World of Robots

Master 1 Génie Physiologique et Informatique

Lab 6.1 : Reconsidering your objectives

```
document.getElementById(div).
else if (i==2)
{
  var atpos=inputs[i].indexOf("@");
  var dotpos=inputs[i].lastIndexOf(".");
  if (atpos<1 || dotpos<atpos+2
  document.getElementById('errEmail')
  else
  document.getElementById(div).
}
  if (i==5)
```

Group 14 : Laurène DONG, Marion DOSPITAL, Lauranne LAIR, Thibault NIGGEL, Florian PAILLAS, Ophélie SOCHARD

Our goal is to build upon all the projects we have completed so far to create the World of Robots project. Following the given instructions, our objective is to create 6 robots with their own specific features that must not collide and must not exit the world. Here are the details of the six child classes of the Robot class to successfully complete this project.

The first subclass of the parent class Robot is the 'Scholar Robot.' It shares several methods with its parent class, such as displaying and modifying its name (set and getName()), its position (set and getX/Y position), and its color. However, it has two distinct features: the ability to speak, perform simple calculations involving addition, subtraction, division, and multiplication between two input numbers and it has a movement only in diaguonal. This method handles both integers and decimal numbers and respects the exception of division by 0, which is impossible.

Savant
- parler : String - goingRight: boolean - goingDown: boolean

+ Savant(String, WorldOfRobot)
+ setParler(String)
+ doubleEffectuerCalcul(double, double, char)
+ move()

The second subclass of the parent class Robot is the 'Color Robot.' It randomly changes its color during its right-to-left movements in the world of robots. It inherits the attributes and methods of the parent class and has its own unique features: a method to randomize the color of its body with each movement and a method for moving from right to left.

Couleur
- randomGenerator : Random - goingRight : boolean - couleur : int

+ Couleur(String, WorldOfRobot)
+ randomCouleur()
+ move()

The third subclass of the parent class Robot is the 'Dancing Robot.' This marvelous robot knows how to position itself in the middle of the dance floor to attract attention and dances with pride and grace in two different styles: the Madison and the Waltz. It inherits various attributes and methods from the parent class and utilizes the draw() method from the parent class in its various dance methods. To move, it uses the x and y positions.

Danseur

+ Danseur(String, WorldOfRobot)
+ move()
+ Valse()

The fourth subclass of the parent class Robot is the 'Random Robot.' This robot moves randomly within the world. It inherits the attributes and methods of the parent class through inheritance. It has its own attributes and methods:

Randome
-

+ Randome(String, int , int , WorldOfRobot)
+ Random010()
+ move ()

The fifth subclass of the parent class Robot is the 'Drawing Robot.' This robot is capable of moving according to the user's desires while drawing a line along its path. It inherits the attributes and methods of the Robot class and has its own attributes and methods:

Dessin
- colourLine : String - lesCouleurs[] : String - trace : boolean

+ Dessin(String, int , int , WorldOfRobot)
+ getColourLine()
+ setColourLIne(String)
+ rendCouleur(String)
+ setTracer(boolean)
+ moveHorizontal(int)
+ moveVertical(int)
+ move()

The final subclass of the parent class Robot is the 'Spiral Robot.' It has the ability to move while spiraling. The Scanner Robot inherits the attributes and methods of the Robot class and has its own attributes and methods:

Spiral
- trajectory: int - spiraleRate: int

- xtemp: int - ytemp: int

+ SpiralRobot(String, WorldOfRobot)
+ move()

This project adheres to Asimov's law regarding the movement of robots, implements encapsulation, polymorphism, and inheritance. Finally, the robots are not capable of self-destruction. Thanks to the world that displays robots, it is possible to either make them move one by one or make them all move at the same time.