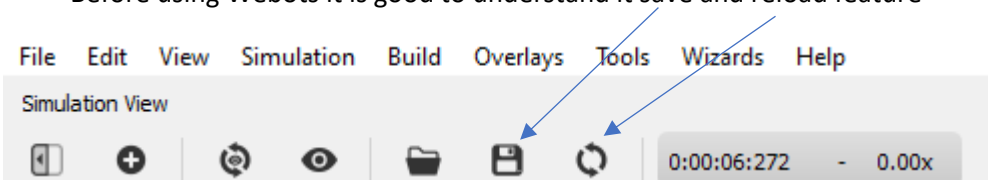
Webots setup Guide

Contents

Before using Webots it is good to understand it save and reload feature



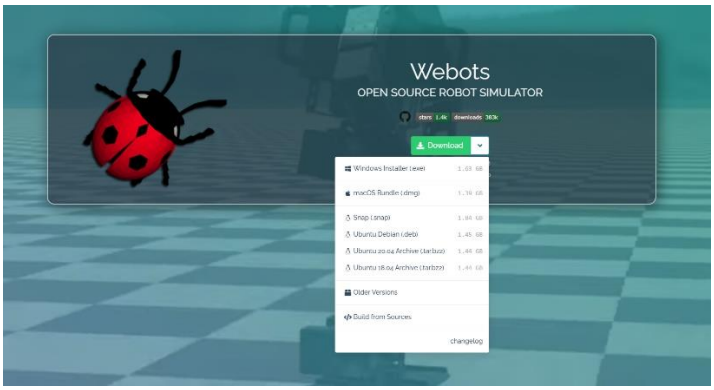
Save will save the entire simulation and any setting or changes you have made meaning you can resume at the point you left off at. Reload will reset the world to the last save, this will undo any changes you have made to both the robot and the environment, including adding object or changing sensor or robot parameters. Subsequently it's a good idea to keep backups (by coping the .wbt file into another directory) in case you accidently make unwanted changes and save them. I would also advice you to make a new save file (not an overwrite) whenever you make significant changes to the world. Ctrl-z can be used to undo a mistake

Installation and Setup:

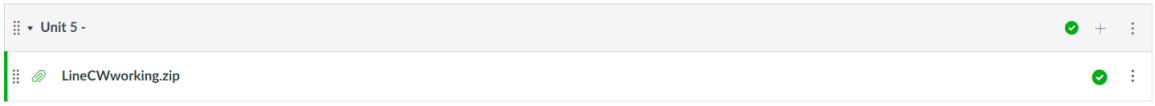
Please make sure you have a suitable version of python installed, 3.7, 3.8 and 3.9 are recommended. To check open a terminal and type python. If you do not have python installed refer to the previous worksheets.

```
C:\Users\john>python
Python 3.9.4 (tags/v3.9.4:1f2e308, Apr  6 2021, 13:40:21) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> _
```

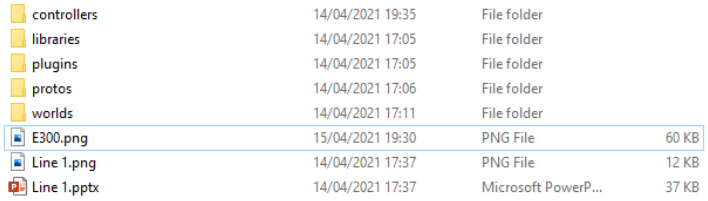
Download Webots from https://cyberbotics.com/ and follow the install instructions. The Windows.exe and the Snap (Linux) are likely the easiest.



Download the zip under unit 5 from our canvas module site.



Extract the zip. When you open the folder, you should see the following



The three key folders to know about are

1) **controllers** contains the python files for the robot.

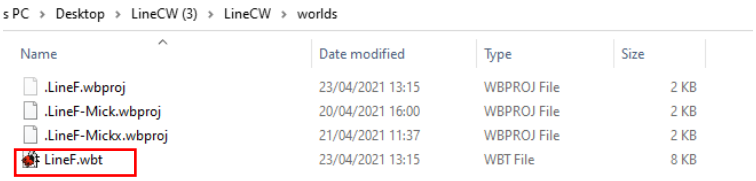
2) **protos** contains the .wbo files defining the robots shape and sensors.

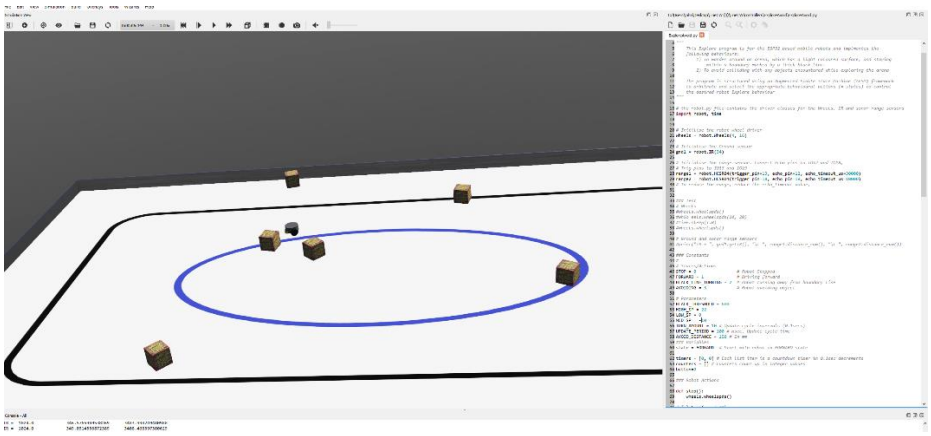
3) **worlds** contains the .wbt files defining the environment

You can think of the simulation structure as being an Environment (worlds) containing a robot (proto) controlled by python (controller)

Running Webots

The easiest way to run is to open the worlds folder and double click the LineF.wbt.





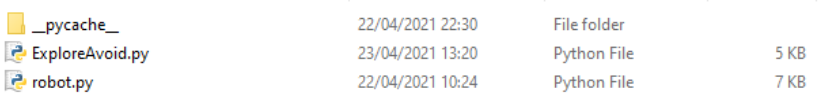
At this point in time the robot should be moving around the environment. Actively avoiding objects and staying within the boundary defined by the black line.

Possible issue

Runtime errors or robot not moving

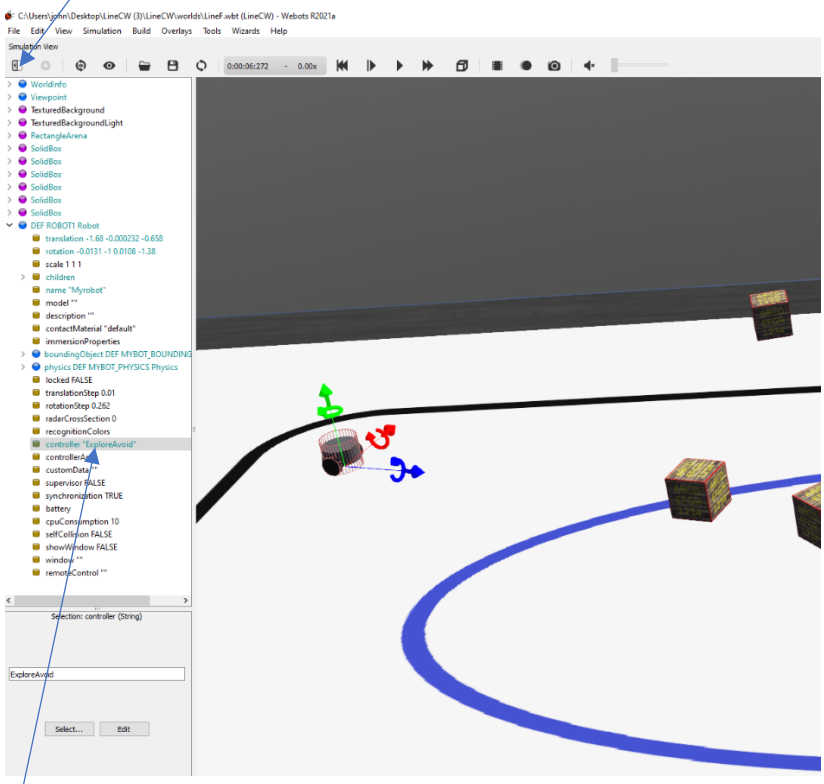
If the robot is not moving check the play button is pushed and the simulation is actively running

If a file not found warning is displayed in the console. Firstly, check you have both ExploreAvoid.py and robot.py in the controller folder \controllers\ExploreAvoid



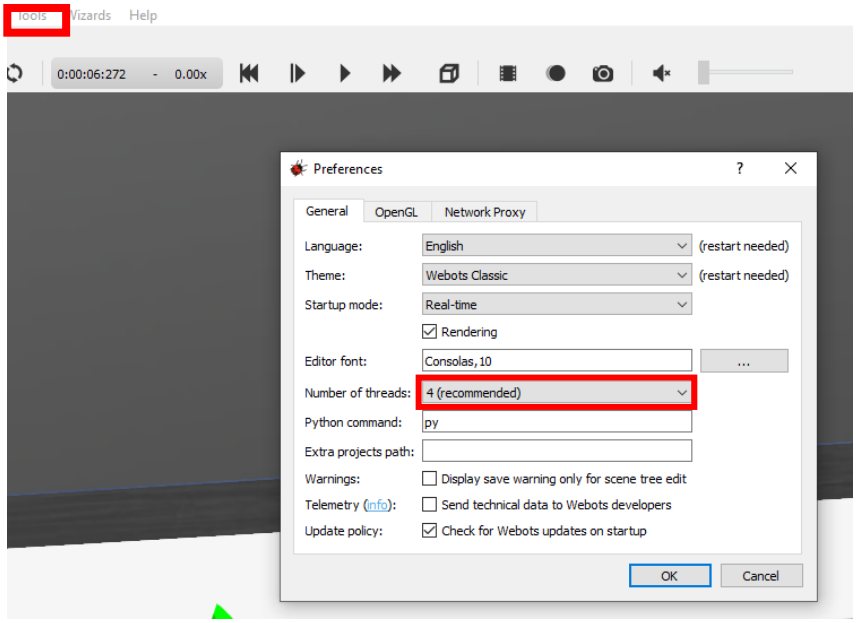
Next check the correct controller is selected for the robot

Open the scene menu



Check ExploreAvoid is selected (default controller)

If ExploreAvoid is selected and you still get a console error open tools, preferences and make sure python command is set to py

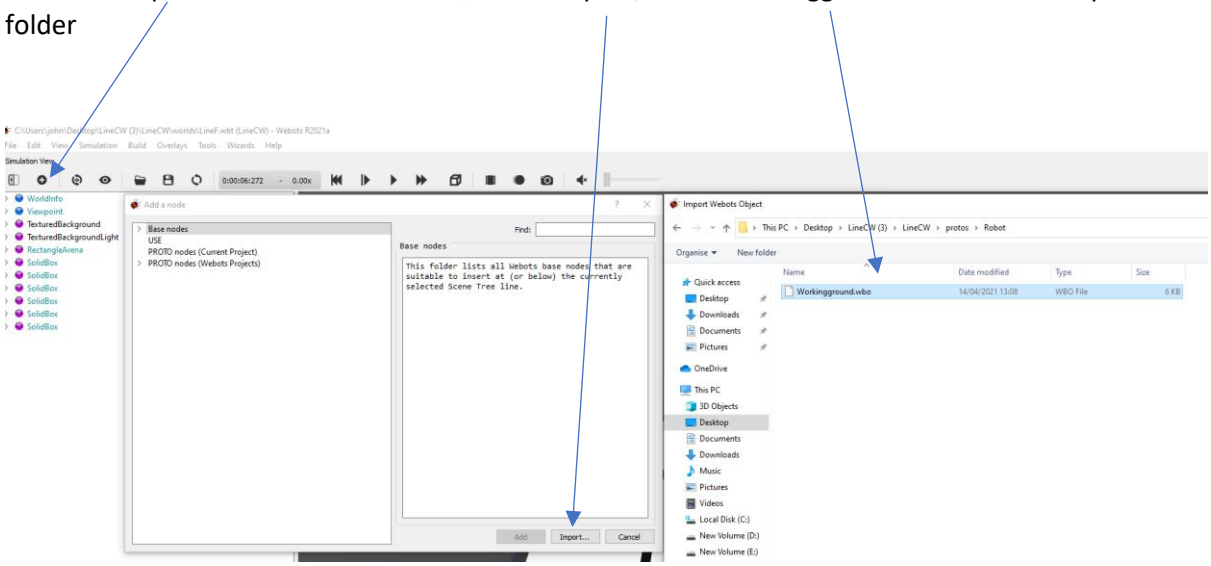


If the robot is still not running and you have followed the above steps, please send us an email.

No Robot

If the robot is missing from the simulation, you can add a new one in the following way

Click on the plus icon in the scene tree, select import, select workinggroundrobot from the proto folder

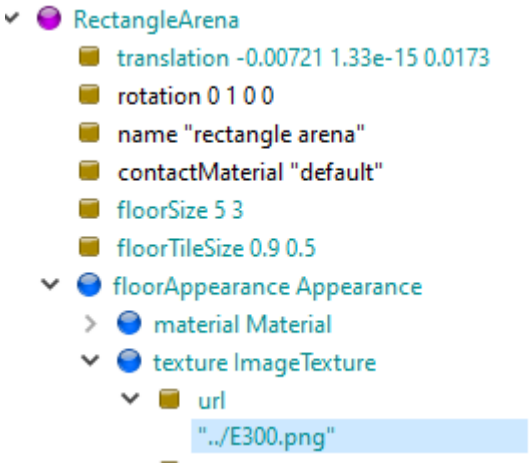


You can add multiple robots if you wish to.

Under the proto nodes in the above screenshot you will also find a large number of prebuilt objects and robots which you may wish to add to your world.

No ground textures

If ground texture is missing expanded the RectangeArena as per the screenshot bellow and ensure the URL links to the E300.png.



Key Simulator elements

A short video will be uploaded under unit 5 going over the key features

Coding a robot in webots

A short Video will be uploaded under unit 5 showing how to code the simulated robot