PRESENTATION 1 · LEARNING WITH STATEMENTS

First moves

How statements work, how two wheels move the robot, the Time sensor, and Session 1 step by step.

TheBotCup 2026 · AiBotCup Simulator



TODAY

Four steps to your first program

1

Statements

How the robot reads and runs your list.

2

Two wheels

Differential drive: going straight and turning.

3

Time sensor

Use the clock to say WHEN each move happens.

4

Session 1

Solve First moves, step by step.

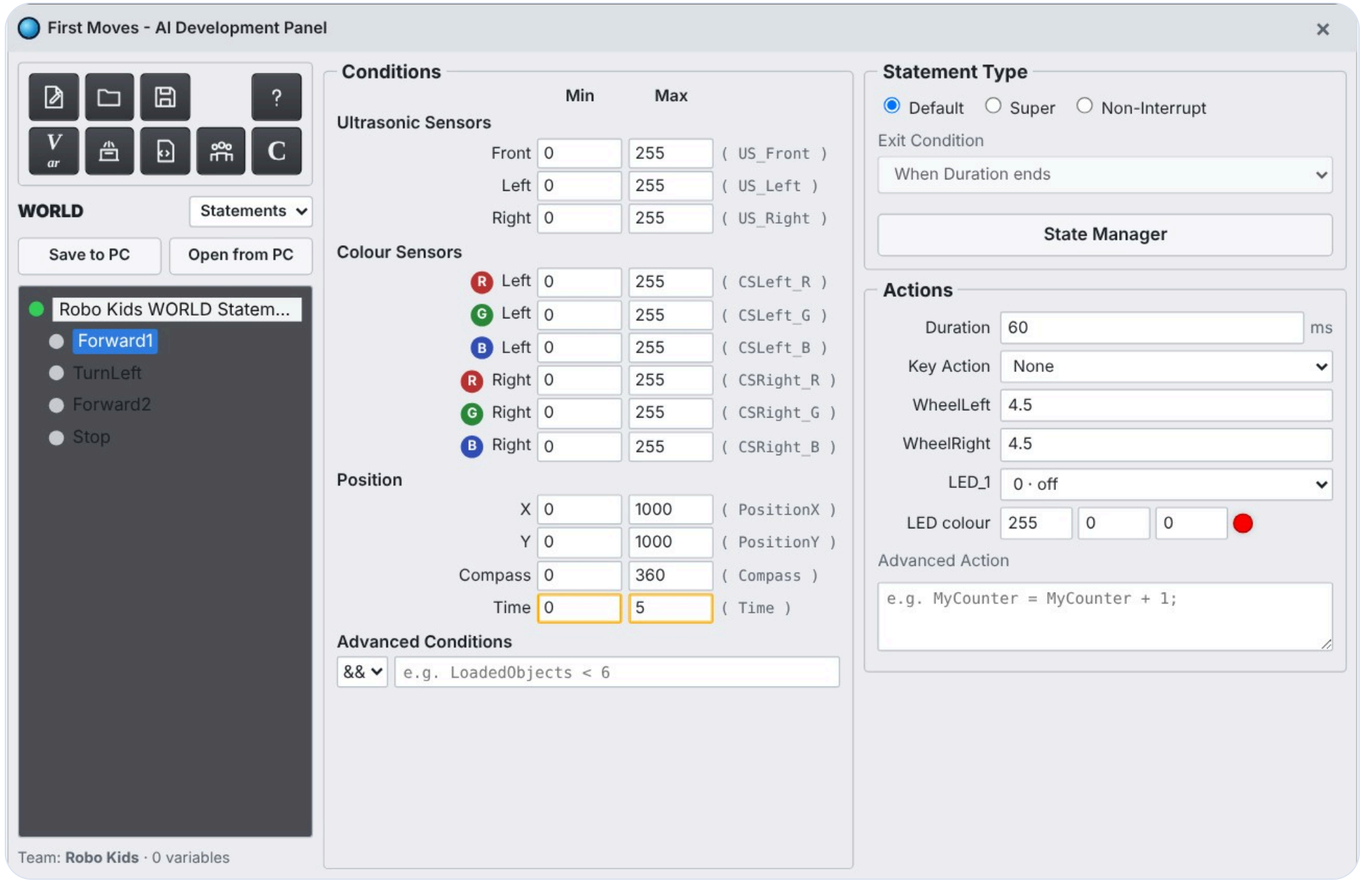
PART 1

How coding with statements works

Conditions · Actions · the 60 ms loop

STATEMENTS

The four parts of a statement



1 · Name in the list on the left, like **Forward1**.

2 · Conditions a Min and Max for each sensor. **All** must be true. Left at full range = always true.

3 · Actions WheelLeft, WheelRight, LED and **Duration** in ms.

4 · Type Default, Super or Non-Interrupt (top right).

STATEMENTS

What the robot does every 60 ms

STEP 1

Still busy?

If the last action’s Duration has not ended, keep going.

STEP 2

Read the list

Check statements from the **top** down.

STEP 3

First match runs

Its actions start, for its Duration.

Order matters. Two statements can both match: only the higher one runs.

Nothing matches? The wheels stop. A last statement with no conditions avoids this.

STATEMENTS

Three statement types, plus Key Actions

Default · we use this today

Runs for its Duration, then the list is checked again. A Super statement can interrupt it.

Super

Jumps in as soon as its conditions are true, and keeps going until its **Exit Condition**: Duration ends, its condition is false, or a custom rule.

Non-Interrupt

Nothing can stop it until its Duration ends. Good for a turn that must finish.

Key Action

Collect (red LED), **Collect in water** (blue), **Deposit** (green): stops, sets the LED, 3.2 s.

STATEMENTS

Duration: how long to keep an action

1000 ms

= 1 second. Duration is always in milliseconds.

60 ms

The smallest step. The robot looks at the list again right away. **We use 60 today.**

400 ms

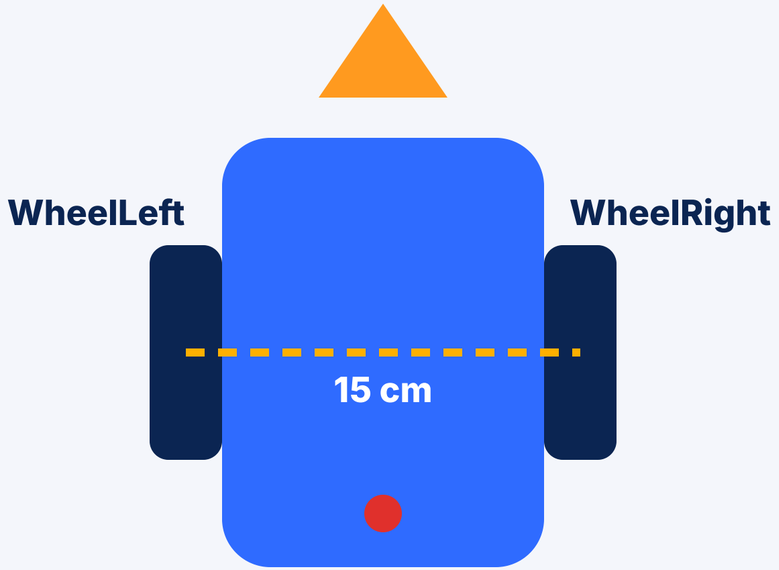
Commit to the action for 0.4 s before looking again.

Short Durations + conditions = the robot keeps checking and reacts quickly.

PART 2

Differential drive: how the robot moves

Two wheels · no steering wheel



DIFFERENTIAL DRIVE

Two motors, one on each side

WheelLeft and **WheelRight** each go from **−5** (full reverse) through **0** (stop) to **5** (full forward), in steps of 0.1.

Each unit = **6 cm per second**. Full speed 5 = 30 cm/s.

Same speeds → **straight**. Different speeds → **it turns** toward the slower wheel.

DIFFERENTIAL DRIVE

Six moves from two numbers



Forward

5 and **5**



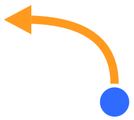
Backward

−5 and **−5**



Stop

0 and **0**



Curve left

2 and **5**



Spin left

−2 and **2**



Spin right

2 and **−2**

Values are **WheelLeft** then **WheelRight**. Opposite values spin the robot on the spot, without moving forward.

DIFFERENTIAL DRIVE

Handy numbers for planning

WheelLeft	WheelRight	What happens
5	5	Straight, **30 cm** each second (fastest)
4.5	4.5	Straight, **27 cm** each second
2.5	2.5	Straight, **15 cm** each second
−2	2	Spin left, about **92°** each second
−1	1	Spin left, about **46°** each second
2	−2	Spin right, about **92°** each second

Speed

(Left + Right) ÷ 2 × 6 cm/s

Distance

speed × time
27 cm/s × 6 s = **162 cm**

PART 3

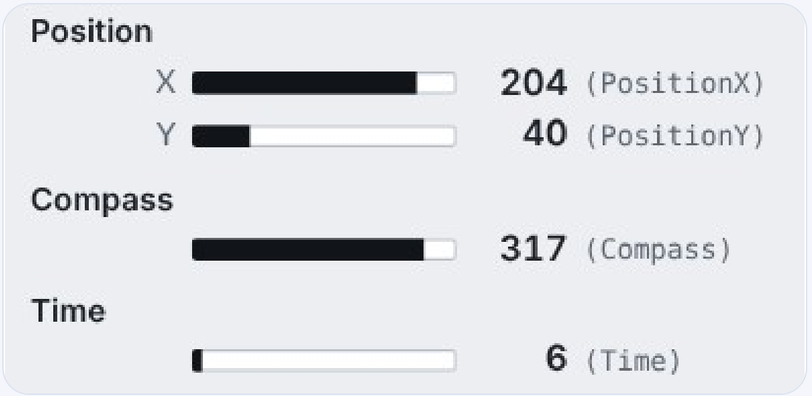
The Time sensor

The robot’s own stopwatch

TIME SENSOR

Seconds since you pressed Play

- Starts at **0** and counts up: 0, 1, 2, 3 …
- Whole seconds only: from 6.0 to 6.9 it reads **6**.
- In a statement: **Time** row, Min and Max.
- Watch it live in the Sensors list.



6.5 s after Play: **Time = 6**

TIME SENSOR

Plan your program as a timeline



Min 0, Max 5

= **6 seconds** (0.0 to 5.9), because Time counts whole seconds.

Min 6, Max 6

= exactly **1 second**: just right for a 92° turn.

No gaps

Each range starts where the last one ended, so one statement always matches.

PART 4

Solving Session 1 · First moves

Look · plan · calculate · write · test

STEP 1 · LOOK

The mission

1

Session 1 · First moves

Drive the robot from START into the green deposit area.

Reach the deposit area: +100. Time limit 3:00.

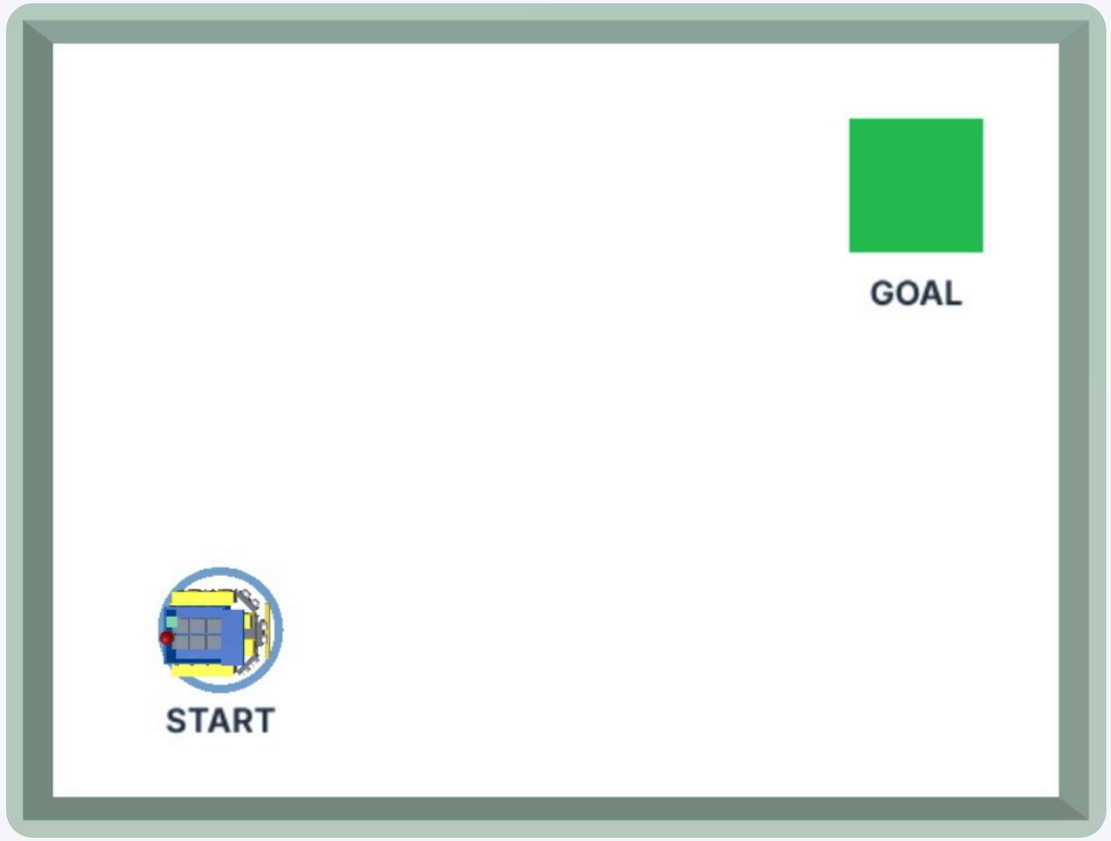
Not done yet

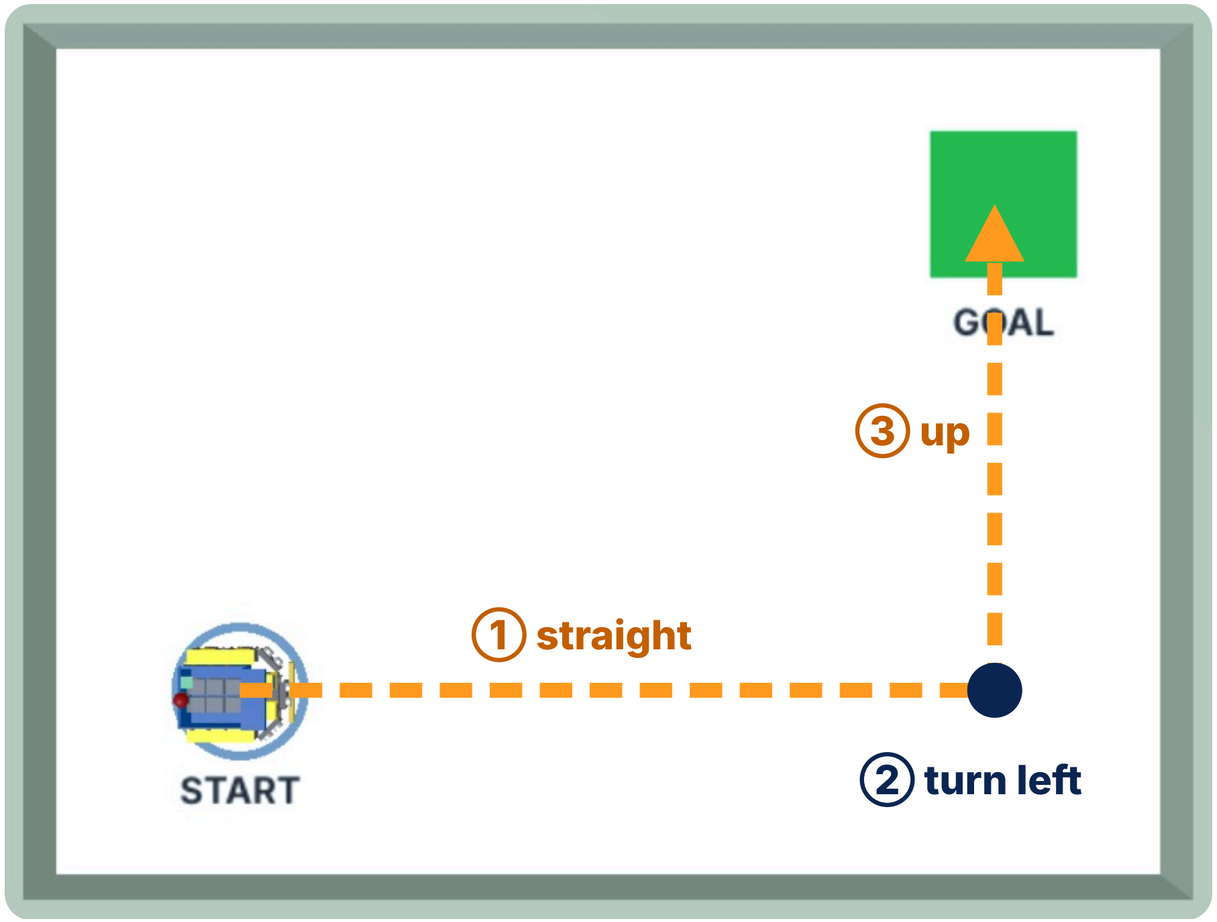
Field 240 × 180 cm with a wall around it.

START bottom left, robot facing **right**.

GOAL green square, 32 × 32 cm, top right.

Win robot centre inside the green: **+100**, within 3:00.





STEP 2 · PLAN

An L-shaped route

① Drive straight right until under the goal.

② Spin left a quarter turn (90°).

③ Drive straight up into the green.

Moves we know: straight and spin. No curves needed.

STEP 3 · CALCULATE

Turn the route into numbers

① **Straight**	Goal centre is ≈ 166 cm to the right. Wheels 4.5 = 27 cm/s. **6 s × 27 = 162 cm** → Time 0–5.
② **Turn**	Wheels −2 and 2 spin ≈ 92° each second. **1 second** → Time 6.
③ **Up**	The green starts 90 cm above. 90 ÷ 27 ≈ **3.3 s**. Give it extra time → Time 7–12.
④ **Stop**	No conditions: the safe “else” at the bottom of the list.

STEP 4 · WRITE

Open the AI panel and add a statement

1 Learning → Sessions → **Session 1**.

2 Press **AI**. First button = **New Project**: name it **First Moves**.

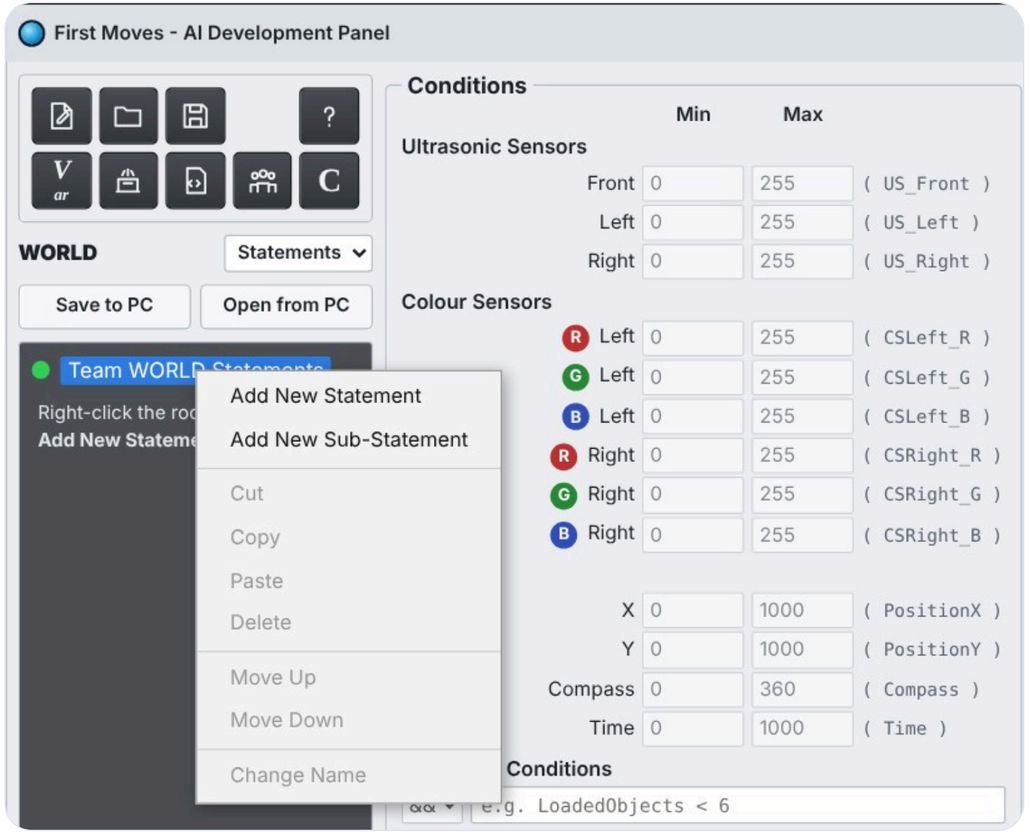
3 People button = **Team Name**, e.g. Robo Kids.

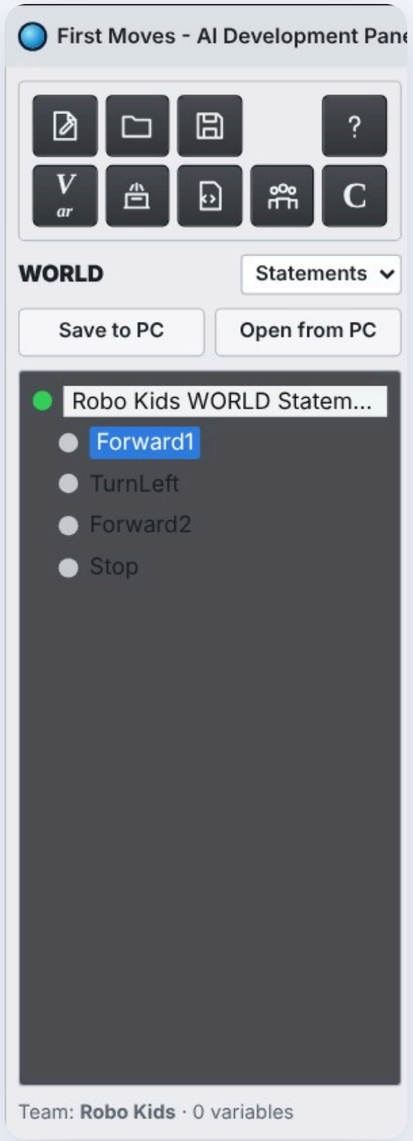
4 WORLD menu shows **Statements**.

5 Right-click the top line → **Add New Statement** → name **Forward1**.



Top: New · Open · Save · Help
Bottom: Var · **Build** · Language · **Team** · Code





STEP 4 · WRITE

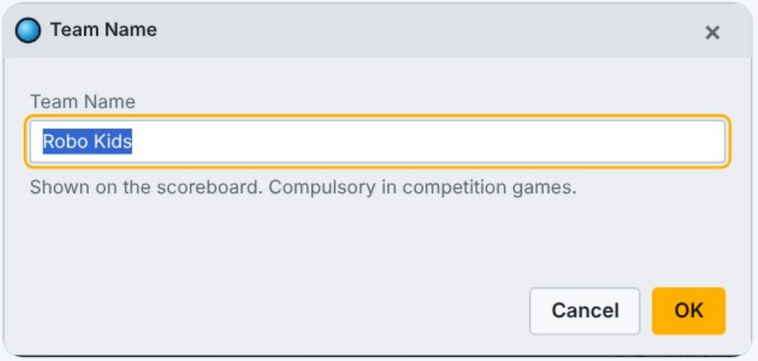
Set or change your Team Name

1 In the AI panel press the **people** button (2nd row, 4th).

2 Type the name: up to **24** characters.

3 Press **OK**. The bottom line shows **Team: Robo Kids**.

4 Press **Build** again so the robot gets the new name.



The team name shows on the **scoreboard**. In competition it is **compulsory**: no name = a warning, the second time you are disqualified from that game.

STEP 4 · WRITE

Four statements, in this order

Statement	Time Min–Max	WheelLeft	WheelRight	Duration
1 Forward1	0 – 5	4.5	4.5	60
2 TurnLeft	6 – 6	−2	2	60
3 Forward2	7 – 12	4.5	4.5	60
4 Stop	(none)	0	0	60

Type **Default** for all four.

Change only the **Time** row. Leave every other sensor as it is.

LED_1 stays **0 · off**.

THE PROGRAM

Our four statements in the AI panel

1st
Forward1

Time 0 5 (Time)

Duration 60 ms
Key Action None
WheelLeft 4.5
WheelRight 4.5

2nd
TurnLeft

Time 6 6 (Time)

Duration 60 ms
Key Action None
WheelLeft -2.0
WheelRight 2.0

3rd
Forward2

Time 7 12 (Time)

Duration 60 ms
Key Action None
WheelLeft 4.5
WheelRight 4.5

4th · always true
Stop

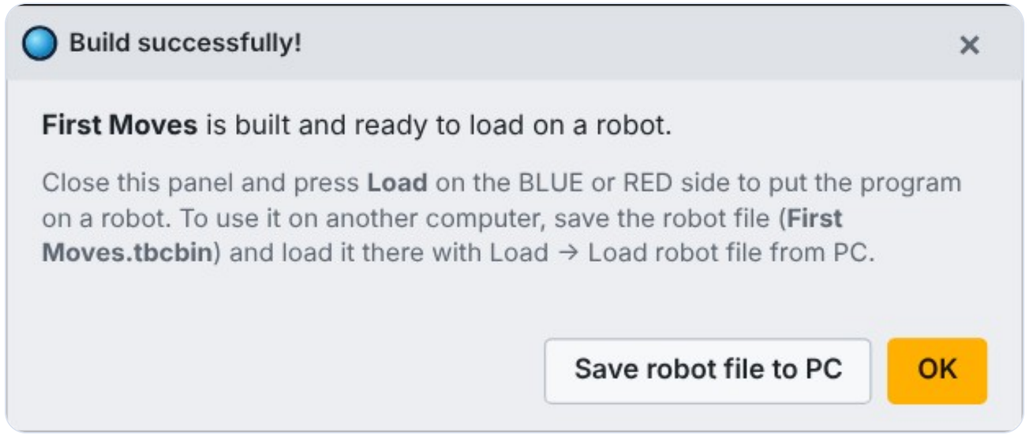
Time 0 1000 (Time)

Duration 60 ms
Key Action None
WheelLeft 0.0
WheelRight 0.0

STEP 5 · TEST

Build, Load on BLUE, Play

1 Press **Build** → 2 OK, close the panel → 3 BLUE **Load** → First Moves → 4 Press ▶ **Play**





At **Time 6** the robot is under the goal and turning left.

KEEP YOUR WORK

Download your code to your PC

TO KEEP EDITING

Project file

AI panel → **Save to PC**

File: **First Moves.tbcbot.json**

Back later: **Open from PC** → choose the file.

READY TO RUN

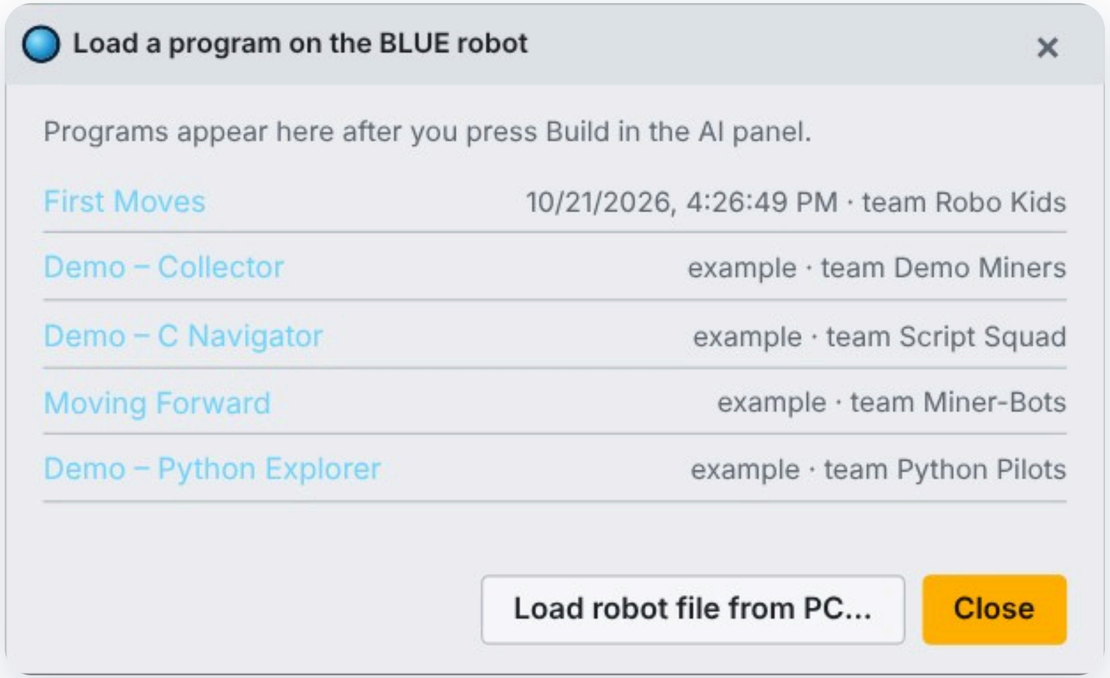
Robot file

Build → **Save robot file to PC**

File: **First Moves.tbcbin**

For another computer or a match. It cannot be edited.

The **disk** button (Save Project) keeps a copy in this browser only. Use **Save to PC** to be safe.



RUN YOUR CODE

Upload your code to the robot

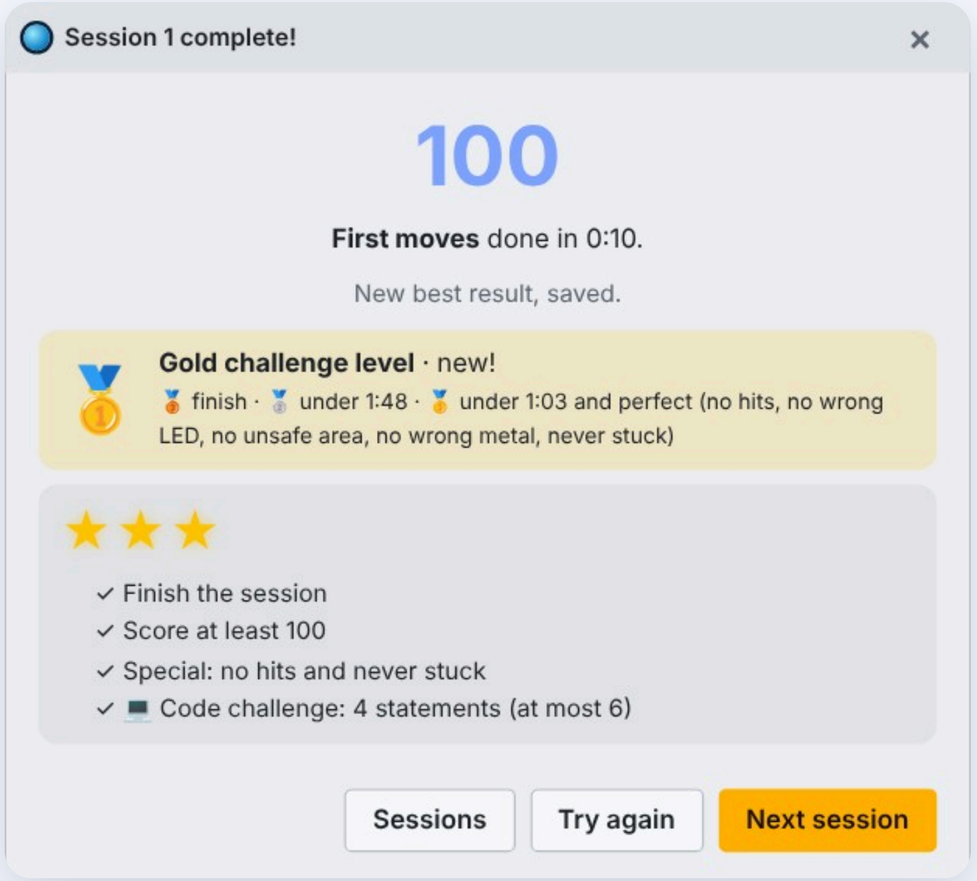
1 Press **Load** on the **BLUE** side.

2a Built here? Click **First Moves** in the list.

2b File on your PC? **Load robot file from PC...** → choose the **.tbcbin** (or .tbcbot.json).

3 Check the name under the BLUE buttons.

4 Press ▶ **Play**.



RESULT

100 points in 0:10 · Gold

Medals · Bronze: finish · Silver: under 1:48 · Gold: under 1:03 and perfect.

Stars · finish · score 100 · no hits and never stuck.

Code challenge · at most 6 statements. We used 4.

DID NOT WORK?

Find the problem, change one thing

What you see	What to change
Robot does not move at all	Build again, then **Load** on BLUE: the name shows under the buttons
Turns too early or too late	**Forward1**: change the Time Max (5) or the wheel speed (4.5)
Turns too much or too little	**TurnLeft**: −1.9 and 1.9 turns ≈ 87° in a second
Stops below the goal	**Forward2**: make the Time Max bigger
Stops after a few seconds	A gap between Time ranges: no statement matches
“Reached, but by hand”	Do not touch the joystick while the program runs

TRY IT YOURSELF

Three challenges

Other way round

Turn left first, drive up, then spin **right** (2, −2) and drive right.

Hard mode 🔥

Tick Hard mode in Sessions and run your program again.

Next session

Presentation 2 · **Three targets**: visit 1, 2, 3 in order.