Reeborg's World

Reeborg is a programming environment of a little robot using a small language, that will help us getting to know Python. To use it, just go on the webpage `https://reeborg.ca/`.

Reeborg is a robot moving in a two-dimensional world. He is pictured by a humanoid robot (but we can change his picture to a rover). He moves from cell to cell, according to the grid, and leaves a green line where he moves. The red thick lines are walls, through which Reeborg cannot go. He can take objects that lie on the ground or put objects on the ground if he carries some.

To make Reeborg do something, we write a code, using the orders that Reeborg knows, or those that we define ourselves. Reeborg can then move, rotate, interact with its immediate neighborhood (presence or absence of walls, of objects at his feet...). The easiest way to understand how he works is to start the introduction worlds.

TODO: Click on the link "Reeborg's world" on the homepage, then help Reeborg do what is required in the first worlds (click on "World Info" to understand what is asked) by adding commands in the "Python Code" box (commands can be chosen thanks to "Reeborg's keyboard").

Reeborg's common commands are explained in the link "Additional options", "Documentation" then "G. Reeborg's World Reference". There are mainly 3 types of code that you can instruct Reeborg to do, that are summed up here:

- simple commands:
  - `move()` : move in the front cell,
  - `turn_left()` : rotate a quarter turn on the left,
  - `take()` : pick an object on the floor (if there are none, Reeborg screams),
  - `put()` : deposit an object on the floor (if Reeborg has different kinds of objects in his bag, he screams),
  - `build_wall()` : build a wall between his cell and the front cell (if there is already a wall in front of Reeborg, he screams),
  - `done()` : turn off,
  - `think(n)()` : change the time between two successive actions (useful for long programs),
- the tests:
  - `wall_in_front()` : true if there is a wall in front,
  - `wall_on_right()` : true if there is a wall on the right,
  - `object_here()` : true if there is an object on the floor,
  - `carries_object()` : true if Reeborg carries at least an object,
  - `is_facing_north()` : true if Reeborg is looking toward the North,
- the Python structures:
  - a conditional structure: `if` <condition>: <actions> that executes the actions if the condition is true, and else do nothing,
  - a fixed-length loop: `repeat` <number>: <actions> that executes the actions the requested number of times,
  - a conditional loop: `while` <condition>: <actions> that executes the actions as long as the condition stays true,
  - a function definition: `def` <function_name()>: <actions> that defines function_name() as a shortcut for a given sequence of actions.

Don't forget: A tutorial is available by clicking on the link "Additional options", "Documentation" then "A. Basic tutorial". You can of course also call me anytime.