

Your simulation task

Deadline: 13.6.2021

Points: 25

Material requirements: PC or Laptop (no Tablet) with Browser (no Safari)



Topic description

So far you have been solving our tasks in the Hedgehog development environment, now you are to take on our role: think up a programming task and design it in the IDE so that others could solve the task.

We wish you a lot of fun with this and the following tasks!

ATTENTION: For this task you have to program a bit of JavaScript in addition to Blockly. If this is your first time in our simulation environment, try one of the other tasks beforehand and familiarize yourself with the workings of the environment.

Your simulation task

Deadline: 13.6.2021

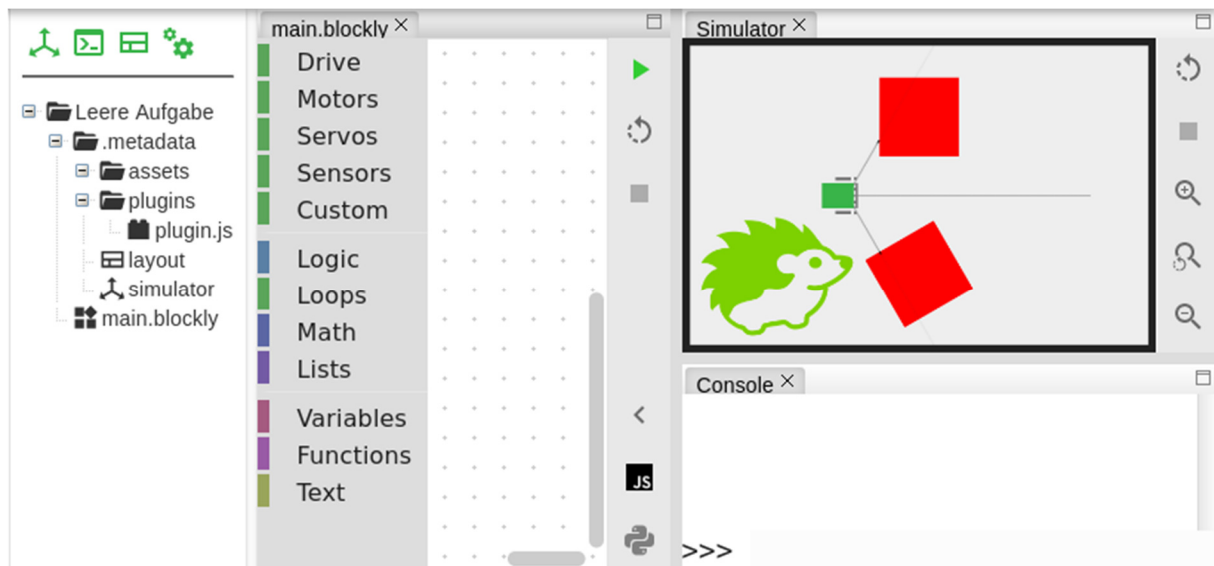
Points: 25

Material requirements: PC or Laptop (no Tablet) with Browser (no Safari)



Task description

Visit our online simulation environment (ide.pria.at) once more and create a new project with the "Leere Aufgabe" task as a base. This template already contains some important files and folders; show them by opening the project settings menu (⚙️) and selecting "Show metadata". When all folders are open, the project should look something like this:



Within the ".metadata" folder there are some files that configure the project, and with which you can create your own task:

- "layout": stores the layout of tabs (Blockly editor, simulation, console) which is used when you first open the task; you don't need to worry about this further.
- "simulator": configures the simulation: where the robot is located, what other objects there are, whether they can be moved, etc.
- "assets/": in this folder you can store images that you want to use as "sprites" for objects in the simulation.
- "plugins/plugin.js": this and all other JavaScript files in the "plugins" folder can add additional functionality to the project. For example, the output displayed in the console when solving a task was generated by plugins.

Your simulation task

Deadline: 13.6.2021

Points: 25

Material requirements: PC or Laptop (no Tablet) with Browser (no Safari)



Your task now is to use these tools (we list further resources below) to create a task that could be solved for example by other ECER participants. Be sure to consider the following things:

- Think about the **target audience** of the task: complete beginners, or users with more experience? Choose the level of difficulty accordingly.
- Write a **description**: what is the task about, what needs to be done?
- Design the **simulation**: add the necessary objects; it also doesn't hurt to provide navigation points - objects that the robot can orientate itself through sensors.
- Think about **grading**: what must be done for the task to be complete? Are there criteria according to which you could also evaluate a "half-finished" solution? You don't need to come up with a point system here, a rough description is enough.
- Create a **plugin** that recognizes the completion/progress of the robot in the task and "comments" on it similar to our tasks. Ideally, you'll also be able to handle the program terminating early and/or the simulation resetting - but at the very least, your plugin should be able to handle a single complete run.
- To test your task you will need a **sample solution** - submit it as part of this task as well.

Misc SDK - this SDK (Software Development Kit) contains general functions and events of the IDE, which are not specifically assigned to a single component of the IDE.

Blockly SDK - with this SDK you can create your own blocks that users can use in their programs.

Simulation SDK - with this SDK you can influence the simulation: Creating or moving objects and reacting to collisions are the most common tasks that can be done with this SDK.

Hedgehog SDK - this SDK is not that important for plugins, but is mentioned for the sake of completeness: with it you can control the robot. The programs you have written so far have used this SDK internally.

We have some resources to help you get started with creating tasks:

- [Hedgehog Cloud: customizing projects with plugins](#) - this demonstration is from June 2020, so some things may look a bit different than today. Overall, though, it's still a fitting introduction and shows most of the features available to you in action.
- [Plugins](#) - part of the Hedgehog IDE documentation on GitHub, gives an overview of what functionality plugins have at their disposal.
- [Misc SDK](#) - this SDK (Software Development Kit) contains general functions and events of the IDE, which are not specifically part of a single component of the IDE.
- [Blockly SDK](#) - with this SDK you can create your own blocks that users can use in their programs.



Your simulation task

Deadline: 13.6.2021

Points: 25

Material requirements: PC or Laptop (no Tablet) with Browser (no Safari)

- [Simulation SDK](#) - with this SDK you can influence the simulation: Creating or moving objects and reacting to collisions are the most common tasks that can be done with this SDK.





Your simulation task

Deadline: 13.6.2021

Points: 25

Material requirements: PC or Laptop (no Tablet) with Browser (no Safari)

- [Hedgehog SDK](#) - this SDK is not that important for plugins, but is mentioned for the sake of completeness: with it you can control the robot. The programs you have written so far have used this SDK internally.
- All the EGER tasks - you can get inspiration from both the descriptions and the tasks in the IDE. Just show the metadata folder and have a look at how we created certain plugins or simulations.
 - you can reuse assets from our tasks
 - our plugins are multilingual; we don't require you to do that, but feel free to follow our example.

Don't let our tasks dictate your structure. Your task can look or work completely different. Let your creativity run free!

Submission

Submit a zip with all project files (unfortunately, at the moment you have to download each file separately). In particular it should contain

- all assets
- all plugins
- the simulator configuration
- (you can leave out the layout configuration)
- your sample solution

Attention: When downloading, please check if all data is complete and has been downloaded successfully. Sometimes there are problems with the download.

If possible, please keep the folder structure of the project. Additionally submit

- Your task description as PDF. The description should name the target audience, include a description, and describe how the assignment would be graded.
- A screen video of your sample solution in action.

Evaluation criteria

- Complete submission including IDE project, PDF and video
- Quality of simulation, plugin and task description
- Jury evaluation: The jury evaluates the quality and creativity of the submission.