

Autonomous Car

60 minutes

Digital Creation Lab

WHAT THIS LESSON DOES

In this lesson, you'll explore how autonomous cars function by learning about their sensors and navigation systems. Then, follow step-by-step instructions to create your own autonomous car simulation in Scratch, programming it to navigate a custom track.

WHAT STUDENTS SHOULD BE ABLE TO DO

- Understand the concept and workings of an autonomous car.
- Develop skills in creating a new Scratch project and manipulating sprites.
- Learn to use variables and conditional statements in Scratch to control sprite movements.
- Apply knowledge of sensors in programming an autonomous car to navigate a track.
- Enhance problem-solving skills by implementing speed control and reverse functions in the autonomous car project.

BEFORE THE LESSON

- Open and test <https://scratch.mit.edu> on pupil devices (check the school filter and logins).
- Have the car picture file from the lesson ready to download and upload.
- Plan to show the short step 1 video on a large screen; sound on if the room needs it.
- Devices charged, ideally one per pupil or solid pairs.

Autonomous Car

HOW THE LESSON RUNS

How an autonomous car works

Create a new Scratch project

Add the car sprite

Shrink the car

Draw a test track

Place your car on the track

Create a speed variable

Drive forward

Turn right

Turn left

Speed up and slow down

Reverse

Wrap up

TEACHING MOVES

- 1. How an autonomous car works. Play only enough of the video for pupils to notice cameras and sensors. Pause and ask what the car is detecting and why that matters for driving without a person.
- 2. Create a new Scratch project. Get everyone to a fresh Scratch project and delete the cat together so the stage is clear before the car arrives.
- 3. Add the car sprite. Guide the download-then-upload. Point out the red, blue and orange ellipses on the front: these will act as the car's sensors.
- 4. Shrink the car. Have pupils add the green-flag size block set to 10% and click the flag so they see the car become small enough for the track.
- 5. Draw a test track. Walk the backdrop editor path: green rectangle over the stage, then a black brush track. Keep tracks simple and wide enough for the shrunk car.
- 6. Place your car on the track. Pupils drag the car onto the track first, then add go to x y so values fill from that spot. Add point in direction 90 so it starts facing along the road.
- 7. Create a speed variable. Create variable speed and set it to 5 on green flag. Briefly say a variable is a named value that can change, like a score in a game.
- 8. Drive forward. Inside a forever loop, if the blue sensor colour is touching black, move speed steps. Check colour picks match the blue blob and the track.
- 9. Turn right. Add the red-sensor touching green check that turns clockwise 10 degrees. Run it: the car should correct when the right side meets grass.
- 10. Turn left. Mirror with the orange sensor turning counter-clockwise on green. Test both edges so left and right corrections balance.
- 11. Speed up and slow down. Add up and down arrow handlers that change speed by 1 and -1. Let them try high speed so they see sensors fail if the car moves too fast.
- 12. Reverse. If blue touches green, reverse and turn slightly. Stress the small turn: straight reverse sends the car off the track again.
- 13. Wrap up. Name what they built: sensors deciding move, turn, speed and reverse. Invite a quick spoken share of one change they would try on a new track.

WHAT TO WATCH FOR

- Colour-touching blocks use the wrong pixel colours, so sensors never fire or always fire. Zoom the costume and backdrop; eyedropper the exact blue, red, orange, black and green used in the if blocks.
- Pupils place go to x y before positioning the car, so it always jumps to the wrong start point. Drag the car onto the track first, then drag in go to so x and y fill from the current spot.
- On reverse they move backward with no turn, so the car leaves the track again on the same line. Remind them the step adds a slight turn while reversing so the car re-finds the black road.

DIFFERENTIATION

- Emerging: Pair them and complete upload, shrink and start position together before any sensing code. Have them explain which coloured blob is left, right and centre before building the if blocks.
- Developing: Prompt them to test after each new if block rather than writing all sensors at once. If the car wobbles, suggest slightly wider track lines or a smaller turn angle and retest.
- Proficient: Challenge them to redesign a tighter track and tune speed and turn degrees so the car still stays on. Ask them to try what happens at very high speed and adjust sensor logic or speed limits to cope.