

# After running any code changes make sure to click the button "Restart Connection" above first.

# Also make sure to click Reset in the simulator to refresh the connection.

```
car_parameters = {"throttle": 0, "steer": 0, "brake": 0}
```

```
def control(pos_x, pos_y, time, velocity):
```

```
    """ Controls the simulated car """
```

```
    global car_parameters
```

```
    # TODO: Use WASD keys in simulator to gain an intuitive feel of parallel parking.
```

```
    # Pay close attention to the time, position, and velocity in the simulator.
```

```
    # TODO: Use this information to make decisions about how to set your car parameters
```

```
    # In this example the car will drive forward for three seconds
```

```
    # and then backs up until its y_pos is less than 32 then comes to a stop by braking
```

```
    if(time < 3):
```

```
        car_parameters["throttle"] = 0.2
```

```
        car_parameters["steer"] = 0
```

```
        car_parameters["brake"] = 0
```

```
    elif(pos_y > 41):
```

```
        car_parameters["throttle"] = -0.5
```

```
        car_parameters["steer"] = 0
```

```
        car_parameters["brake"] = 0
```

```
    elif(pos_y < 41 and pos_x < 126):
```

```
        car_parameters["throttle"] = - 0.8
```

```
        car_parameters["steer"] = 24
```

```
        car_parameters["brake"] = 0
```

```
    elif(pos_y < 35 and pos_y>32.5 and pos_x > 126):
```

```
        car_parameters["throttle"] = - 0.8
```

```
        car_parameters["steer"] = -24
```

```
        car_parameters["brake"] = 0
```

```
    else:
```

```
        car_parameters["throttle"] = 0
```

```
        car_parameters["steer"] = 0
```

```
        car_parameters["brake"] = 1
```

```
    return car_parameters
```

```
import src.simulate as sim
```

```
sim.run(control)
```