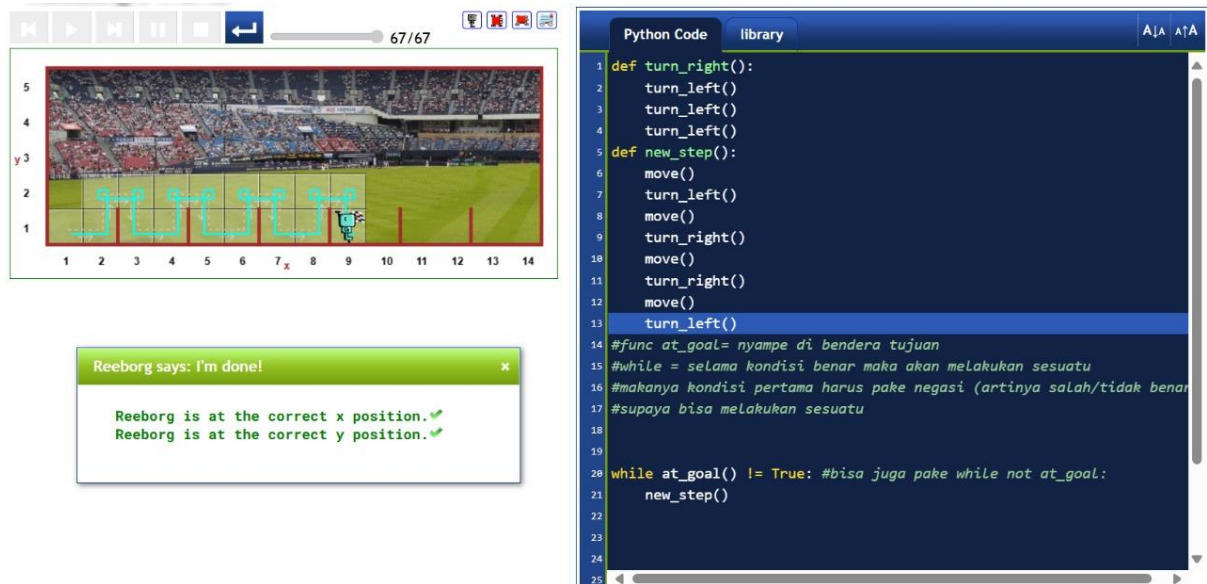


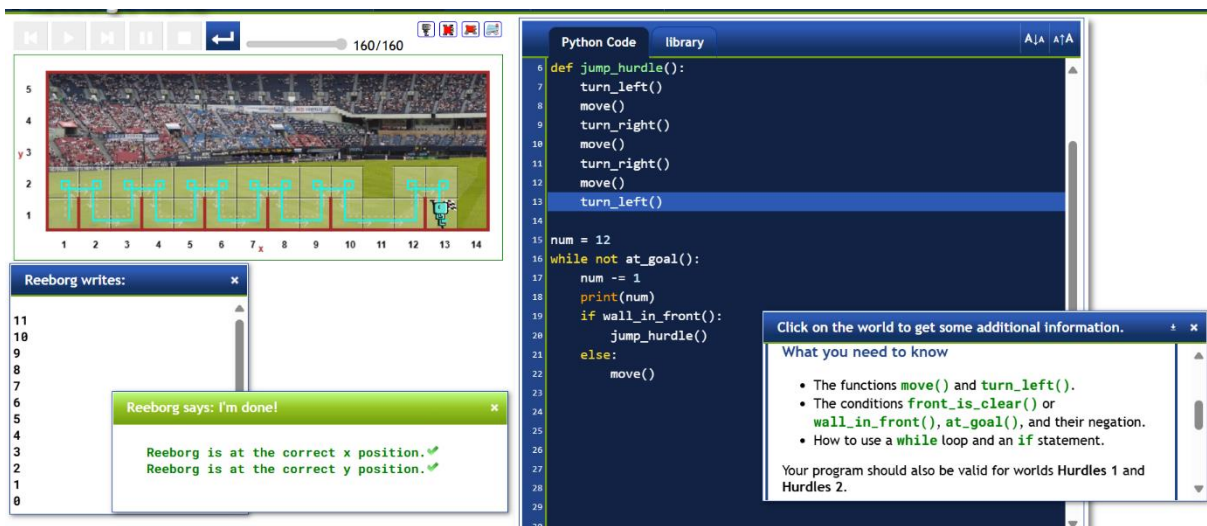
Hurdle 2



The screenshot shows the Hurdle 2 game world on the left and a code editor on the right. The game world displays a field with 14 hurdles and a goal at the end. A progress bar at the top indicates 67/67. Below the game world, a green box says "Reeborg says: I'm done!". Below that, a white box with green text says "Reeborg is at the correct x position. ✓" and "Reeborg is at the correct y position. ✓". The code editor on the right shows the following Python code:

```
Python Code library A|A A|A
1 def turn_right():
2     turn_left()
3     turn_left()
4     turn_left()
5     turn_left()
6 def new_step():
7     move()
8     turn_left()
9     move()
10    turn_right()
11    move()
12    turn_right()
13    move()
14    turn_left()
15 #func at_goal= nyampe di bendera tujuan
16 #while = selama kondisi benar maka akan melakukan sesuatu
17 #makanya kondisi pertama harus pake negasi (artinya salah/tidak benar)
18 #supaya bisa melakukan sesuatu
19
20 while at_goal() != True: #bisa juga pake while not at_goal:
21     new_step()
22
23
24
25
```

Hurdle 3



The screenshot shows the Hurdle 3 game world on the left and a code editor on the right. The game world displays a field with 14 hurdles and a goal at the end. A progress bar at the top indicates 160/160. Below the game world, a green box says "Reeborg says: I'm done!". Below that, a white box with green text says "Reeborg is at the correct x position. ✓" and "Reeborg is at the correct y position. ✓". The code editor on the right shows the following Python code:

```
Python Code library A|A A|A
6 def jump_hurdle():
7     turn_left()
8     move()
9     turn_right()
10    move()
11    turn_right()
12    move()
13    turn_left()
14
15 num = 12
16 while not at_goal():
17     num -= 1
18     print(num)
19     if wall_in_front():
20         jump_hurdle()
21     else:
22         move()
23
24
25
26
27
28
29
30
```

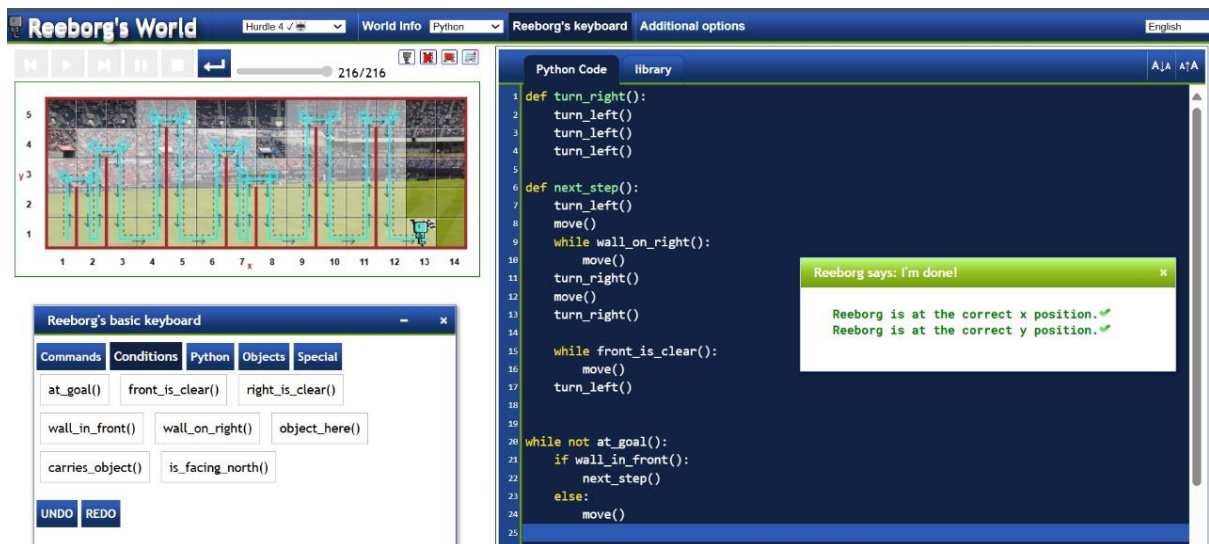
Click on the world to get some additional information.

What you need to know

- The functions `move()` and `turn_left()`.
- The conditions `front_is_clear()` or `wall_in_front()`, `at_goal()`, and their negation.
- How to use a `while` loop and an `if` statement.

Your program should also be valid for worlds Hurdles 1 and Hurdles 2.

Hurdle 4



Reeborg's World

Hurdle 4 ✓ World Info Python Reeborg's keyboard Additional options English

216/216

Reeborg's basic keyboard

Commands	Conditions	Python	Objects	Special
at_goal()	front_is_clear()	right_is_clear()		
wall_in_front()	wall_on_right()	object_here()		
carries_object()	is_facing_north()			

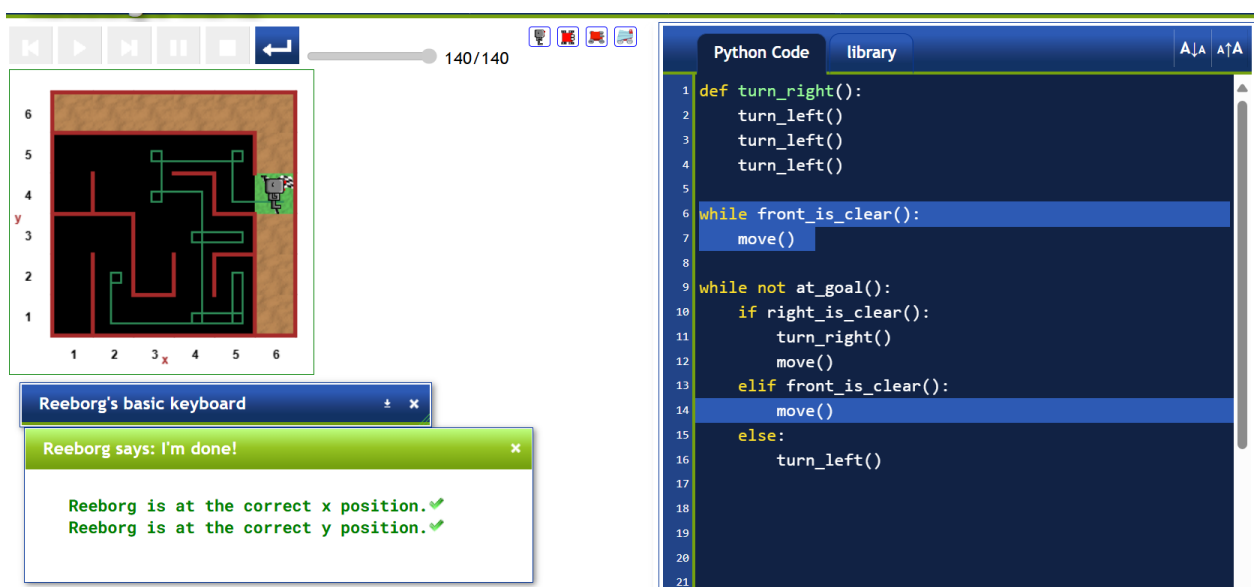
UNDO REDO

```
1 def turn_right():
2     turn_left()
3     turn_left()
4     turn_left()
5
6 def next_step():
7     turn_left()
8     move()
9     while wall_on_right():
10        move()
11    turn_right()
12    move()
13    turn_right()
14
15    while front_is_clear():
16        move()
17        turn_left()
18
19
20 while not at_goal():
21     if wall_in_front():
22         next_step()
23     else:
24         move()
25
```

Reeborg says: I'm done!

Reeborg is at the correct x position. ✓
Reeborg is at the correct y position. ✓

Escaping the Maze



140/140

Reeborg's basic keyboard

Reeborg says: I'm done!

Reeborg is at the correct x position. ✓
Reeborg is at the correct y position. ✓

```
1 def turn_right():
2     turn_left()
3     turn_left()
4     turn_left()
5
6 while front_is_clear():
7     move()
8
9 while not at_goal():
10    if right_is_clear():
11        turn_right()
12        move()
13    elif front_is_clear():
14        move()
15    else:
16        turn_left()
17
18
19
20
21
```