

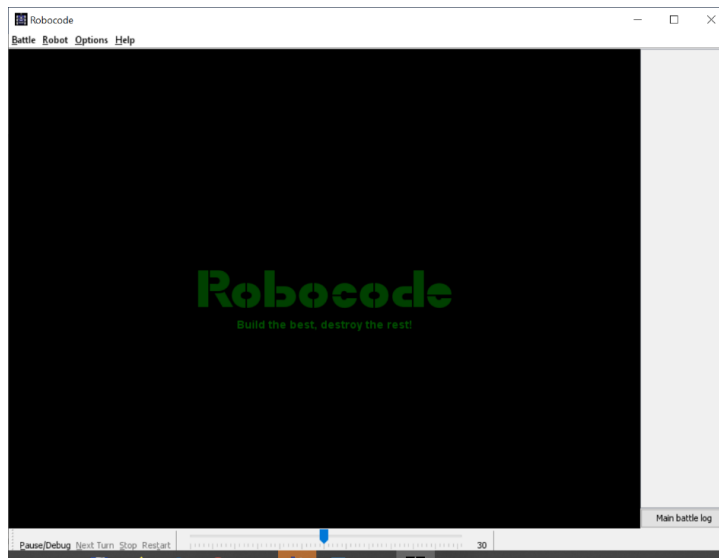
# Getting started with Robocode

## 1 Before doing this tutorial

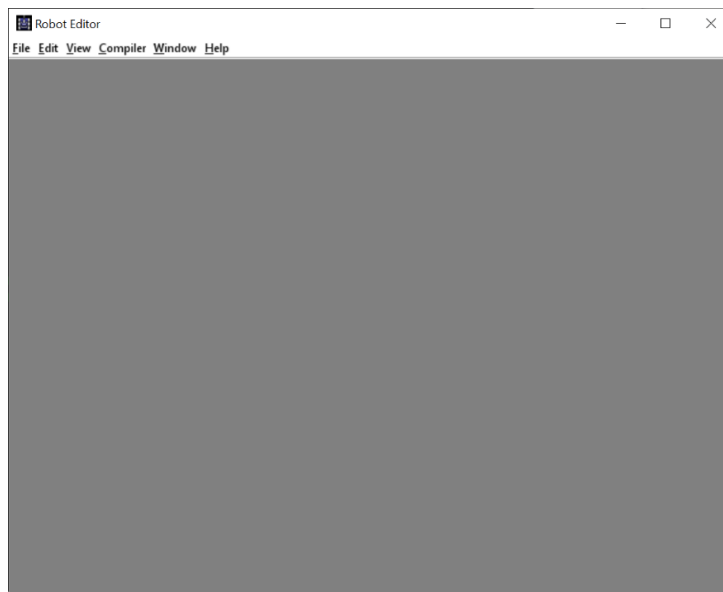
- Make sure you have installed Java and Robocode as described in the day one document **Installing Robocode on Windows.docx**

## 2 Start Robocode and enter the code editor

- Click Start on your computer and type **Robocode**, then click on the Robocode app to start

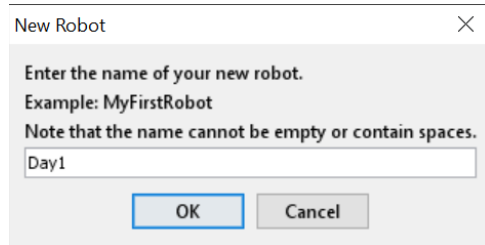


- Now click on **Robot > Source Editor** to start the Java Editor

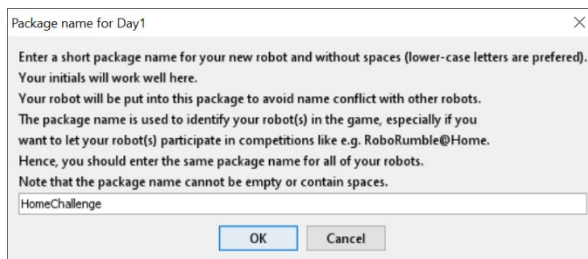


### 3 Create your first robot tank

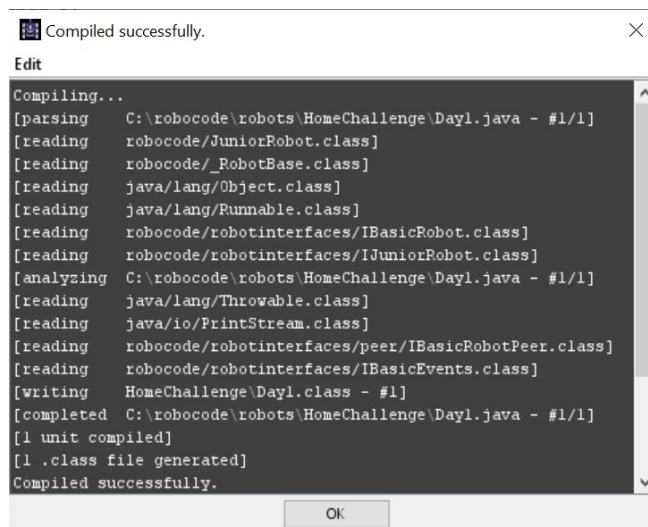
- Click **File > New > JuniorRobot**
- Let's call the first robot Day1



- And let's call the Package **HomeChallenge**, this is the collection where all our robots will be saved



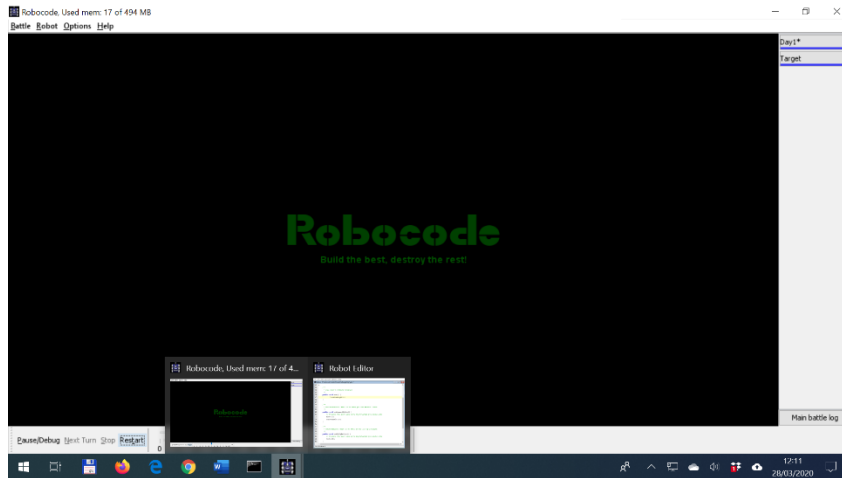
- Save the Robot by clicking **File > Save** and say **yes** to create the directory and click **Save**
- Let's immediately build the robot by compiling it. This checks first if we typed any instructions that the computer could not understand and second turns the Java code into something that the computer can run. Click **Compiler > Compile** and make sure the code compiles successfully.



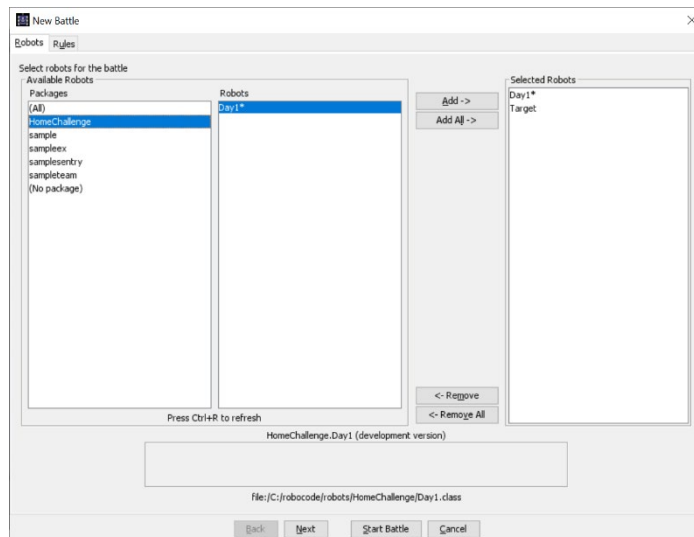
If you have any problems here, then you must have accidentally typed something in the editor. Try and fix it or make a new robot from scratch and try again.

## 4 Your first battle

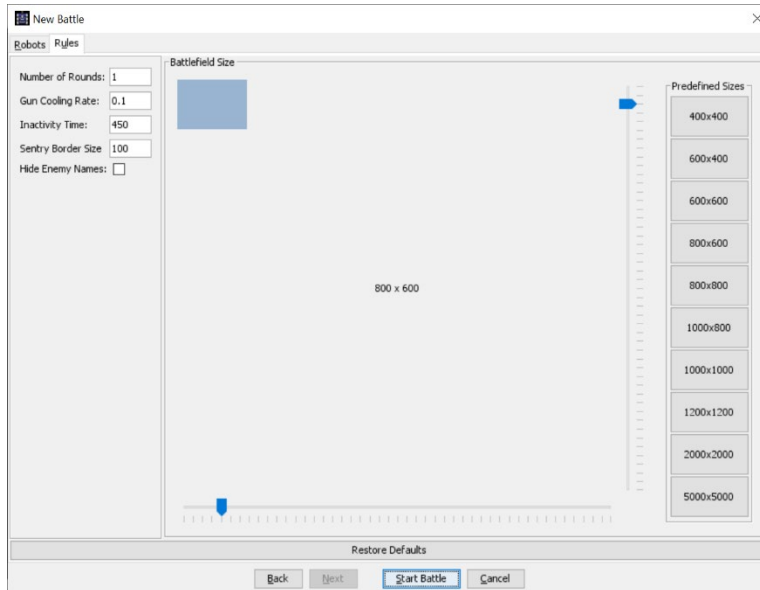
- Go back to the main Robocode window (you can also keep the Robot Editor open because we will make a couple of changes)



- Click on **Battle > New**
- Select **HomeChallenge** from the leftmost column (**Packages**), then select **Day1** from the middle **Robots** column and then click the **Add** button to add this robot to the **Selected Robots** list on the right of the window. From the **Sample** Package select the **Target** and add it to the **Selected Robots** list. Click Next.

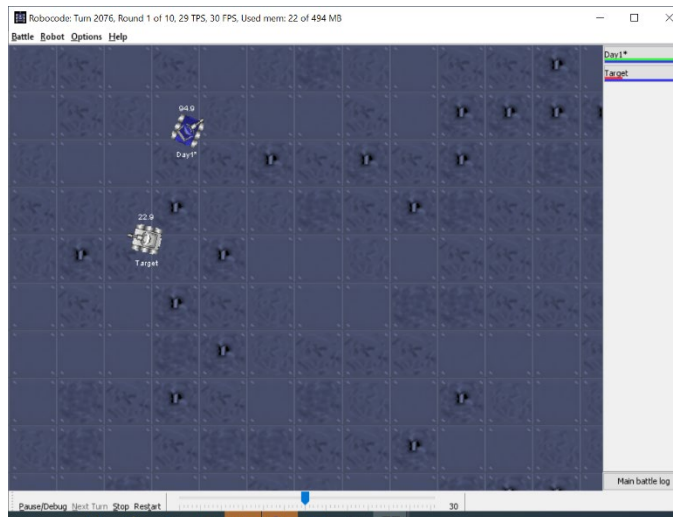


- On the next screen change the Number of Rounds and click **Start Battle**



You will now see a strange and very boring battle taking place in front of you:

- Your tank will move forwards and backwards, swinging its gun around at the end of each of its moves
- When your gun is pointing at the enemy it will fire, but the gun is moving so fast that it will probably swing past the enemy before the shot is fired and miss completely.
- Every time you shoot, you lose 1 energy. If you manage to hit the enemy, you get some energy back.
- After a short while of not moving, the enemy starts losing energy. This is because the game starts to zap your tank if you spend too long not moving.



At the end of the game you get a summary of who won, and where you scored points.

Rank	Robot Name	Total Score	Survival	Surv Bonus	Bullet Dmg	Bullet Bonus	Ram Dmg * 2	Ram Bonus	1sts	2nds	3rds
1st	HomeChallenge.Da...	176 (100%)	50	10	97	19	0	0	1	0	0
2nd	sample.Target	0 (0%)	0	0	0	0	0	0	0	1	0

Our tank is not doing very well, even against a target that doesn't move much, and doesn't shoot back. Let's change that!

## 5 First improvement

Let's change our `run ()` method. A 'method' is a piece of code that the computer can execute and will make your robot do something. There are different kinds of methods that your robot can execute at different times, but right now we will focus on the `run ()` method. The `run ()` method is executed at the beginning of every turn.

Robocode uses turns, measured in ticks, to slow each robot to do something. A tick is a very very short period of time, so it will look to you like the robots are always doing something.

The `run ()` method is between lines 15 and 29 in the Robot Editor. The current `run ()` method does a couple of things:

- It sets the body, gun, radar, bullet and scan arc color to orange, blue, white, yellow and black.
- It then goes into an infinite loop where it moves forwards for 100 pixels (a pixel is 1 dot on a computer screen), turns the gun right for 360 degrees (360 degrees is a full circle), goes backwards for 100 pixels and turns the gun right for 360 degrees. It will do this for ever and ever and ever (or until the game ends).

So, what do we want to change?

- We don't need to set the colors of the tank. So, let's just remove that for now.
- We don't need the loop, because the `run ()` method will be called at the beginning of every round anyway.
- Moving backwards and forwards is pointless right now. No one is shooting at us, so let's just sit still.
- Our gun is moving too fast. So, by the time we have detected the enemy and shoot a bullet, the gun has swept past the enemy and we miss.

So, lets change our run method to look like this:

```
public void run() {  
    turnGunRight(1);  
}
```

That's right. We move the gun by 1 degree to the right. This will move the gun slowly, and when we detect an enemy it will give us plenty of time to take one or more shots!

Remove all the code inside the `onHitWall ()` and `onHitByBullet ()` methods too. We are not moving, and no one is shooting at us, so we don't need that code:

```
public void onHitByBullet() {  
    // Replace the next line with any behavior you would like  
}  
  
public void onHitWall() {  
    // Replace the next line with any behavior you would like  
}
```

So, let's see how well this new robot performs:

- Save your robot by clicking **File > Save** (or press **Ctrl+S** for short).
- Compile your robot by clicking **Compiler > Compile**
- Go to your Robocode main window and start a new battle with your **Day1** robot and a **Target** robot. It should all be setup already, but if it isn't, you can follow the instructions from above.

Hooray! Our robot did a lot better this time. But there is still enough time today to make one small improvement...

## 6 One more small improvement

The main problem now is that we are only shooting at the enemy when our gun is pointing at them, and our gun keeps on swinging around and around. Wouldn't it be great if we keep our gun pointed at the enemy once we have found them? Easy!

At the beginning of every turn when the `run ()` method is executed, the gun is moved right by 1 degree. So, once we have the enemy in our scanners, we should tell the gun to move left by 1 degree, therefore cancelling the effect of the `run ()` method. Then when the enemy moves and leaves our scanners, the gun will continue to swing around in a clockwise direction.

Go back to the Robot Editor.

For this modification we need to look at the `onScannedRobot ()` method. This method should be directly below the `run ()` method on line 21. The `onScannedRobot ()` method is executed (or called) when your robot scans (or sees) another robot. At the moment the `onScannedRobot ()` method has only one comment and one command in it.

A comment begins with a `//` and you can put anything you want after the `//`. The computer won't try to read the rest of the line. This is a place where you can write some notes to remind yourself what your code is doing. You can even put it in front of a command and the computer will ignore the command.

The command in the `onScannedRobot ()` method is `fire (1) ;`. This fires the gun with a strength of 1. You can fire the gun with any strength from 0.1 to 3 in increments of 0.1. The harder you shoot the more damage you will do, but the more energy you will drain.

Now let's add another command to the `onScannedRobot ()` method:

```
public void onScannedRobot() {  
    // Replace the next line with any behavior you would like  
    fire(1) ;  
    turnGunLeft(1) ;  
}
```

The new method will turn the gun to the left by 1 degree, therefore cancelling out the gun being turned right in the `run ()` method. Also note that the command ends with a `;` (semicolon). In java every command ends with a `;`. This is to tell the computer that we have reached the end of a command. It's the same as putting a full stop at the end of a sentence to tell you that we have reached the end of this sentence.

Now **Save** and **Compile** your code again and run a new battle with a **Target** and **Day1** robot.

Success! You now have a pretty efficient robot for battling an enemy that doesn't shoot back or move around very much.

Next time we will make things a bit harder.

## 7 Until next time

Here are some things for you to read before our next session:

[http://robowiki.net/wiki/Robocode/My\\_First\\_Robot](http://robowiki.net/wiki/Robocode/My_First_Robot)

The link above has some more information about what we were doing, and it may give you some more ideas to play around with until next time.