**Practice "Snake Maze"**

In the same project, take the robot out of the "snake" maze.

Start the project and explore this type of maze yourself.

Additional restrictions:

1. It is forbidden to use more than one cycle in one method.

2. It is forbidden to have methods longer than 12 lines of code.

3. It is forbidden to use the catch keyword

4. It is allowed to create helper methods, but only with friendly names, including argument names.

Please note that your code does not have duplicate almost identical methods.

// Paste the final file content here SnakeMazeTask.cs

namespace Mazes

{

public static class SnakeMazeTask

{

public static void MoveOut(Robot robot, int width, int height)

{

int right = width - 3, down = 2;

while (robot.Finished == false)

{

MoveRight(robot, right);

MoveDown(robot, down);

MoveLeft(robot, right);

if (robot.Finished == false)

{ MoveDown(robot, down); }

}

}

public static void MoveRight (Robot robot, int a)

{

for (int i = 0; i < a; i++)

{ robot.MoveTo(Direction.Right); }

}

public static void MoveLeft(Robot robot, int a)

{

for (int i = 0; i < a; i++)

{ robot.MoveTo(Direction.Left); }

}

public static void MoveDown (Robot robot, int a)

{

for (int i = 0; i < a; i++)

{ robot.MoveTo(Direction.Down); }

}

}

}