# A dummy agent – Launching Python Programming

In this lab session, you are required to define an Agent (class) that is able to move on a 6x6 grid board:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| (0,0) | (0,1) | (0,2) | (0,3) | (0,4) | (0,5) |
| (1,0) | (1,1) | (1,2) | (1,3) | (1,4) | (1,5) |
| (2,0) | (2,1) | (2,2) | (2,3) | (2,4) | (2,5) |
| (3,0) | (3,1) | (3,2) | (3,3) | (3,4) | (3,5) |
| (4,0) | (4,1) | (4,2) | (4,3) | (4,4) | (4,5) |
| (5,0) | (5,1) | (5,2) | (5,3) | (5,4) | (5,5) |

1. It has some basic information about itself (Name, current position (x,y), battery level). If the battery level is 0, it cannot do anything further.
2. It can move (“up”, ”down”, ”right”, ”left”), each move uses 1% of its battery.
3. It can not move out of the board or it dies.
4. At a particular cell, it can randomly choose its action. For example, at (3,2), it can choose to move “up”/”down”/”left”/”right”. At (0,0), it can only move “down”/”right”, at (0,1), only “left”/”down”/”right”
5. It always starts at (0,0) and if it happens to land on (5,5) the game finishes. If it runs out battery, the game finishes.
6. Once the game finishes, you are required to print out the trajectory it has gone through ( (0,0),(0,1),(1,1)….) and calculate/print how many steps it’s taken.

Some hints:

You need a decision function which considers the current position of the agent to find out the next possible move.

To generate random actions, you can use random.choice(list) to select a move action from a list randomly.