editor > Settings에서 Vehicle Start Position 설정
  - Positioning: Route
  - Active Route : VehicleDynamics



# 소과제 #3

## TestRun 구성

- CarMaker Main GUI > Parameters > Maneuver Open
  - New 버튼을 이용해 2개의 Minimanuever 생성

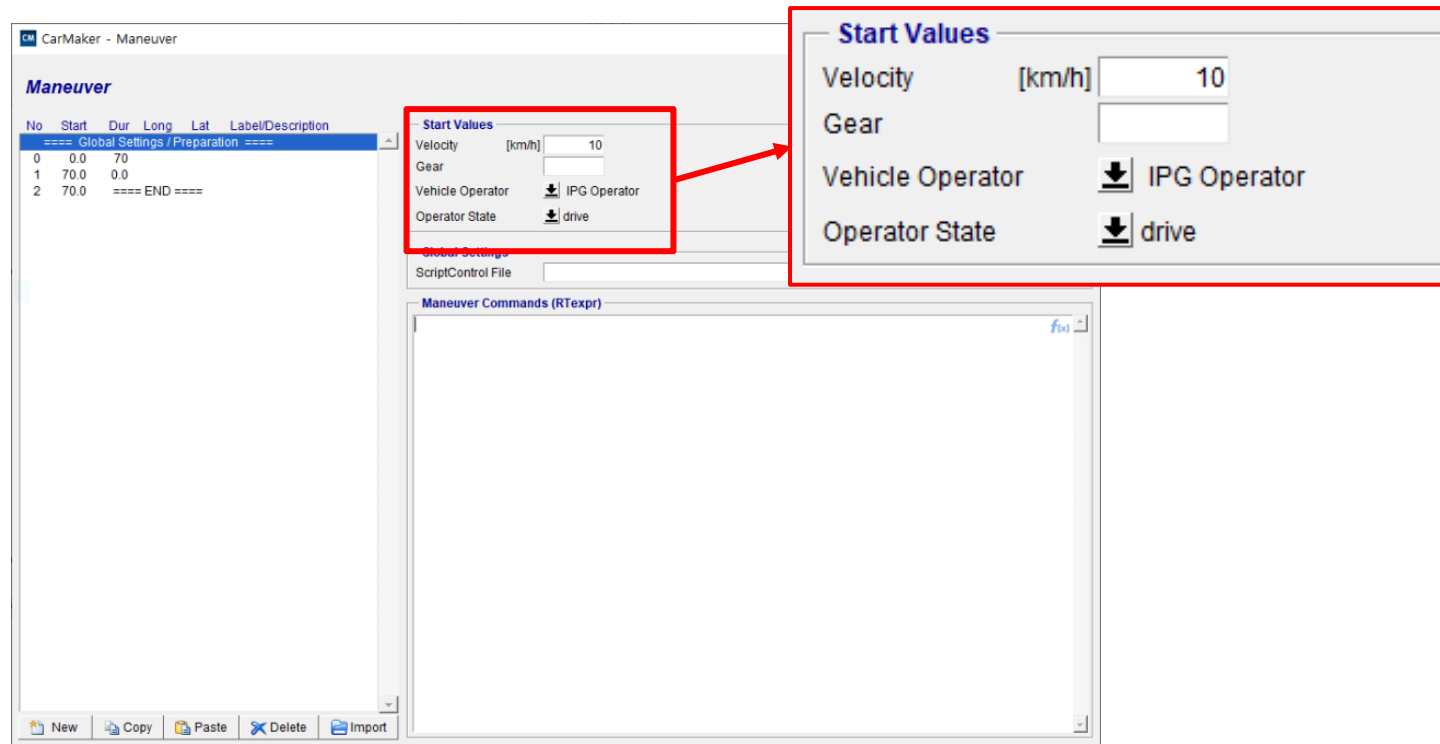


# 소과제 #3

## Minimanuever 구성 – Global settings/Preparation

### ■ Global Settings/Preparation

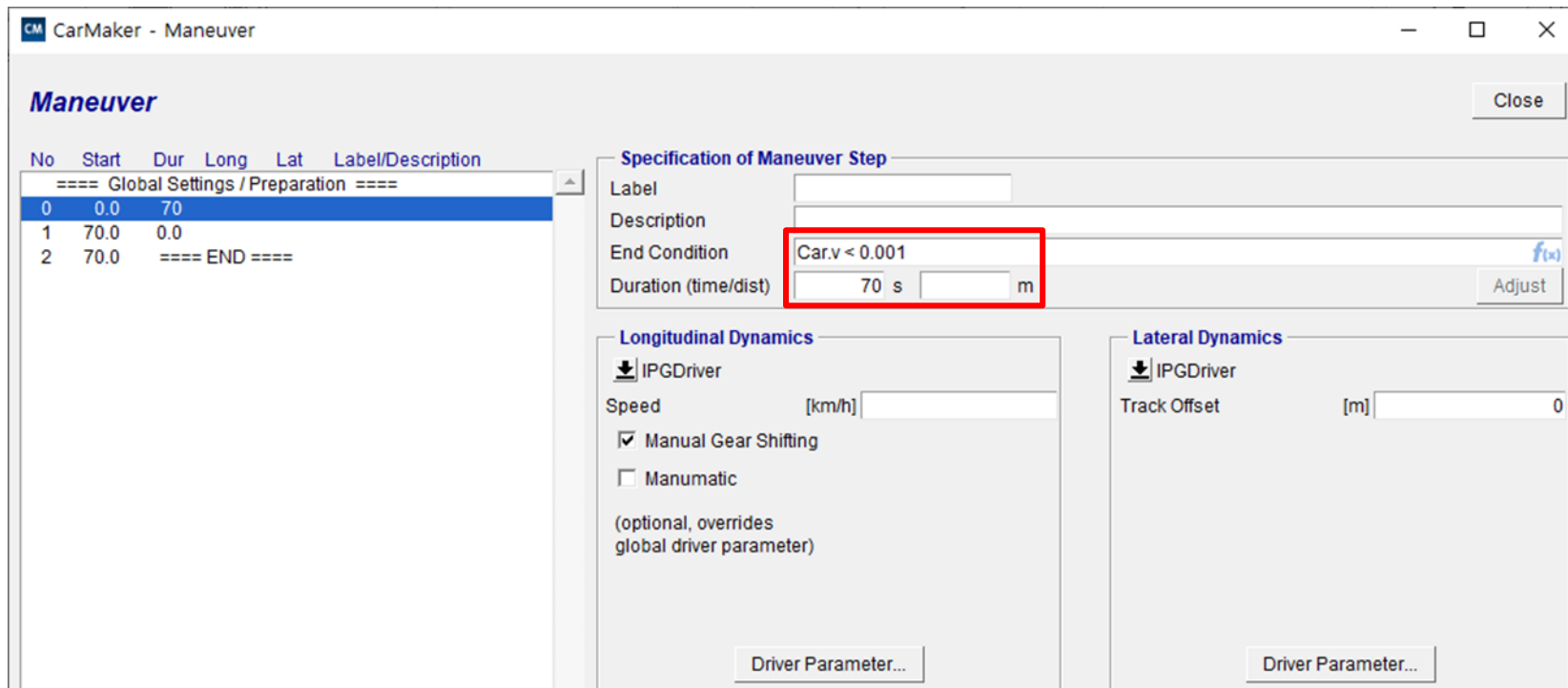
- Start Values > Velocity 값을 0에서 10 으로 변경. (0 초과, 30km/h 이하의 적절한 값)
  - Minimanuever 01에서 End Condition으로 인해 시뮬레이션이 종료되지 않기 위함.



# 소과제 #3

## Minimaneuver 구성 – Minimaneuver No.0

- Minimaneuver No.0
  - 시뮬레이션 시작 후 70초 이상 주행 시 또는 차량의 속도가 0.003 km/h 이하일때 시뮬레이션 종료 조건 추가.
    - End Condition과 Duration은 둘 중 가장 먼저 만족하는 조건을 따라간다.
    - Car.v는 기준 단위가 m/s이기 때문에  $0.001\text{m/s} \approx 0.003\text{km/h}$  이므로  $\text{Car.v} < 0.001$ 로 표현한다.



# 소과제 #3

## Minimaneuver 구성 – Minimaneuver No.0

### ■ Minimaneuver No.0

- 주행 시작 30m 후 Sin 파형을 이용하여 Lane Change를 하기위한 Minimaneuver Commands 구성
  - ① sin 파형 계산을 위한 CarMaker Quantities 생성
  - ② sin함수로 구성된 LaneOffset의 sin함수 특성을 활용하여 0보다 크고 0.001보다 작은 값을 갖는 주기마다 pulse 함수를 만족하여 true조건이 되면 flag 값이 1씩 증가하도록 구성 (초기 flag = 0)
  - ③ 차량의 주행 거리가 30m 이상이고, LaneOffset으로 계산된 sin 파형이 두번째 일 때 **DM.LaneOffset**에 Sin 값을 입력.  
조건을 만족하지 않는다면 DM.LaneOffset은 0의 값을 가지게 됨.

\* (A ? B : C 는 If A then B else C를 의미)

차량의 횡방향 offset. +값이 왼쪽 방향, -값이 오른쪽 방향을 가짐.



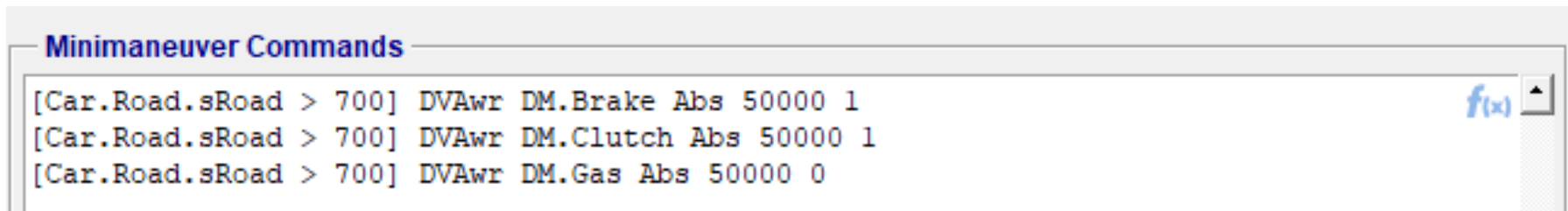
```
Minimaneuver Commands
1 Eval first() ? Qu::LaneOffset = 0
2 Eval first() ? Qu::flag = 0
3 Eval pulse(LaneOffset < 0.001 && LaneOffset > 0) ? flag = flag+1
4 Eval LaneOffset = 1.5*sin(DM.ManTime/1.5-pi/2+3)+1.5
5 Eval (Car.Road.sRoad > 30) && (flag < 2) ? DM.LaneOffset = LaneOffset : DM.LaneOffset = 0
```

# 소과제 #3

## Minimaneuver 구성 – Minimaneuver No.0

- Minimaneuver No.0
  - 차량의 주행 거리가 700m 이상이 되었을 때 5초 동안 Full Braking 진행
  - DVA Command를 활용하여 구성할 수 있음.
    - Car.Road.sRoad : 차량의 주행 거리
    - DM.Brake, DM.Clutch, DM.Gas : 각각 차량의 Brake pedal, Clutch pedal, Accelerator pedal position을 나타냄. (Full = 1)
    - [조건] DVAwr UAQ Mode time(ms) Value 의 문법 사항을 이용할 수 있음.

\* ( [A] B 는 If A then B를 의미)



```
Minimaneuver Commands
[Car.Road.sRoad > 700] DVAwr DM.Brake Abs 50000 1
[Car.Road.sRoad > 700] DVAwr DM.Clutch Abs 50000 1
[Car.Road.sRoad > 700] DVAwr DM.Gas Abs 50000 0
```



# 소과제 #3

## Minimaneuver 구성 – Minimaneuver No.0

### ■ Minimaneuver No.0

- 차량의 방향지시등 및 비상등 점등

- ① Car.Road.sRoad가 700 m 이상일 경우 Brake Pedal position이 0에서 1로 변경 시 change 함수가 이를 확인, true를 내보내게됨. hist 함수를 이용하여 현재 Brake Pedal positio과 바로 직전의 timestep(1 ms)보다 클 때를 확인. 조건을 모두 만족시키면 차량의 비상등 점등.
  - DM.Light.Hazard으로 차량의 비상등 점등 가능. (On = 1, Off = 0)
- ② Route에 구성된 우회전 경로를 따라 주행 시 방향지시등을 작동 해야함. 우회전을 위해 Steer wheel angle 값이 생기고, 차량의 주행 거리가 우회전 범위 내에 있을 때 Indicator에 -1를 입력, 그렇지 않다면 0을 입력하여 방향지시등 off
- ③ sin 파형을 이용하여 주행 시 차선 변경이 이루어짐. sin 파형이 입력되는 구간과 hist 함수를 활용하여 좌/우 방향지시등 점등

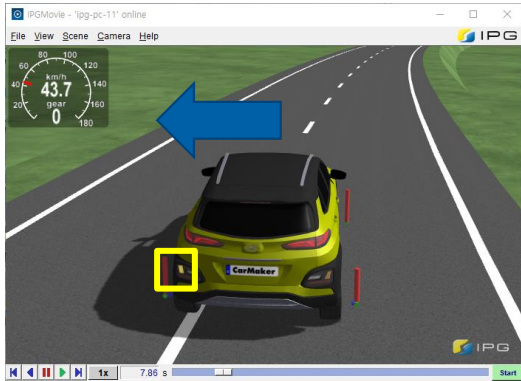
#### Minimaneuver Commands

```
1 Eval ((Car.Road.sRoad > 699) && change(DM.Brake) && DM.Brake > hist(DM.Brake)) ? DM.Lights.Hazard = 1
2 Eval ((DM.Steer.Ang < -0.2) && (Car.Road.sRoad < 450) && (Car.Road.sRoad > 250)) ? DM.Lights.Indicator = -1 : DM.Lights.Indicator = 0
3 Eval (Car.Road.sRoad > 30) && (Car.Road.sRoad < 200) && (DM.LaneOffset > hist(DM.LaneOffset)) ? DM.Lights.Indicator = 1
Eval (Car.Road.sRoad > 30) && (Car.Road.sRoad < 200) && (DM.LaneOffset < hist(DM.LaneOffset)) ? DM.Lights.Indicator = -1
```

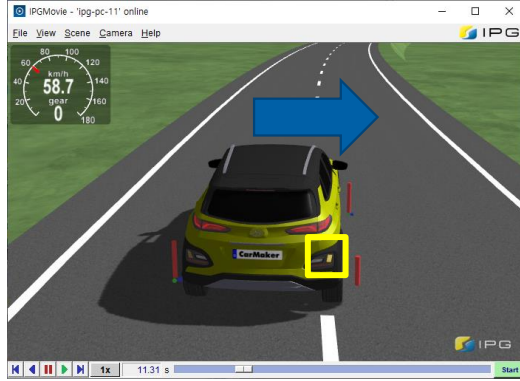
# 소과제 #3

## Minimaneuver 구성 – Minimaneuver No.0

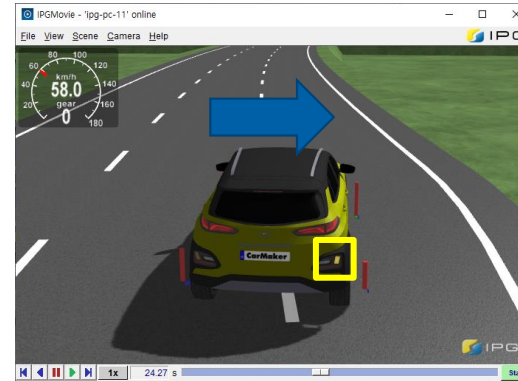
- Minimaneuver No.0
  - 차량의 방향지시등 및 비상등 점등



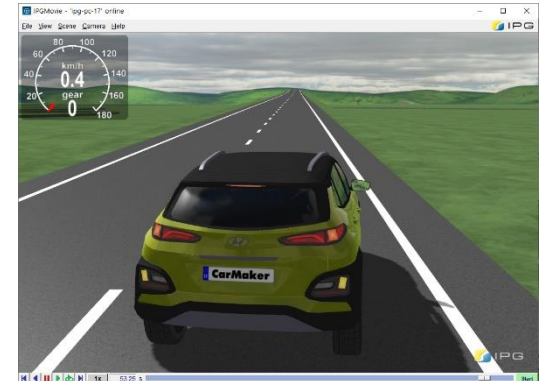
Sin함수를 이용한 차선  
변경(옆 차선 변경)



Sin함수를 이용한 차선  
변경(차선 복귀)



우회전을 위한 차선 변경

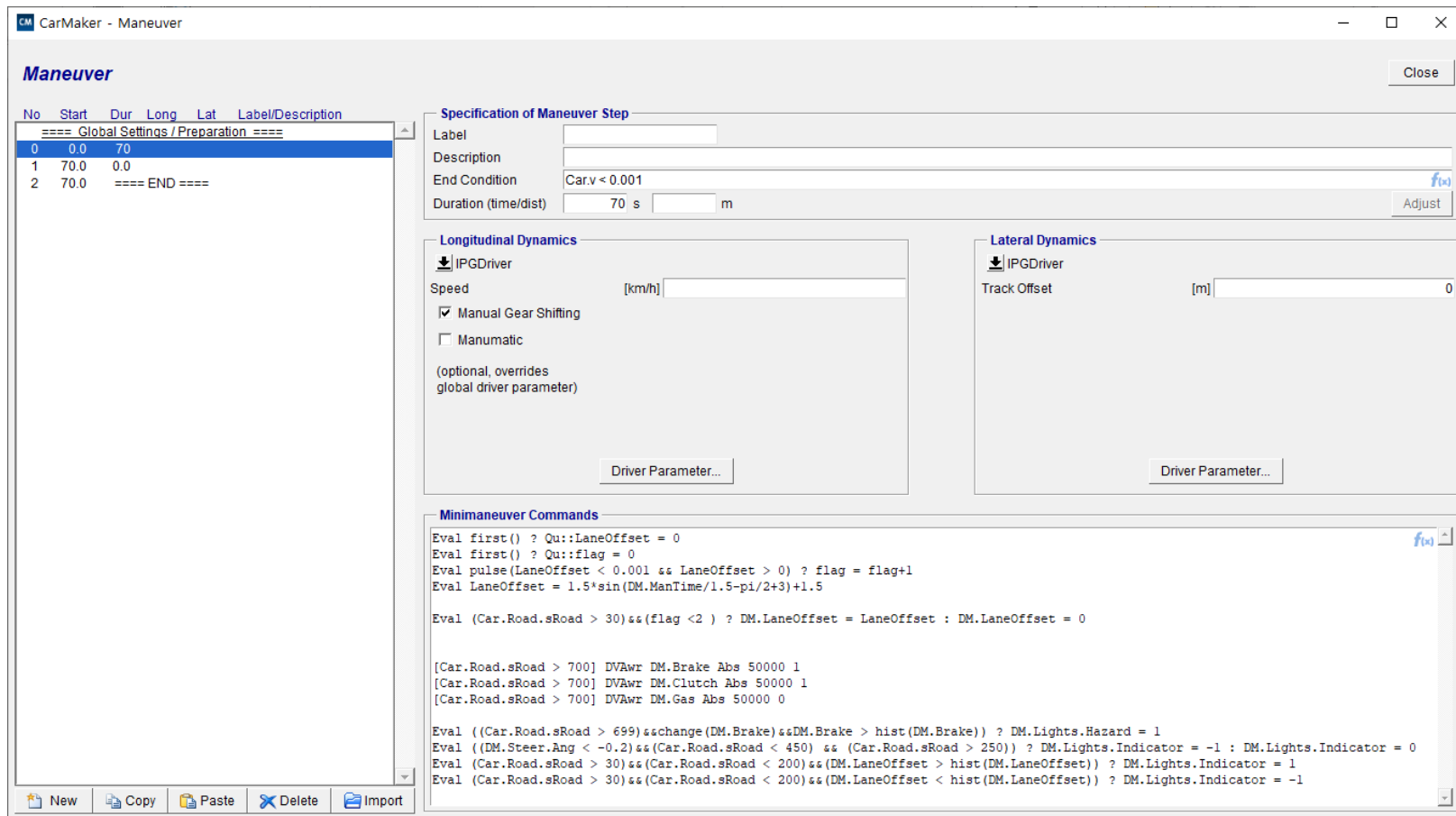


Full Brake 시 비상등 점등

# 소과제 #3

## Minimaneuver 구성 – Minimaneuver No.0

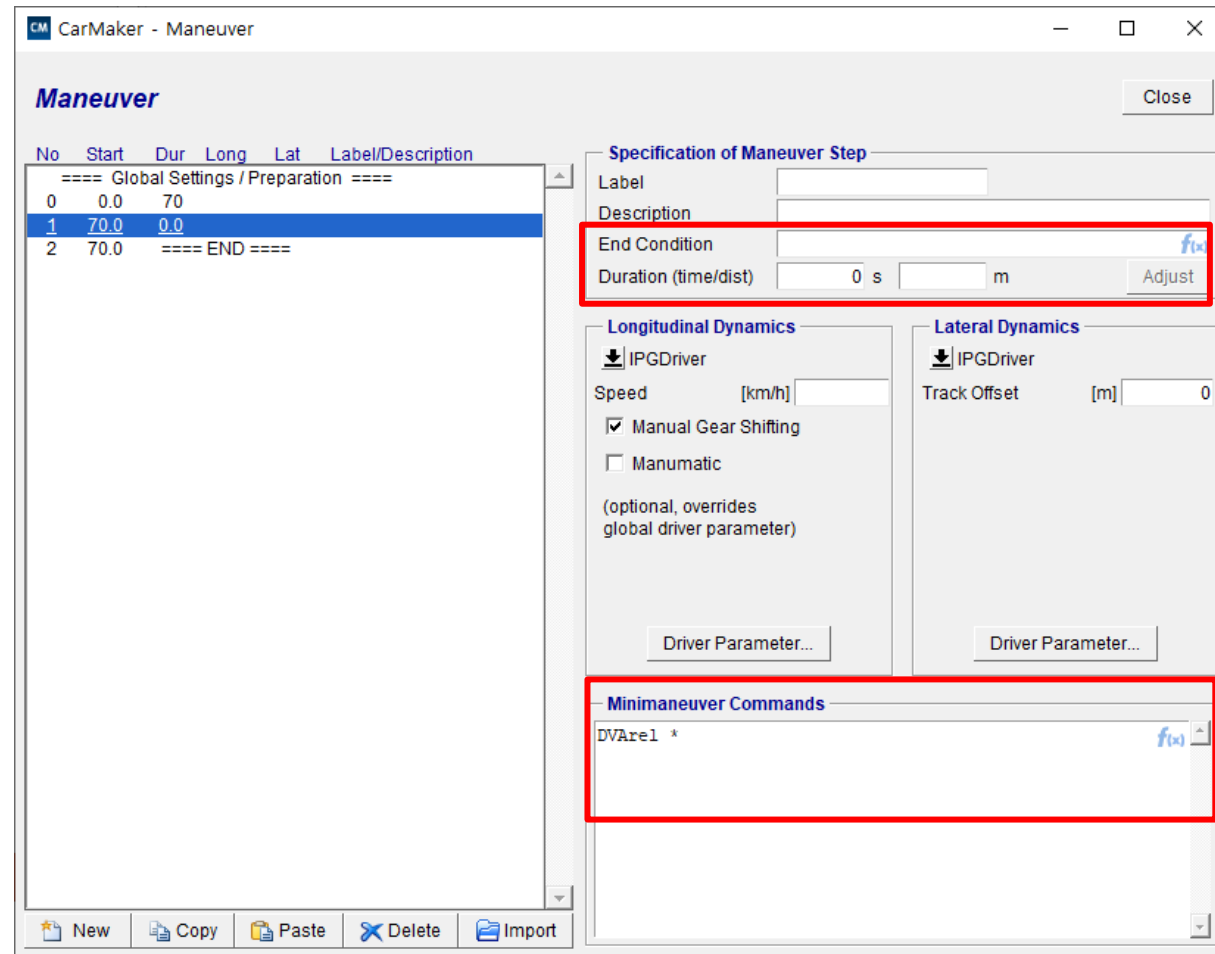
- Minimaneuver No.0
  - 최종적으로 구성된 Minimaneuver No.0



# 소과제 #3

## Minimaneuver 구성 – Minimaneuver No.1

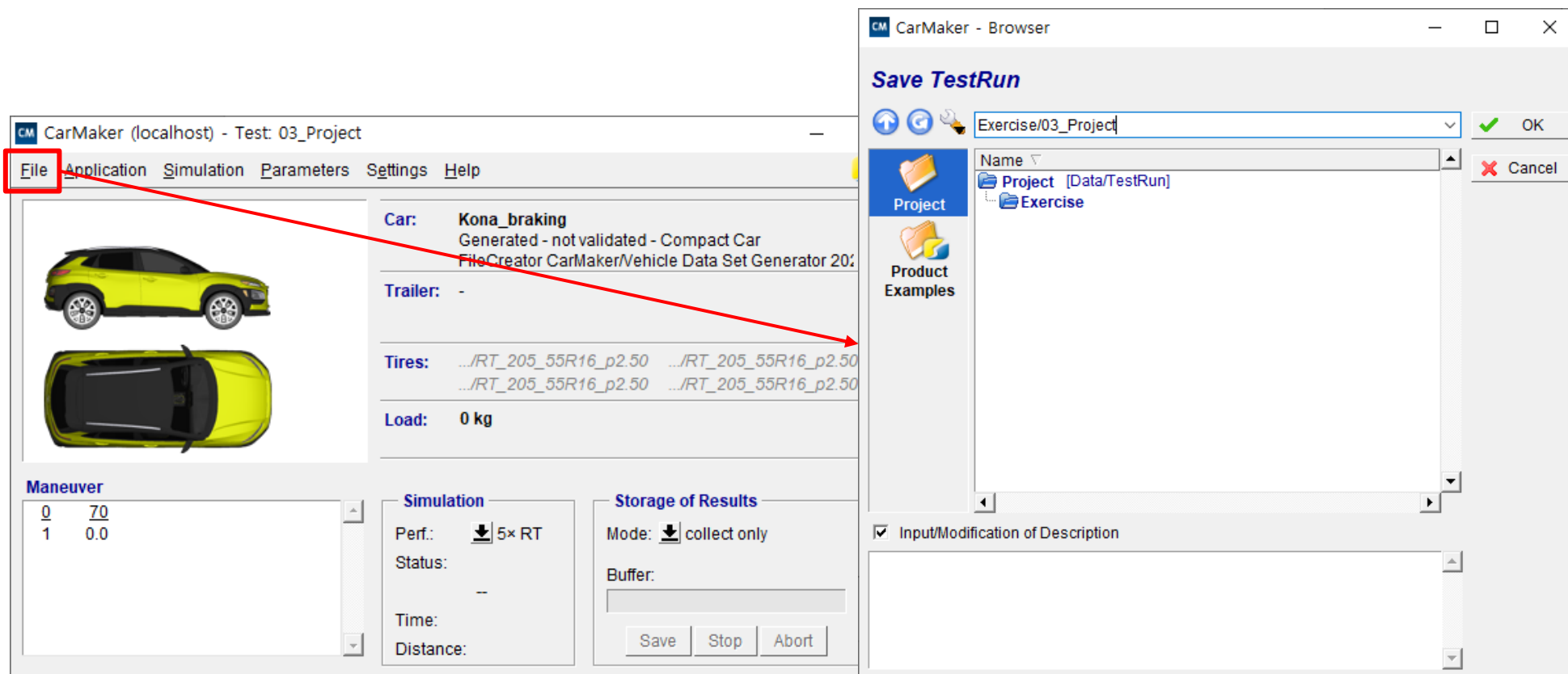
- Minimaneuver No.1
  - DVA command 사용에 따른 Quantities 값 초기화
    - DVA command를 활용할 시 다음 시뮬레이션에 영향을 주지 않도록 사용한 Quantities의 값을 초기화하는 과정이 필요.
    - Duration 0 s인 Minimaneuver에서 초기화 command 작성
    - DVArel \* : DAV Release command



# 소과제 #3

## TestRun 저장

- CarMaker Main GUI > File > save 버튼을 이용하여 구성한 TestRun 저장.
  - 이름은 임의 지정 가능.
  - 한글이 포함되지 않아야 함.



# SOLUTIONS FOR VIRTUAL TEST DRIVING

**Locations:** Germany | China | France | Japan | Korea | Sweden | UK | USA

Sales partners: India | Italy | Taiwan | Turkey



**IPG Automotive GmbH**

Bannwaldallee 60, 76185 Karlsruhe | Germany | Tel.: +49 721 98520 0 | [ipg-automotive.com](http://ipg-automotive.com)  
[sales@ipg-automotive.com](mailto:sales@ipg-automotive.com)

Find us on:

