A Step-by-Step Guide to

CoSpace Autonomous Driving

(Experience Pack)

RoboCup SINGAPORE

CoSpaceRobot

Learning Point

RoboCup SINGAPORE

CoSpaceRobot

1 Installation

RoboCup SINGAPORE

CoSpaceRobot

System Requirement

Notebook Requirement:

Intel i5 or i7

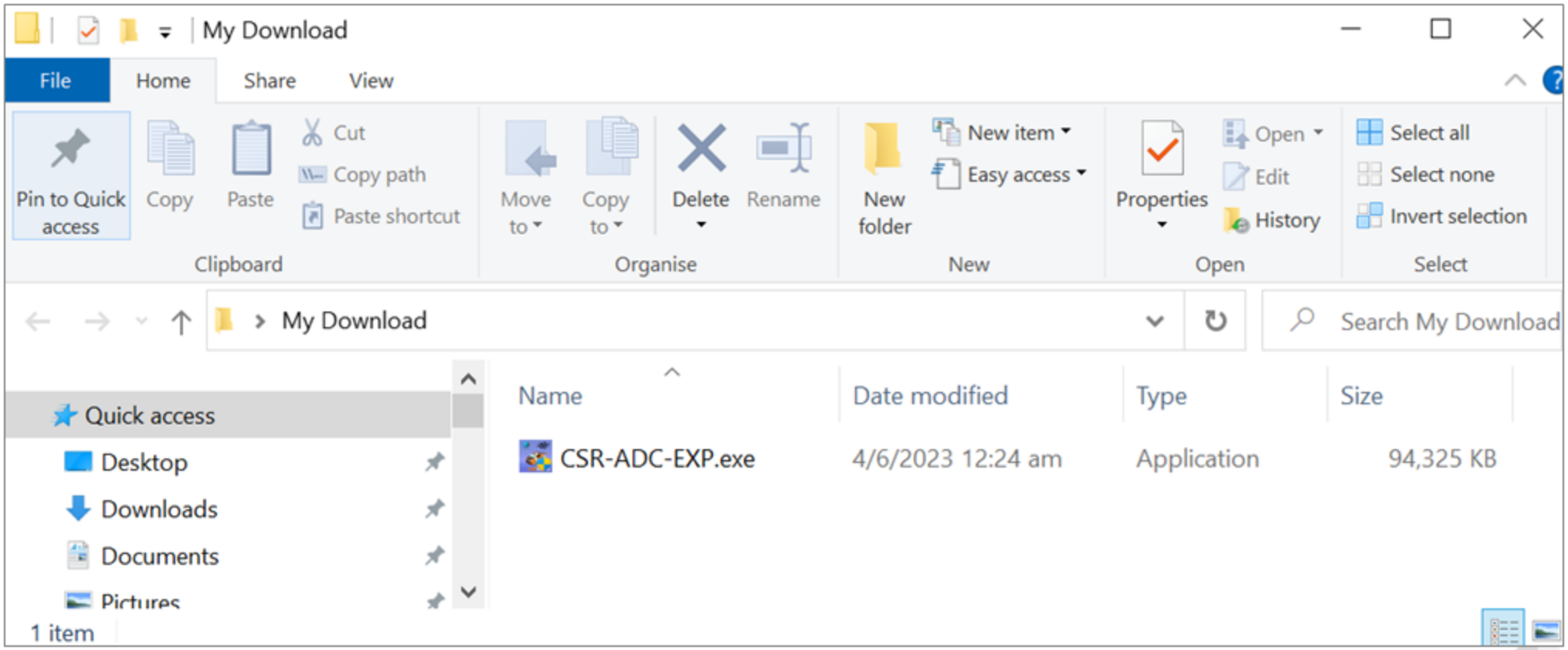
Operating System:

Windows 10 or 11

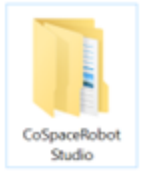
RoboCup SINGAPORE

CoSpaceRobot

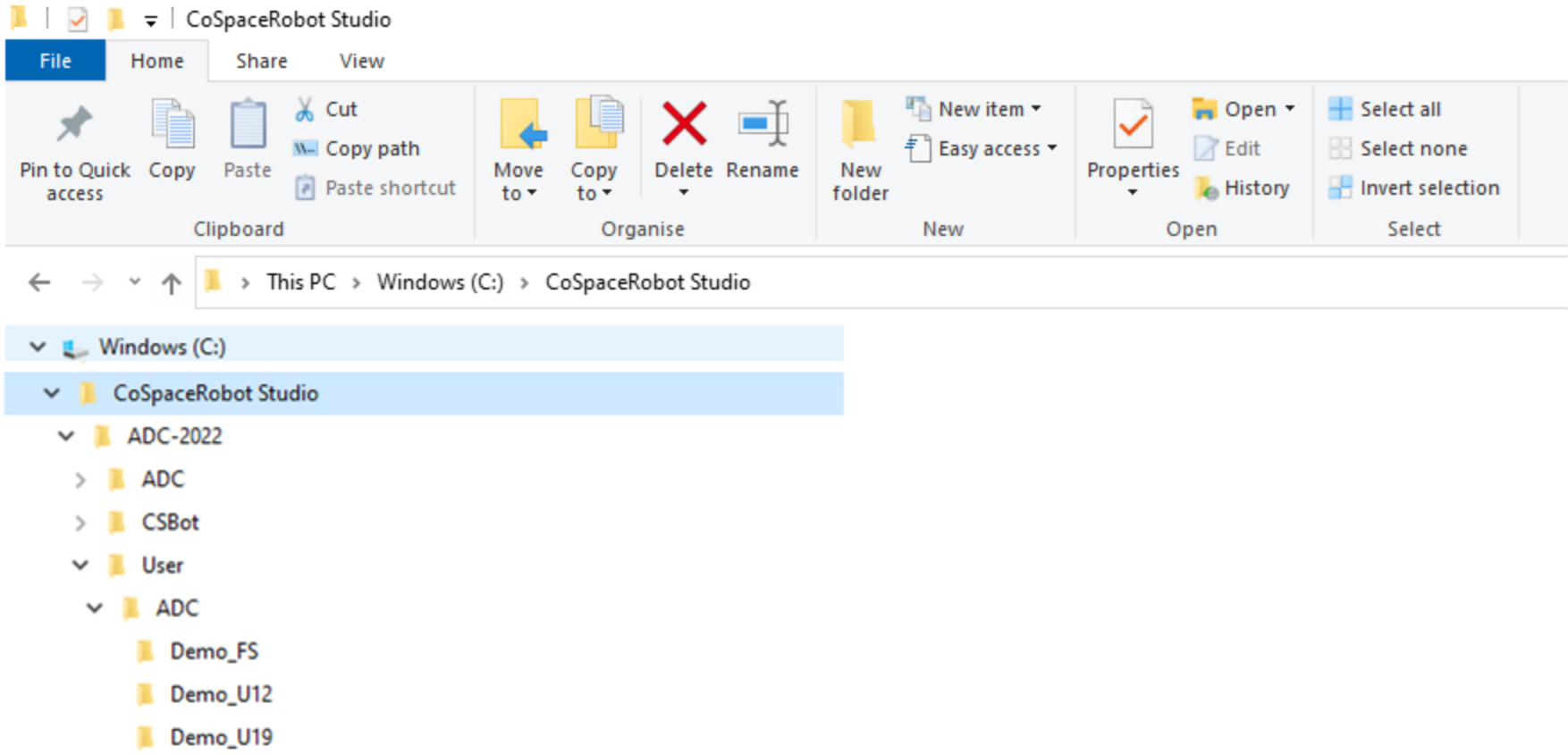
Please download the CoSpace Autonomous Driving installation file and save it on your hard disk.



Installation

1) Download the CoSpace Auto-Driving Simulator
2) Double click on  CSR-Auto-Driving.exe
3) After installation, a shortcut will be created on desktop 
4) A folder “CoSpaceRobot Studio” will be created. 
5) A sub-folder “ADC-2022” will also be created in folder “CoSpaceRobot Studio”

1. “ADC-2022” contains all files for the Autonomous Driving simulator.
2. Your own codes will be saved in “C:\CoSpaceRobot Studio\ADC-2022\User\ADC\”



2

Launch CoSpace Auto-Driving Experience Pack

RoboCup SINGAPORE

CoSpaceRobot

Launch CoSpace Auto-Driving Experience Pack

To access learning resources

To access RCAP Academy website

To contact cospace@robocupap.org

News / Event List

To launch CoSpace Simulator

RoboCup SINGAPORE

CoSpaceRobot

Virtual Environment Adjustment

2. **Press the Q, W, E, A, S, D keys**

Up Zoom-in Down

Left Zoom-out Right

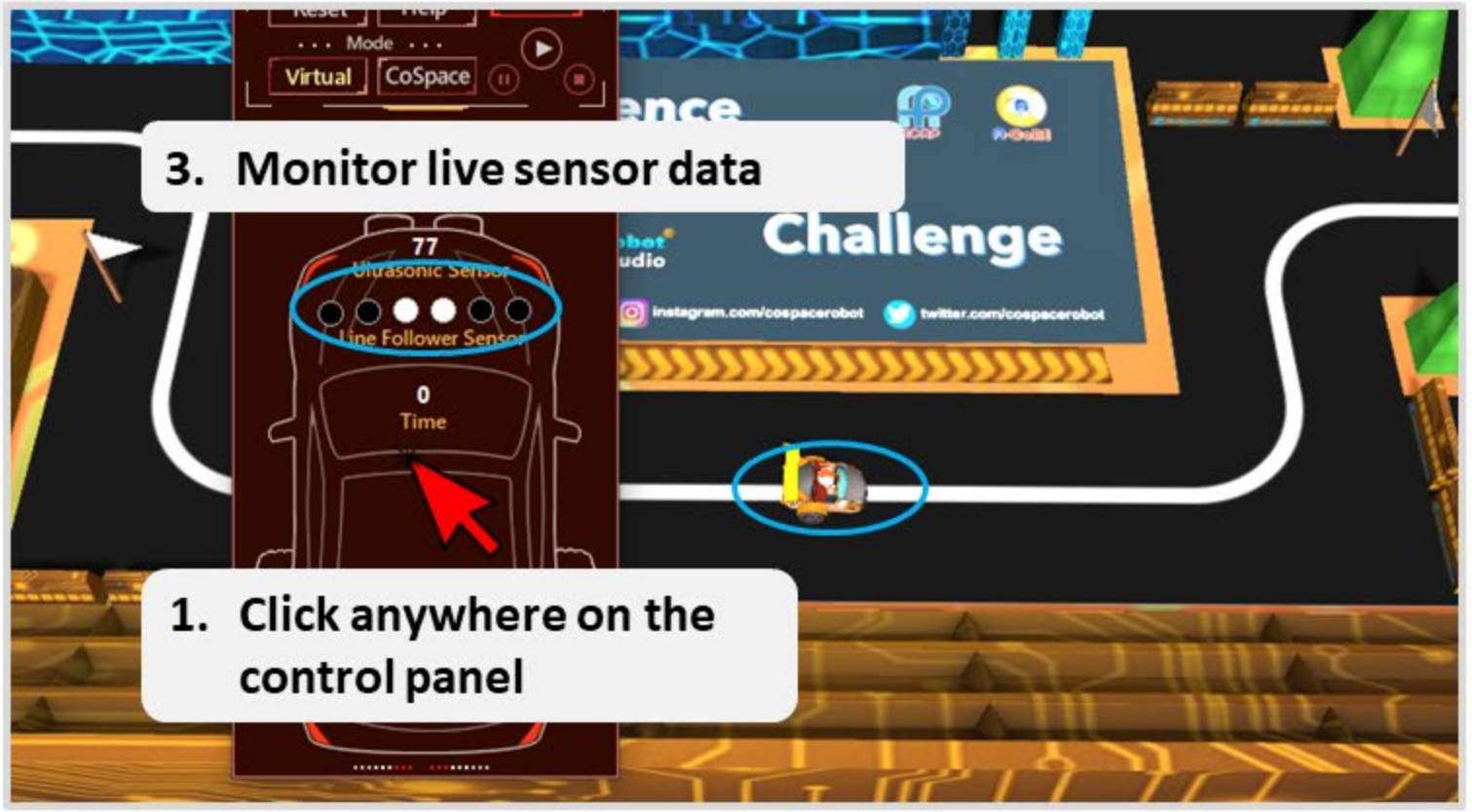
Turn 360°

3. **Press "Shift + the keys" to make the movement faster.**

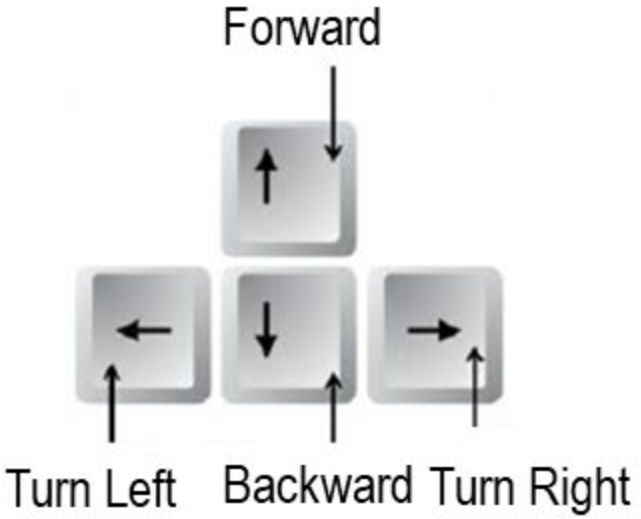
RoboCup SINGAPORE

CoSpaceRobot

Manual Control of Virtual Robot



2. **Press the arrow keys below**



RoboCup SINGAPORE

CoSpaceRobot

3

My First CoSpace Program

RoboCup SINGAPORE

CoSpaceRobot

My First Program

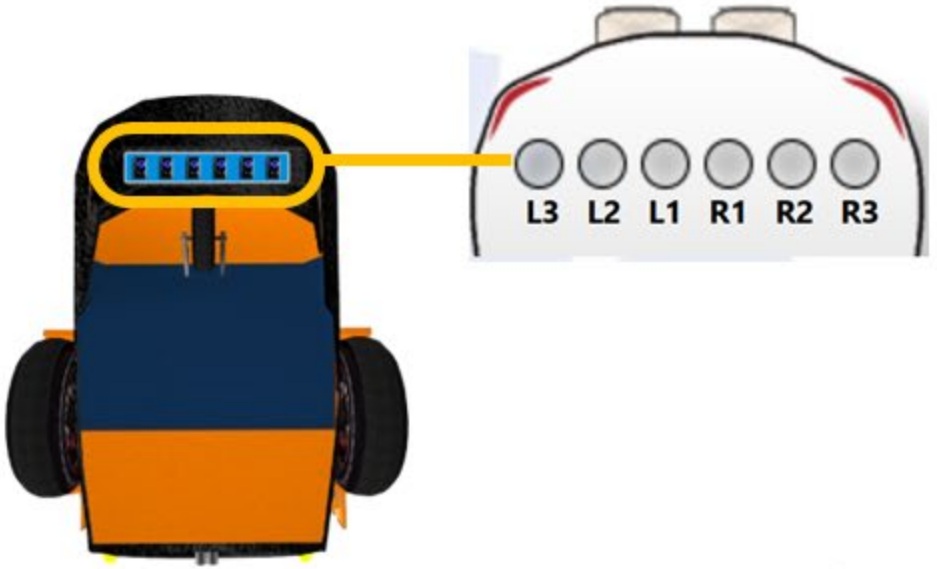
To program our robot to follow the line and travel towards the finish line.

Controlling of a Driverless Car

Real Situation

Controlling of a Driverless Car

A 6-IR sensor array is mounted at the bottom of the virtual robot to sense / detect black or white coloured surface.

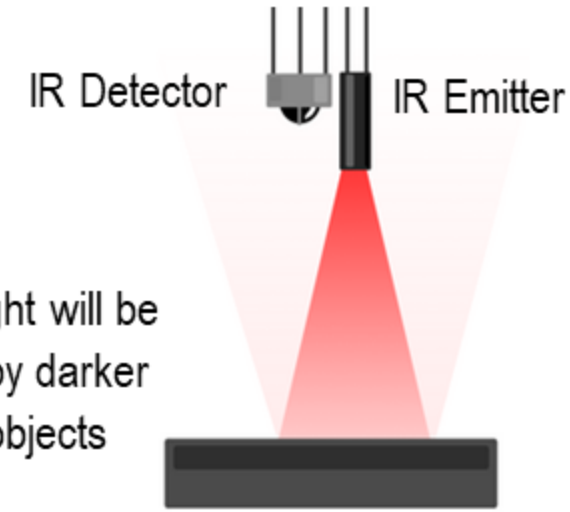


RoboCup SINGAPORE

CoSpaceRobot

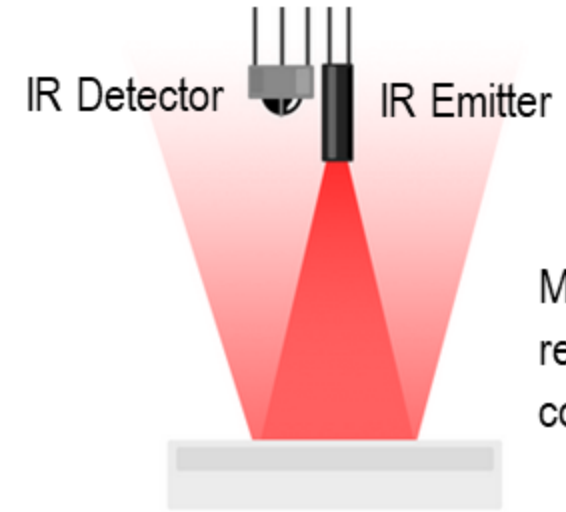
How Infrared Sensor Works?

Less IR light will be reflected by darker coloured objects



Dark Surface

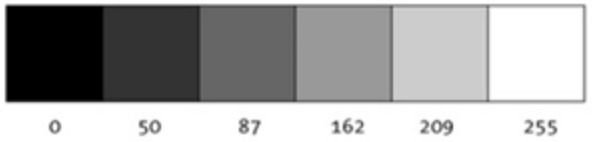
(Dark surface absorbs Infrared light)

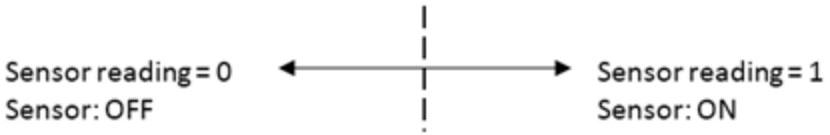


Bright Surface

(Bright surface bounces Infrared light)

More IR light will be reflected by light coloured objects

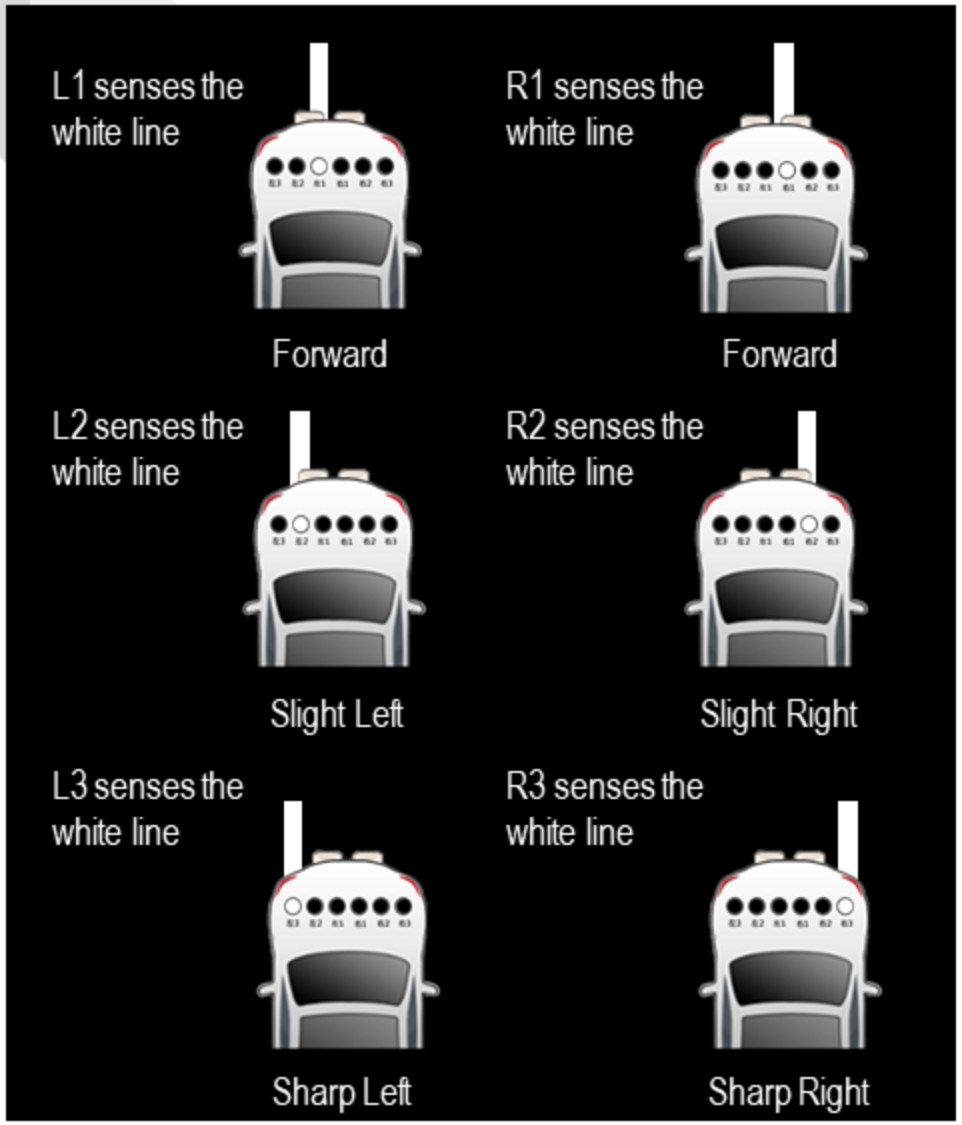


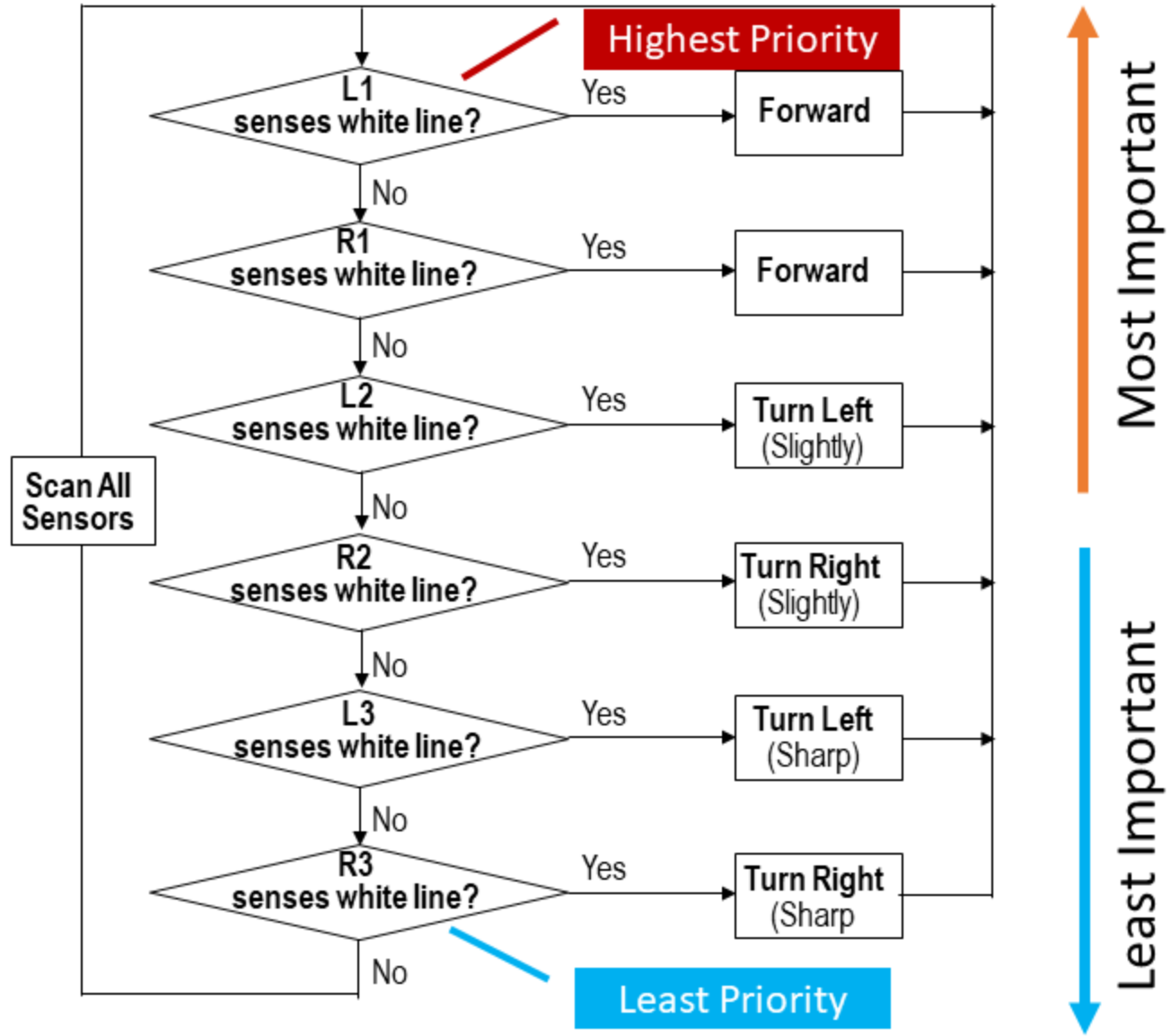


RoboCup SINGAPORE

CoSpaceRobot

Line Tracking Logics





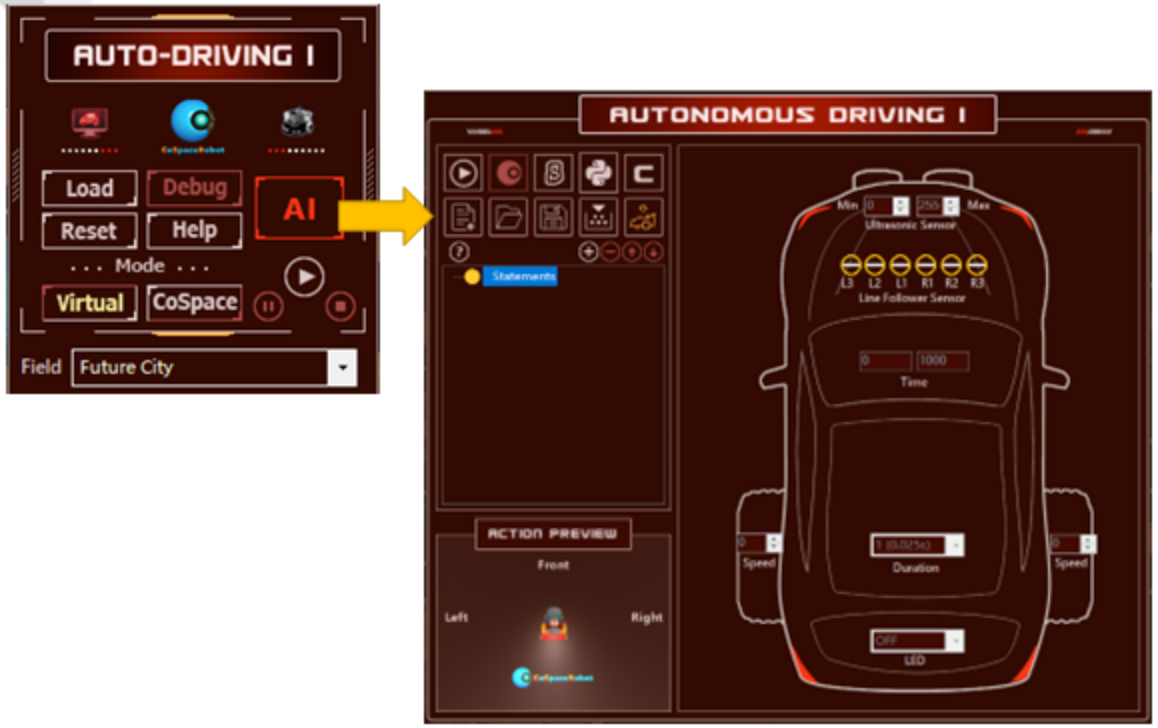
My First Program (Step-by-Step)

Virtual Map - Practice: Future City

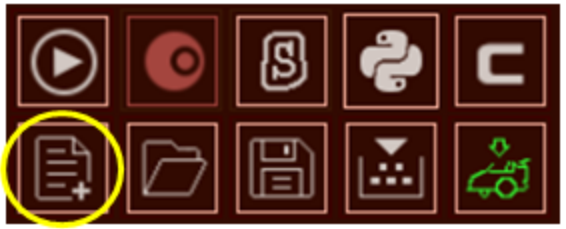
Step 1: Launch CoSpace Auto-Driving

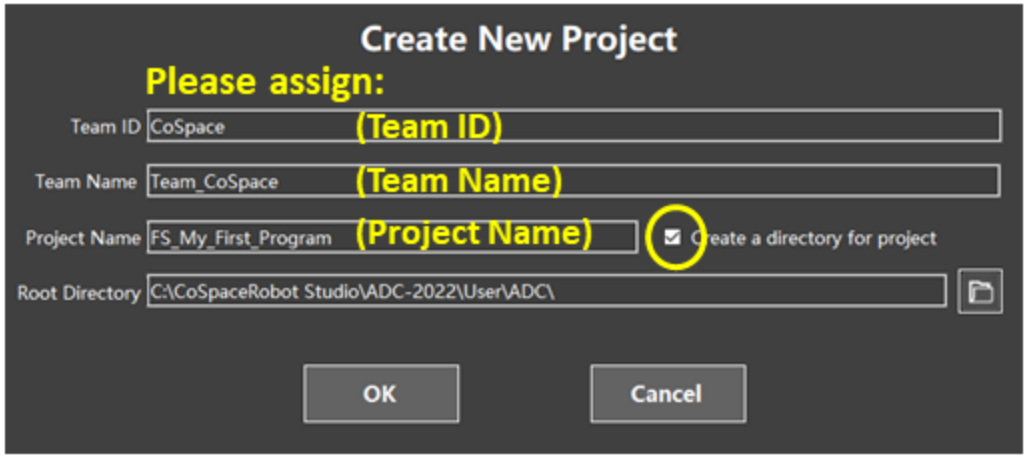
My First Program (Step-by-Step)

Step 2: Launch AI Panel



Step 3: Create a New Project





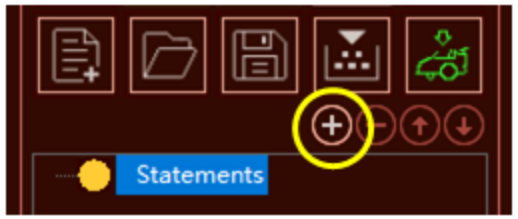
My First Program (Step-by-Step)

Step 4: Add the 1st Statement

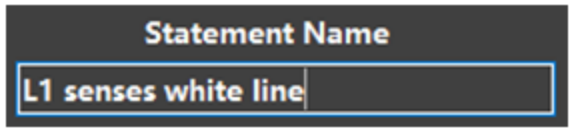
If L1 senses the white line, the robot moves forward

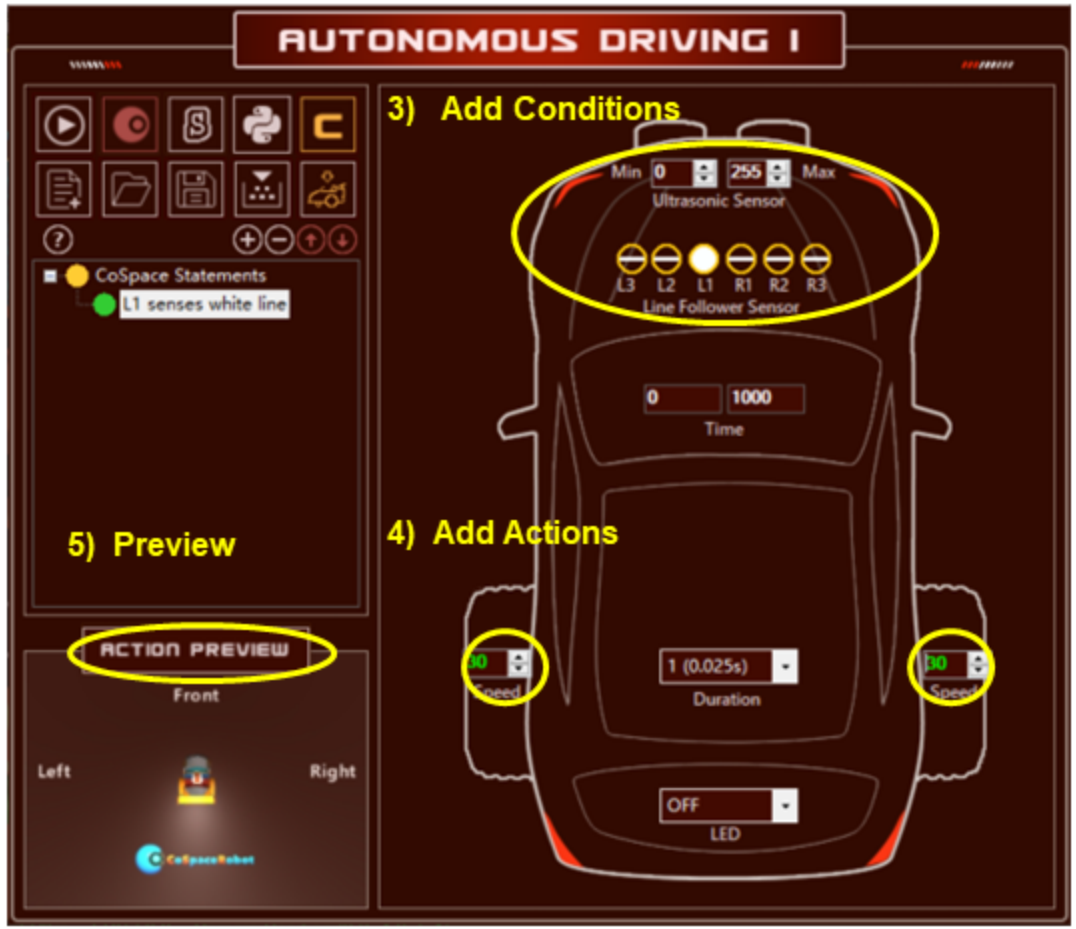
```
If  ( ⦸⦸○⦸⦸⦸ )
      {
         WheelLeft = +30;
         WheelRight = +30;
      }
```

1) Select “root” and click on “+”



2) Give a statement name:





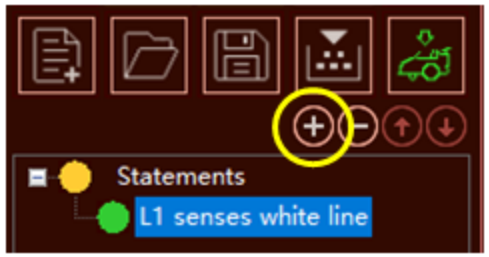
My First Program (Step-by-Step)

Step 5: Add the 2nd Statement

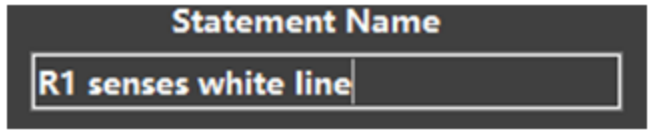
If R1 senses the white line, the robot moves forward

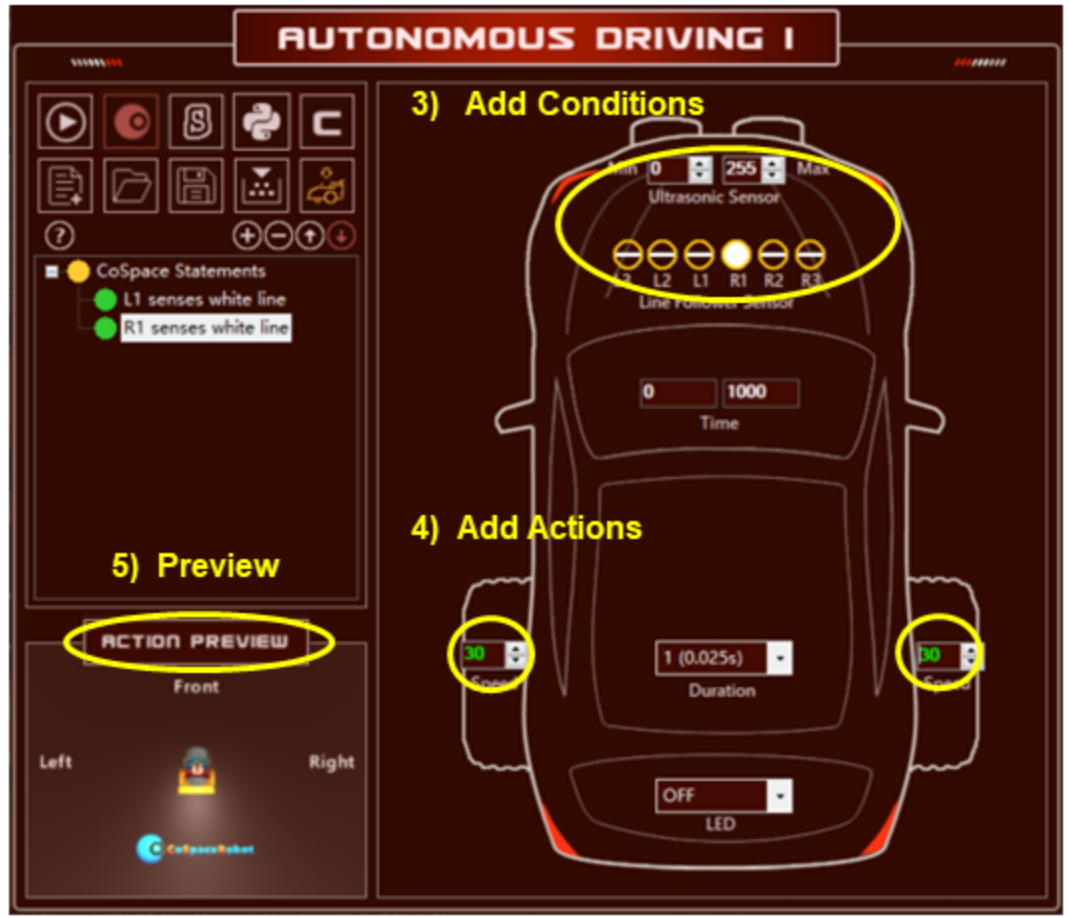
```
If  ( ⦸⦸⦸O⦸⦸ )
      {
          WheelLeft = +30;
          WheelRight = +30;
      }
```

1) Select the 1st statement and click on "+"



2) Give a statement name:





My First Program (Step-by-Step)

Step 6: Add The Rest of Statements

S/N	Statement Name	IR	Left wheel	Right wheel	LED	Action Time
1	L1 senses white line		+30	+30	OFF	0.025 s
2	R1 senses white line		+30	+30	OFF	0.025 s
3	L2 senses white line		+10	+30	FLASH	0.025 s
4	R2 senses white line		+30	+10	FLASH	0.025 s
5	L3 senses white line		-20	+30	ON	0.025 s
6	R3 senses white line		+30	-20	ON	0.025 s

Note: Wheel speed can only be the multiple of 10s in between -100 to 100, such as 10, 20, 30, -10, -20, -30, 0, etc.

Sample Code: My_First_Program

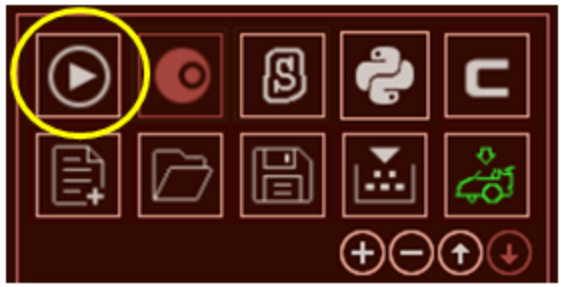
RoboCup SINGAPORE

CoSpaceRobot

My First Program (Step-by-Step)

Step 7: Save Project

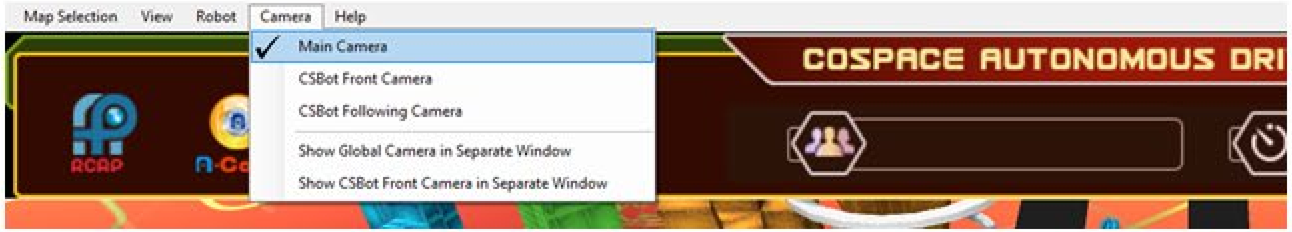
Step 8: Trial Run



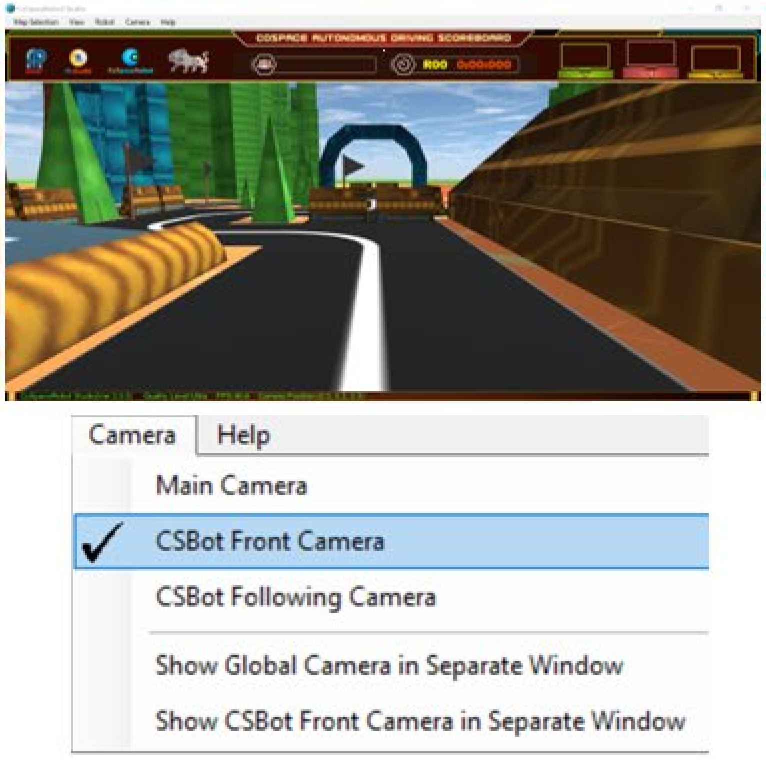
Step 9: Monitor Robot Performance

My First Program (Step-by-Step)

Step 10: Changing Viewing Perspectives

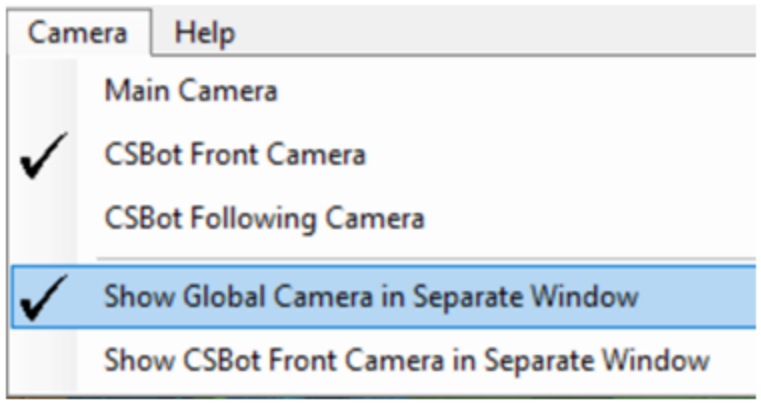


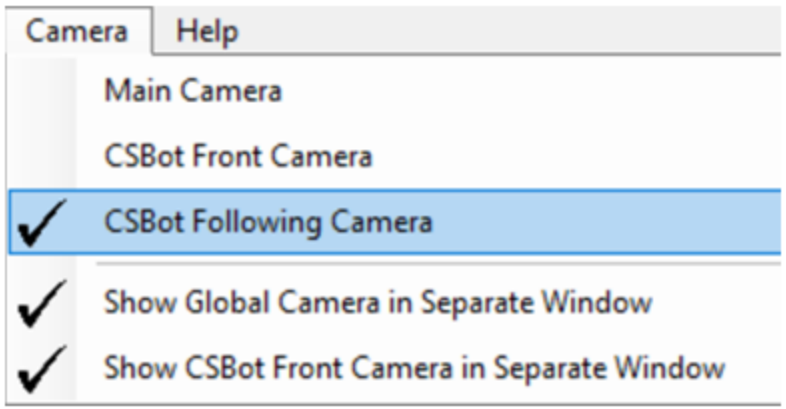
You can select different camera viewpoints to monitor the robot’s movement from different viewing angles.



My First Program (Step-by-Step)

Step 10: Changing Viewing Perspectives

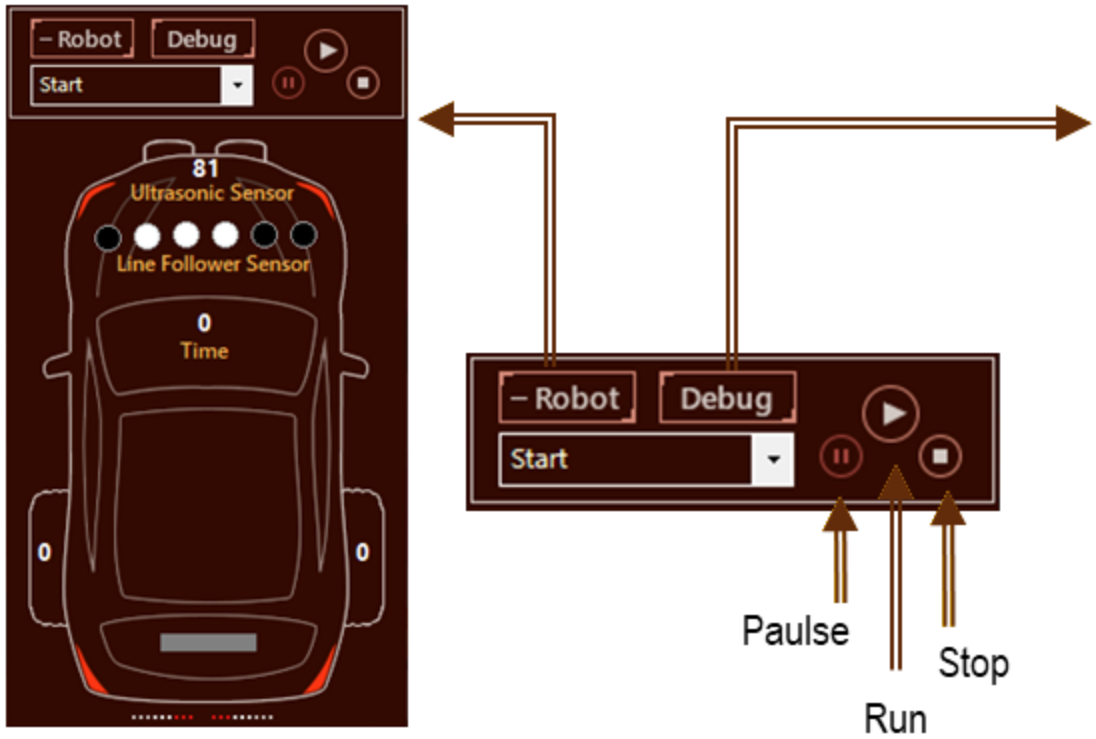




You can choose different viewing perspectives

My First Program (Step-by-Step)

Step 11: You Can Access "Debug Panel" to Monitor All Variables, If You Wish To Do So



Debug Information

Settings

Debug infomation is updated ONLY when AI is running.

Variable	Value	
Duration	0	Duration set in action
CurAction	-1	Current Action ID. If the 2nd statement is executed, CurAction = 2
MyState_1	0	System variable
US_Front	0	Distance between the front obstacle and ultrasonic sensor. Unit: cm
IR_L3	0	6 IR sensors' live value: For example: IR = 0, it senses black line. IR = 1, it senses white line.
IR_L2	0	
IR_L1	0	
IR_R1	0	
IR_R2	0	
IR_R3	0	
CS_R	0	System variables
CS_G	0	
CS_B	0	
RotationX	0	
RotationY	0	
RotationZ	0	
Time	0	Time elapsed from starting of the robot run
WheelLeft	0	Speed of the left wheel
WheelRight	0	Speed of the right wheel
LED_1	0	LED status. LED=0 means the LED is OFF

My First Program (Step-by-Step)

Step 12: You Can Also "Build Project", Load Project From Control Panel, and Execute Project

Build the project

After building, there will be 3 files in the project folder:

1) My_First_Program.smp
2) My_First_Program.dll
3) My_First_Program.c

Path: C:\CoSpaceRobot Studio\ADC-2022\User\ADC\My_First_Program

Load the project

Load "**My_First_Program.dll**" file

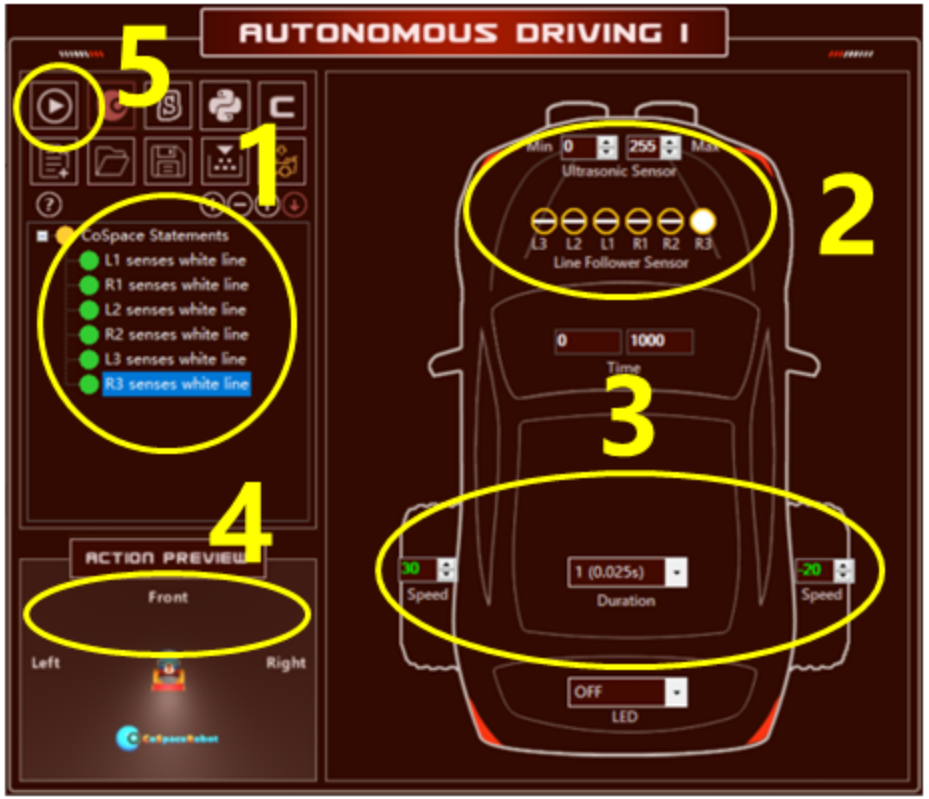
Set a virtual game Pause / Run / Stop

My First Program (Summary)

Summary

Writing CoSpace AI, Steps:

1. Define statement
2. Set conditions
3. Set actions
4. Preview
5. Trial Run



4

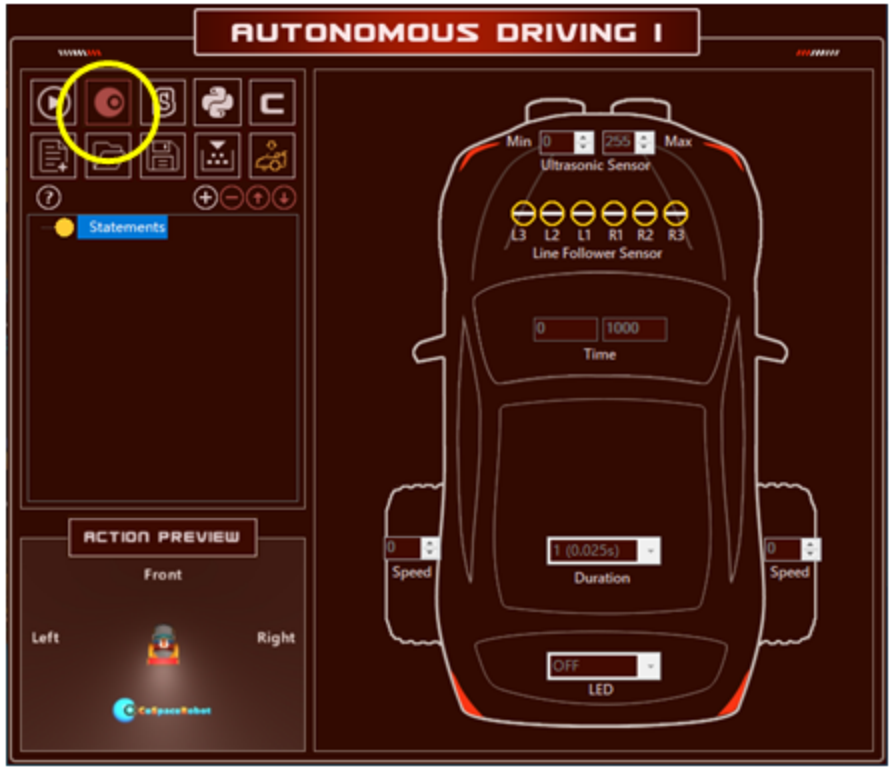
Code in Scratch

RoboCup SINGAPORE

CoSpaceRobot

Code in Scratch

CoSpace Auto-Driving Simulator allows user to code in Scratch. Users can also convert code in CoSpace and Scratch easily.



CoSpace GUI Coding

Scratch Coding

RoboCup SINGAPORE

CoSpaceRobot

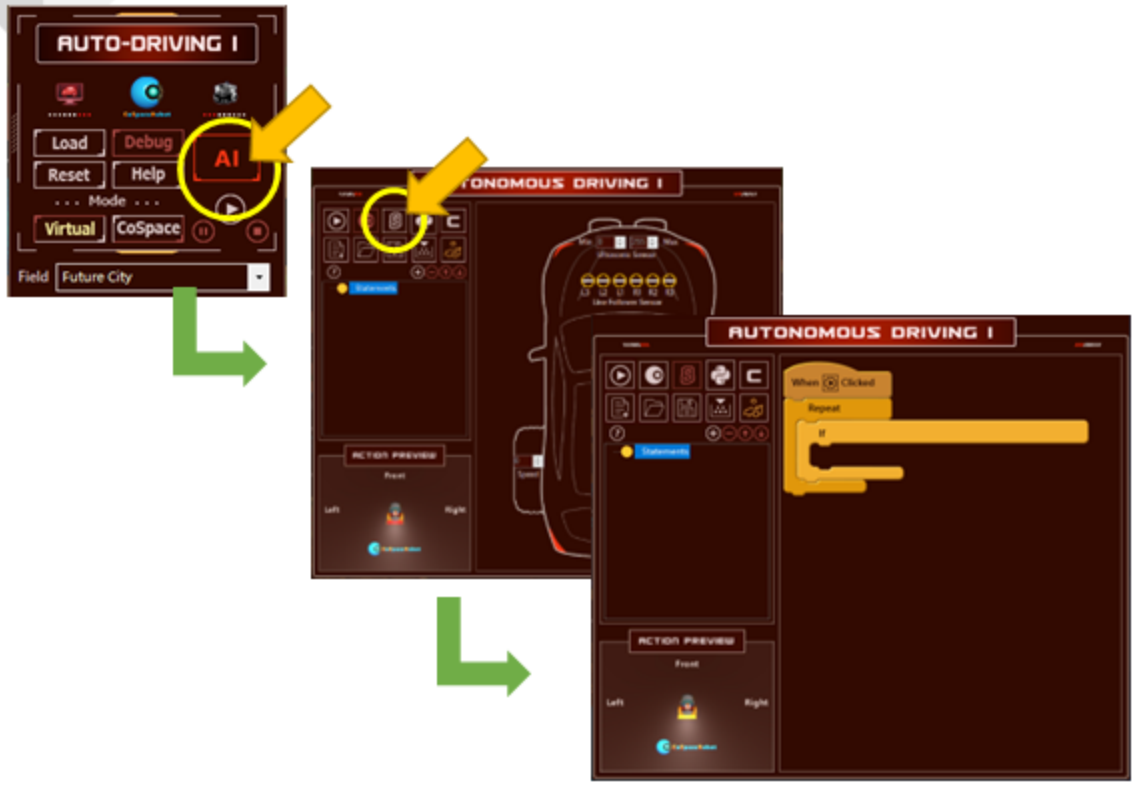
My Scratch Program (Step-by-Step)

Virtual Map - Practice: Future City

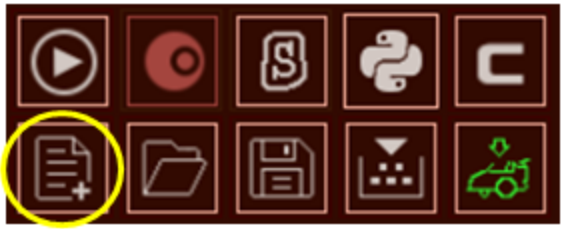
Step 1: Launch CoSpace Auto-Driving

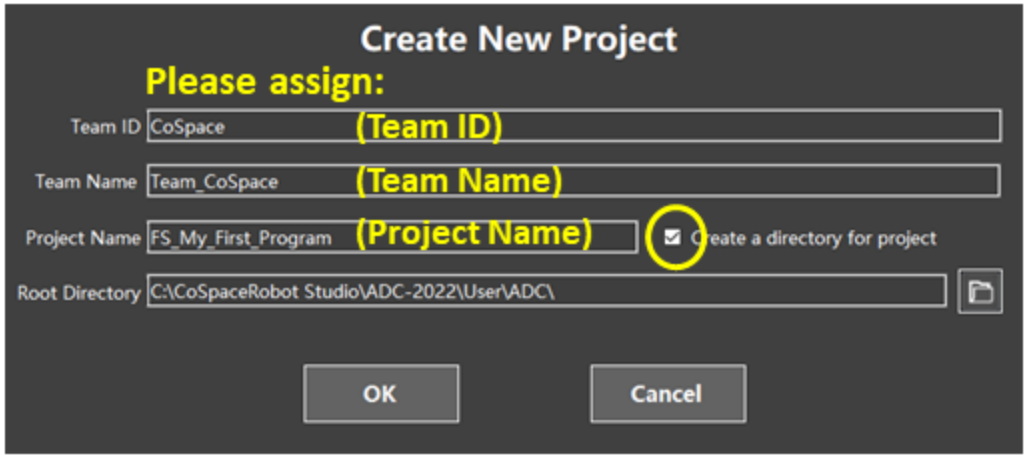
My Scratch Program (Step-by-Step)

Step 2: Launch AI Panel



Step 3: Create a New Project





RoboCup SINGAPORE

CoSpaceRobot

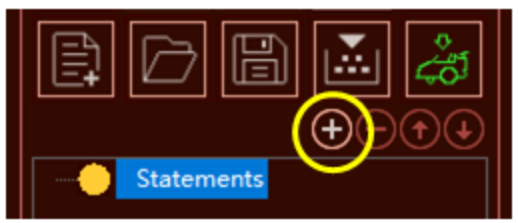
My Scratch Program (Step-by-Step)

Step 4: Add the 1st Statement

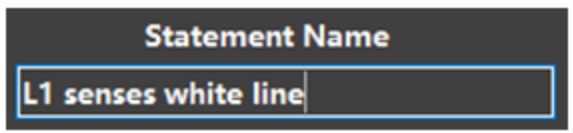
If L1 senses the white line, the robot moves forward

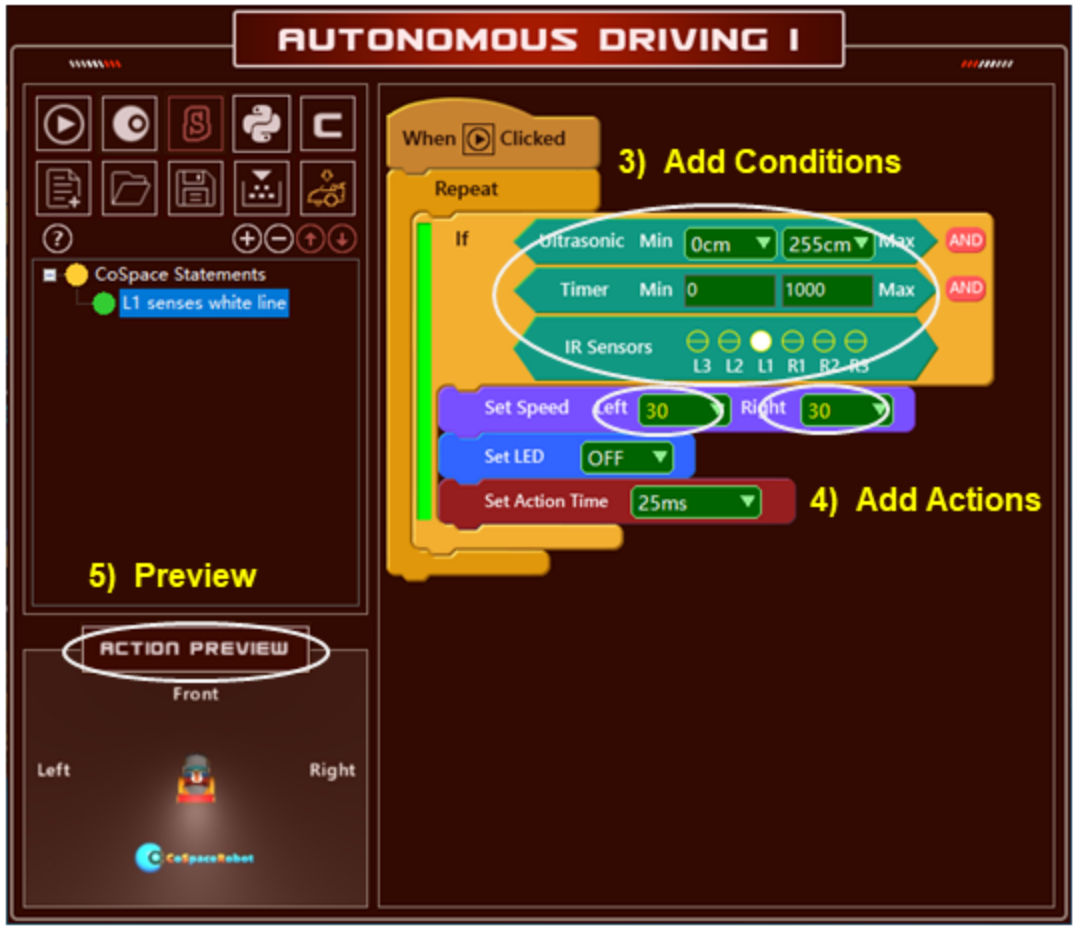
```
If  ( ⦸⦸O⦸⦸⦸ )
      {
         WheelLeft = +30;
         WheelRight = +30;
      }
```

1) Select “root” and click on “+”



2) Give a statement name:





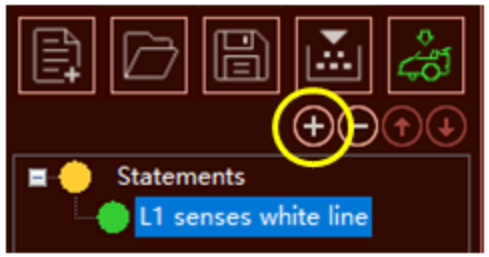
My Scratch Program (Step-by-Step)

Step 5: Add the 2nd Statement

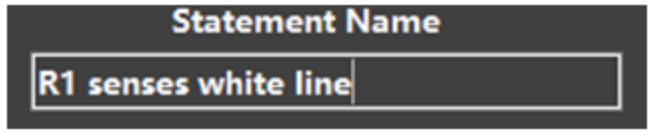
If R1 senses the white line, the robot moves forward

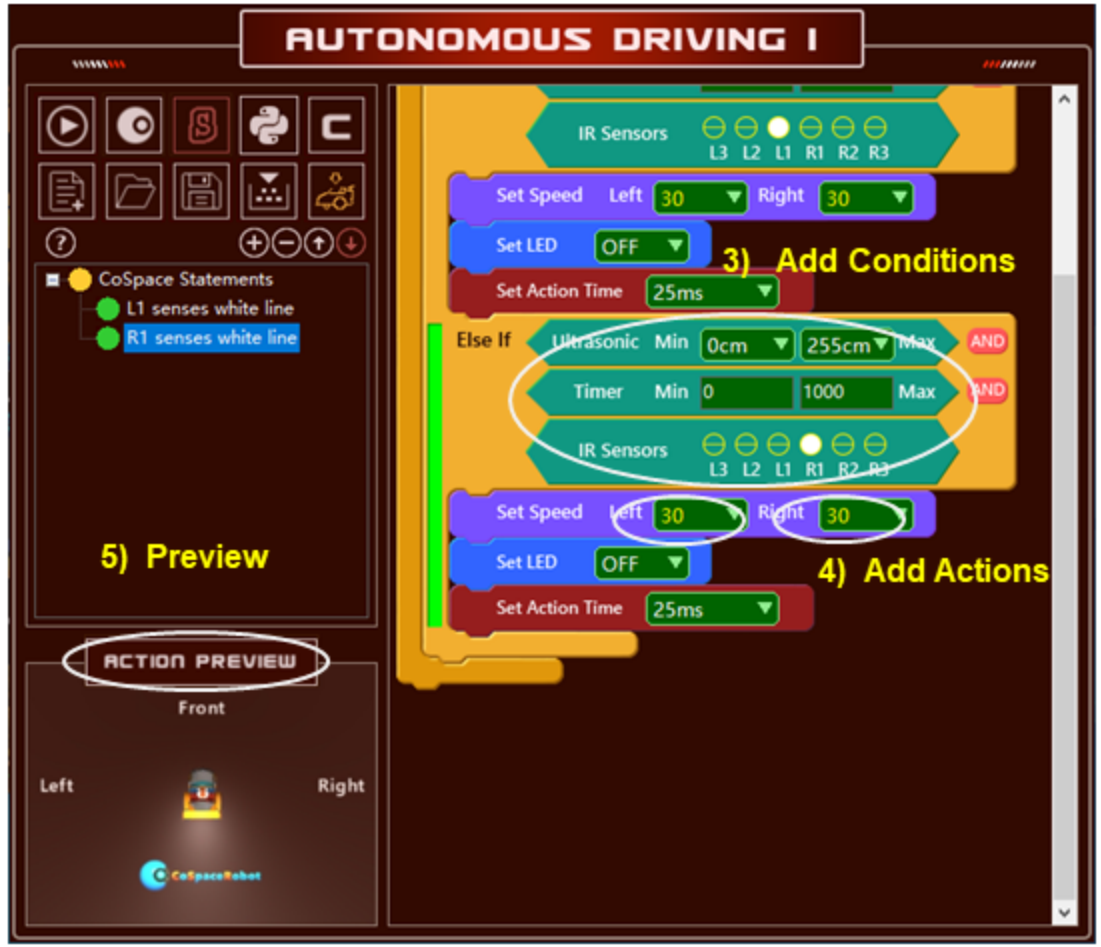
```
If  ( ⦸⦸⦸O⦸⦸ )
      {
          WheelLeft = +30;
          WheelRight = +30;
      }
```

1) Select the 1st statement and click on "+"



2) Give a statement name:





My Scratch Program (Step-by-Step)

Step 6: Add The Rest of Statements

S/N	Statement Name	IR	Left wheel	Right wheel	LED	Action Time
1	L1 senses white line		+30	+30	OFF	0.025 s
2	R1 senses white line		+30	+30	OFF	0.025 s
3	L2 senses white line		+10	+30	FLASH	0.025 s
4	R2 senses white line		+30	+10	FLASH	0.025 s
5	L3 senses white line		-20	+30	ON	0.025 s
6	R3 senses white line		+30	-20	ON	0.025 s

Note: Wheel speed can only be the multiple of 10s in between -100 to 100, such as 10, 20, 30, -10, -20, -30, 0, etc.

Sample Code: My_First_Program

RoboCup SINGAPORE

CoSpaceRobot

5

Experience Python Code

RoboCup SINGAPORE

CoSpaceRobot

Code in Python

CoSpace Auto-Driving Simulator allows user to code in Python. The simulator will auto-convert the CoSpace / Scratch code in Python. You can just simply modify the Python code to modify the robot's movement. In advanced level, you can also write your own Python code.

Code in CoSpace

```
################################################################################
#                                                                              #
#            This Python Code is auto-generated by CoSpace Robot               #
#                          All Right Reserved.                                 #
#                                Jan 2021                                      #
#                                                                              #
################################################################################


#-------------------------------------------------------------------------------
# FUNCTION : GetTeamName
# DON'T modify the function definition.
#-------------------------------------------------------------------------------
# SUMMARY
# Called by the the system to get the team name.
# PARAMETERS
# RETURNS
# Team name string
#-------------------------------------------------------------------------------
def GetTeamName():
    return "CoSpace" # PLEASE Change 'CoSpace' to your Team Name


MyState_1 = 0 # System global variable.
# ------------------------- You can add your self-defined global variables below this line
# For Example :
# MyVar = 0


#-------------------------------------------------------------------------------
# FUNCTION : GetDebugVarName
# DON'T modify the function definition.
#-------------------------------------------------------------------------------
# SUMMARY
# Called by System to gets variables' names to be watched in the debug panel
# PARAMETERS
```

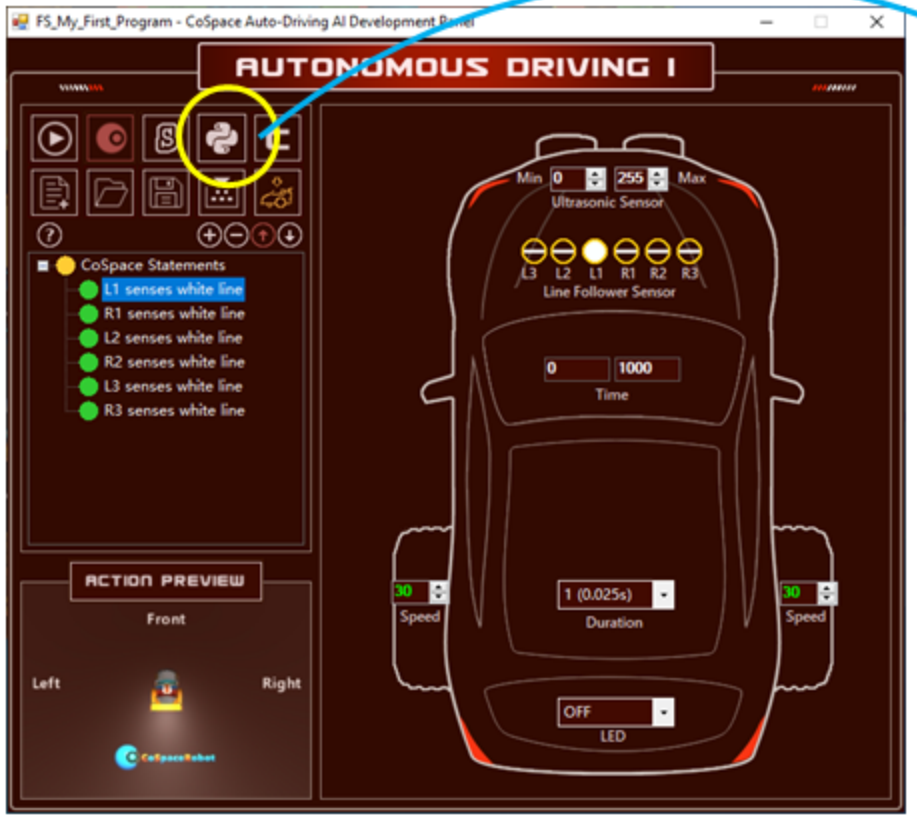
Code in Python

RoboCup SINGAPORE

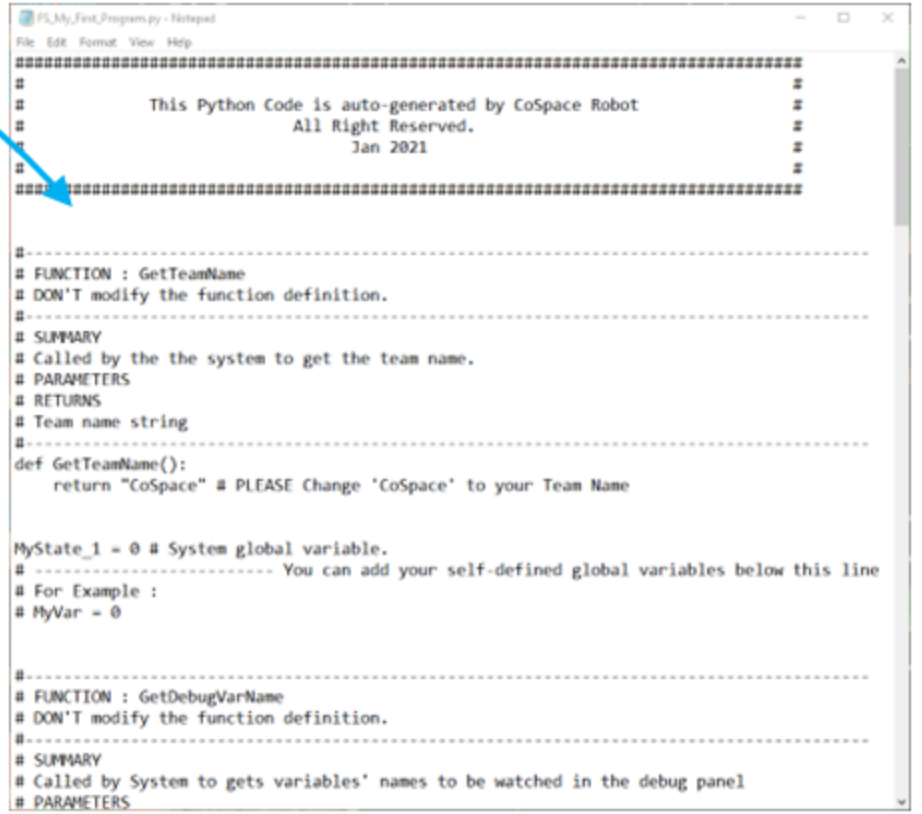
CoSpaceRobot

Experience Python Code

Step 1: Open a Line Tracking Project



Step 2: Click on “Python” to get the Python code.



```
FS_My_First_Program.py - Notepad
File  Edit  Format  View  Help
##############################################################################
#                                                                            #
#            This Python Code is auto-generated by CoSpace Robot             #
#                          All Right Reserved.                               #
#                               Jan 2021                                     #
#                                                                            #
##############################################################################


#-----------------------------------------------------------------------------
# FUNCTION : GetTeamName
# DON'T modify the function definition.
#-----------------------------------------------------------------------------
# SUMMARY
# Called by the the system to get the team name.
# PARAMETERS
# RETURNS
# Team name string
#-----------------------------------------------------------------------------
def GetTeamName():
    return "CoSpace" # PLEASE Change 'CoSpace' to your Team Name


MyState_1 = 0 # System global variable.
# ------------------------- You can add your self-defined global variables below this line
# For Example :
# MyVar = 0


#-----------------------------------------------------------------------------
# FUNCTION : GetDebugVarName
# DON'T modify the function definition.
#-----------------------------------------------------------------------------
# SUMMARY
# Called by System to gets variables' names to be watched in the debug panel
# PARAMETERS
```

RoboCup SINGAPORE

CoSpaceRobot

Experience Python Code

Step 3: Modify the Python Code

The auto-converted python code has the following structurer:

`GeTeamName():` to get the team name

`GetDebugVarName(idx):` to get the variables to be watched in debug panel

`AI_Loop(US_Front, IR_L3, IR_L2, IR_L1, IR_R1, IR_R2, IR_R3, CS_R, CS_G, CS_B, RotationX, RotationY, RotationZ, Time)`

You should or modify the code or write your own code in AI_Loop function

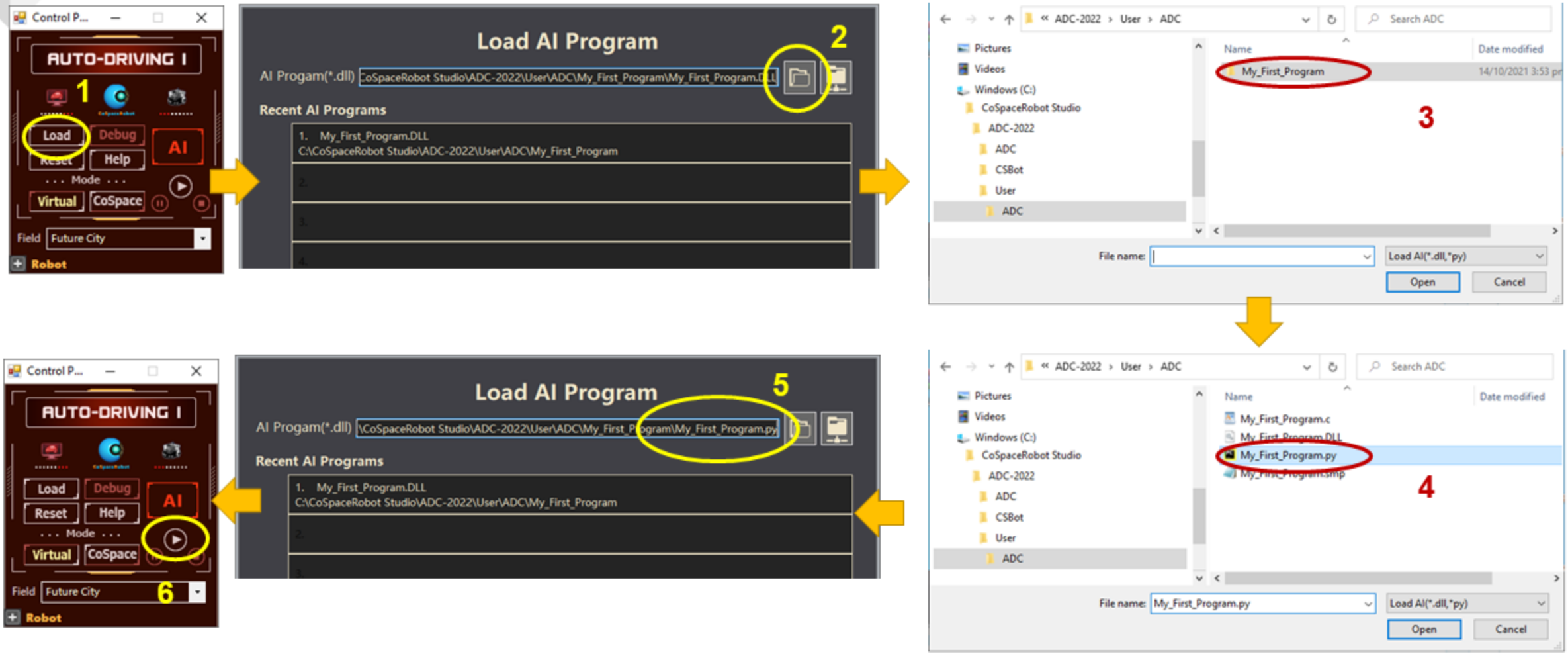
For example, if you wish to increase the forward speed once L1 or R1 senses white line, you can modify the following:

```
# ------ You can modify or add your code below this line
    if IR_L1==1 : # L1 senses white line
        CurAction = 1
        Duration = 1
        WheelLeft  = 30
        WheelRight = 30
        LED_1 = 0
    elif IR_R1==1 : # R1 senses white line
        CurAction = 2
        Duration = 1
        WheelLeft  = 30
        WheelRight = 30
        LED_1 = 0
```

```
# ------ You can modify or add your code below this line
    if IR_L1==1 : # L1 senses white line
        CurAction = 1
        Duration = 1
        WheelLeft  = 70
        WheelRight = 70
        LED_1 = 0
    elif IR_R1==1 : # R1 senses white line
        CurAction = 2
        Duration = 1
        WheelLeft  = 70
        WheelRight = 70
        LED_1 = 0
```

Experience Python Code

Step 4: Load and Run Python Code



6

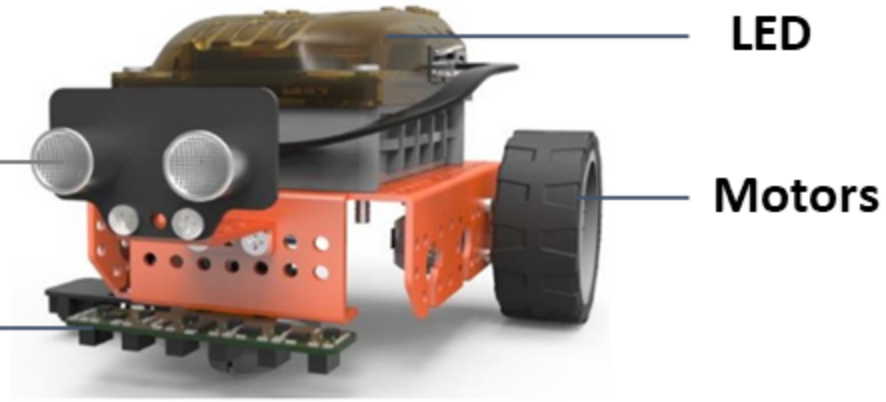
Virtual Environment & Virtual Robot

(Experience Pack)

RoboCup SINGAPORE

CoSpaceRobot

Real Robot vs Virtual Robot



Digital Twins

Real Robot ⟷ Virtual Robot

The real and virtual robots are equipped with the same sensors
They have the same physical models (except for the floor friction and lighting conditions)
The same program works for both the virtual and real robots

RoboCup SINGAPORE CoSpaceRobot

Robot Sensors

Sensors

- 1 ultrasonic sensor is fitted at the front of the robot. The ultrasonic sensor detects objects in front of the robot. If it detects any objects, it will output the distance to the nearest object. Otherwise, it will output the maximum range of the sensor. Unit used is in cm.
- 1 line-following sensor array, comprising of 6 IR sensors, is mounted at the bottom of the robot, towards the front. These sensors are used to detect the colour of the ground beneath the robot. For each sensor, if the ground is light-coloured, it will output WHITE; otherwise, it will output BLACK.

Robot Actuators

LED (Output)

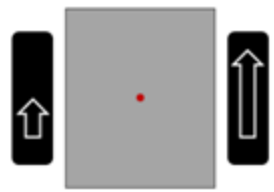
- 1 LED is used to provide visual indication of the robot's status when it is in action.
- This does not have any direct impact on the robot's performance; its main purpose is to facilitate program testing.

Motors (Output)

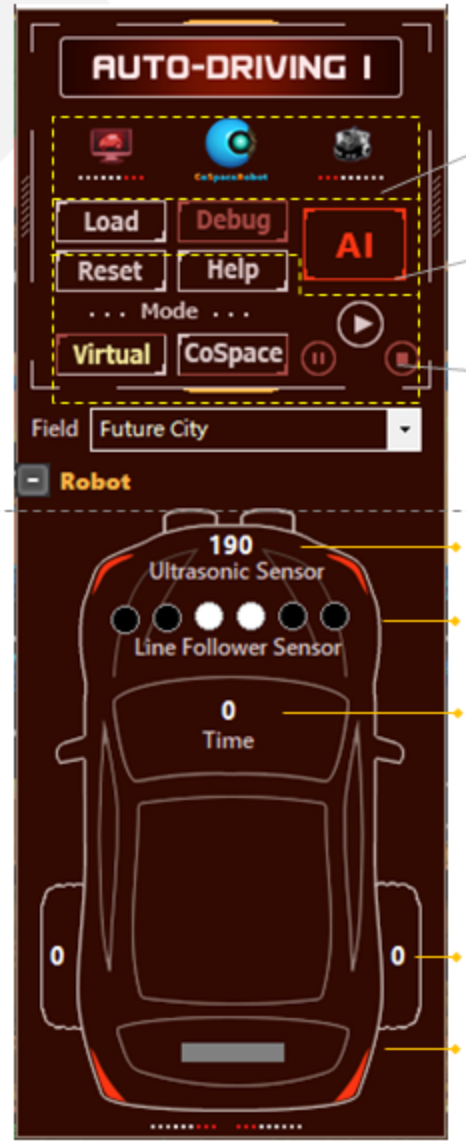
- 2 independently controlled motors provide power to the robot. These are simulated direct-drive, fixed-gear DC motors. Each motor can be assigned its own power setting, ranging from -100 (full reverse) to 100 (full forward). At a power setting of 0, the motor will be idle.


Spin Turn


Pivot Turn


Swing Turn

Control Panel



Robot selection (Virtual/Real)

Load a virtual program

Launch debug panel

Launch AI panel

Reset Robot

Launch Help File

Challenge Modes

Pause/Run/Stop

Ultrasonic sensor's live data (cm)

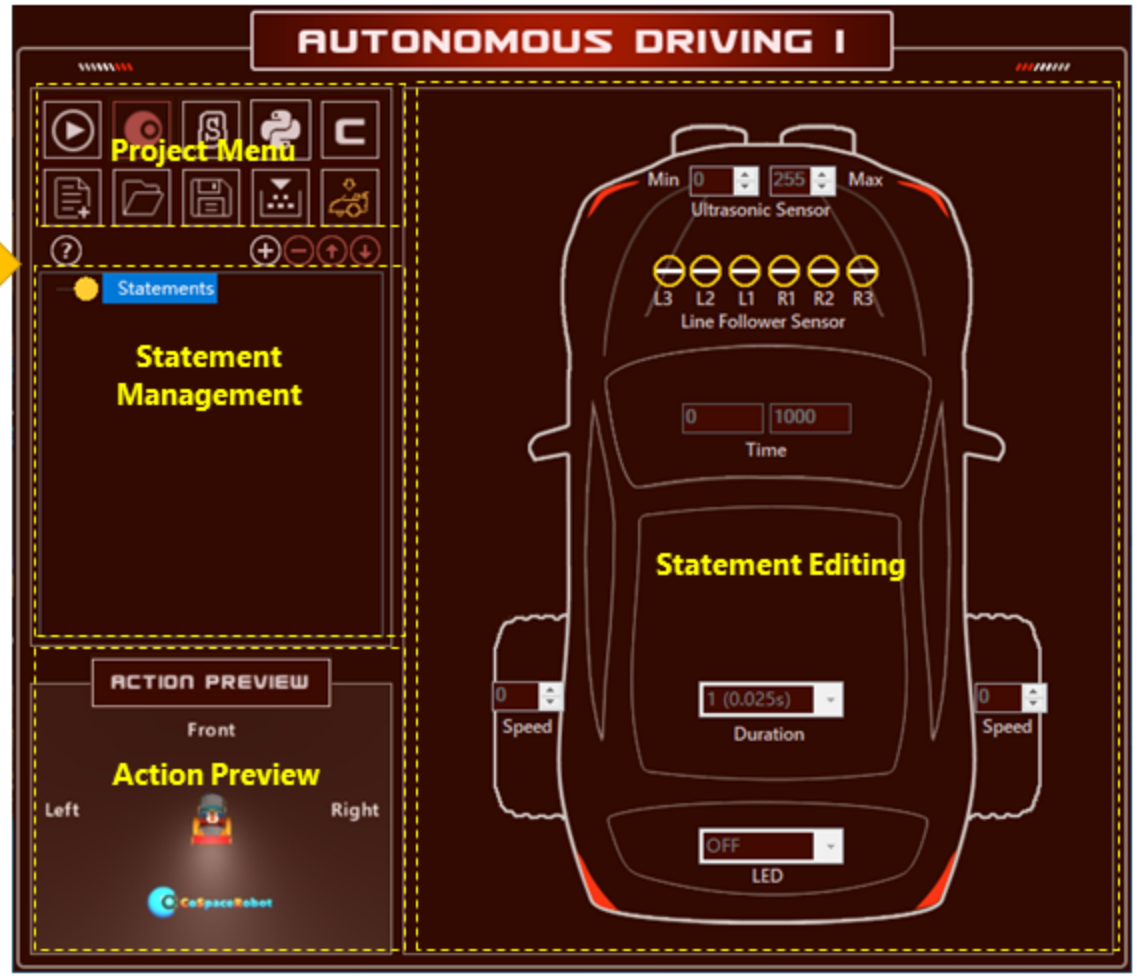
IR sensors' live data (Black or White)

Timer (0 ~ 1000)

Motor speed (-100 ~ +100)

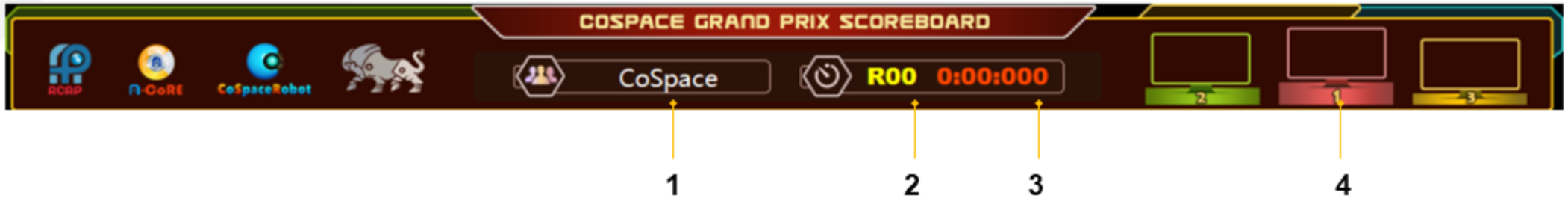
LED

AI Panel



1. Trial RUN
2. CoSpace Program Mode
3. Scratch Program Mode
4. Python Program Mode
5. View C code
6. Create a new project
7. Open an existing project
8. Save the current project
9. Save and build the current project
10. Upload to real robot

Scoreboard



1. ***Team ID:*** Displays the team ID
2. ***Zone:*** Displays the virtual zones travelled.
3. ***Time:*** Records the overall robot movement time since start of the race, including that of the real robot if a real robot started the race
4. ***Rank:*** Only results of the top three teams will be recorded.

7

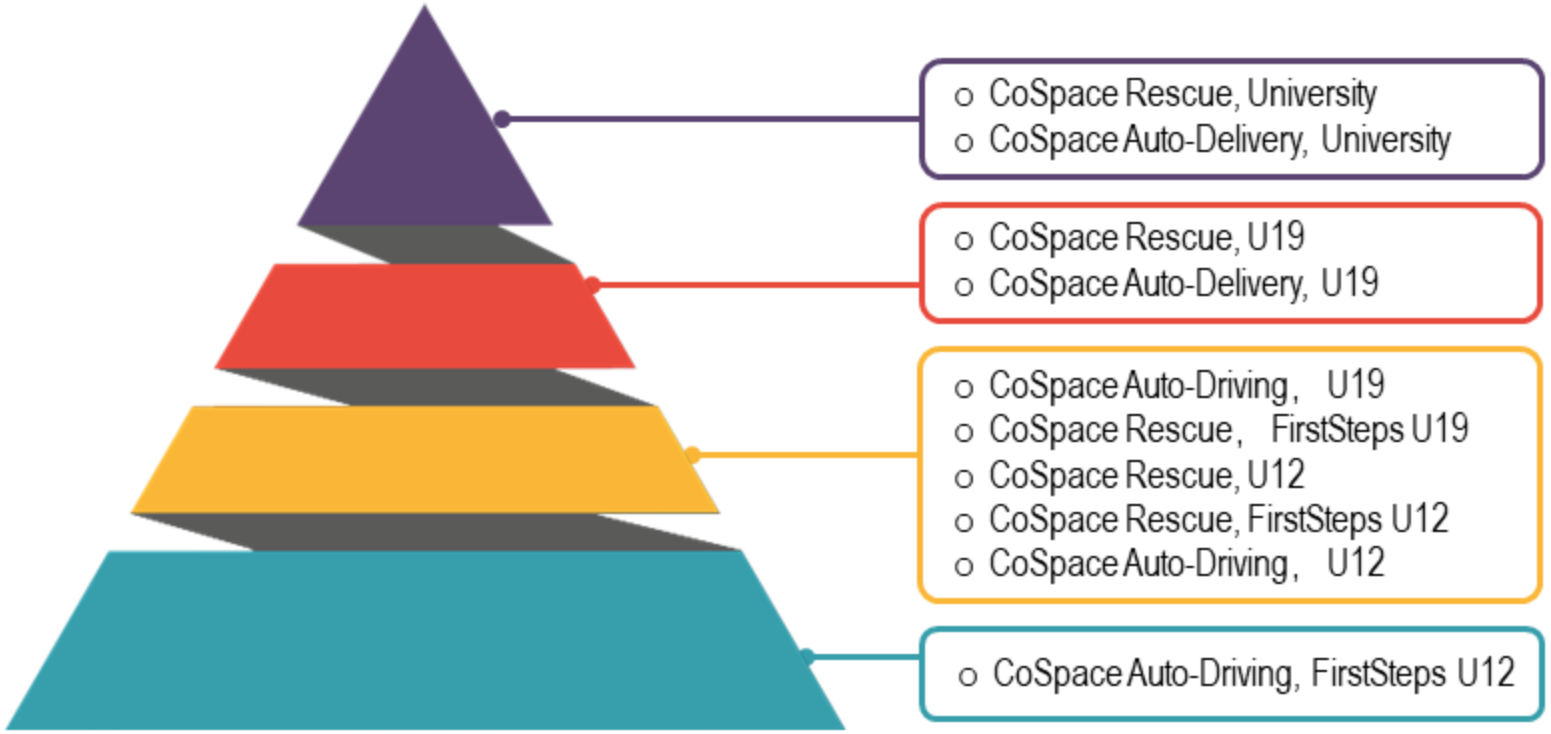
CoSpace Learning Journey

RoboCup SINGAPORE

CoSpaceRobot

CoSpace Learning Journey

CoSpace Autonomous Driving, CoSpace Autonomous Delivery, and CoSpace Rescue are official Challenge Leagues of the Asia-Pacific Robot (RCAP) Competition. CoSpace provides robot enthusiasts with high-quality and efficient robot and AI education and competition solutions.



- ✓ **CoSpace Auto-Driving Challenges**
  - FirstSteps, U12
  - U12
  - U19
- ✓ **CoSpace Rescue Challenges**
  - FirstSteps, U12
  - U12
  - FirstSteps, U19
  - U19
  - University
- ✓ **CoSpace Auto-Delivery Challenges**
  - U19
  - University

RoboCup SINGAPORE

CoSpaceRobot

CoSpace Learning Journey

CoSpace Autonomous Driving, CoSpace Autonomous Delivery, and CoSpace Rescue are official Challenge Leagues of the Asia-Pacific Robot (RCAP) Competition. CoSpace provides robot enthusiasts with high-quality and efficient robot and AI education and competition solutions.

	Beginners	Intermediate	Advanced I	Advanced II
Competitions for Primary Level	○ CoSpace Auto-Driving, FirstSteps U12	○ CoSpace Auto-Driving, U12 ○ CoSpace Rescue, FirstSteps U12 ○ CoSpace Rescue, U12		
Competitions for Secondary Level		○ CoSpace Auto-Driving, U19 ○ CoSpace Rescue, FirstSteps U19	○ CoSpace Rescue, U19 ○ CoSpace Auto-Delivery, U19	
Competitions for University Level				○ CoSpace Rescue, University ○ CoSpace Auto-Delivery, University
Code	GUI, Scratch	GUI, Python, C	GUI, Python, C	GUI, Python, C, Vision

Welcome to the CoSpace Robot Family!

CoSpaceRobot® is an official competition platform for RoboCup Asia-Pacific.

http://robocupap.org/rcap-cospace-challenges/

Email: CoSpace@robocupap.org