



ALICE

*Say mê và
sáng tạo*



Giáo viên hướng dẫn

❖ Trần Phúc Minh Khoa

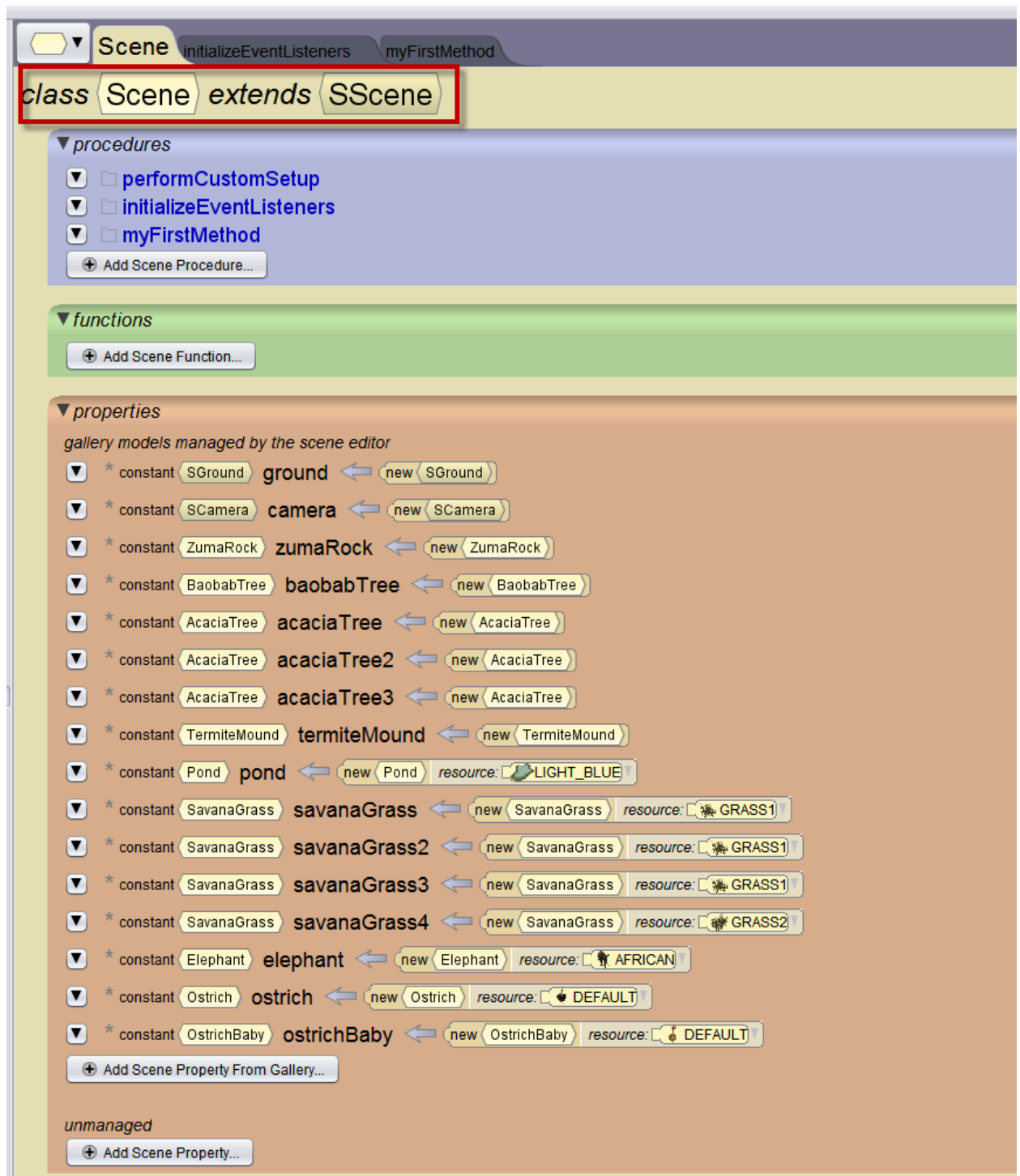
❖ Nguyễn Võ Phi Long

Lập trình với lớp Scene

Khi khởi tạo thành công một project Alice thì Trình chỉnh sửa Code (Code Editor) sẽ tự động tạo sẵn **3 thẻ**: **Scene (Lớp Scene)**, **initializeEventListeners**, **myFirstMethod** (2 thủ tục của lớp Scene)

1. Tab Scene (Thẻ dành cho lớp Cảnh)

Một lớp là một định nghĩa của một đối tượng cụ thể. Lớp Scene xác định cách tạo cảnh trong thế giới 3D của Alice. Trong Tab Scene chứa các định nghĩa cho các thủ tục (các phương thức hành động), các hàm (các phương thức tính toán), và các thuộc tính (các trường - đối tượng hoặc các mục dữ liệu, chẳng hạn như màu hoặc độ mờ). (Hình 3)



Hình 3: Tab Scene

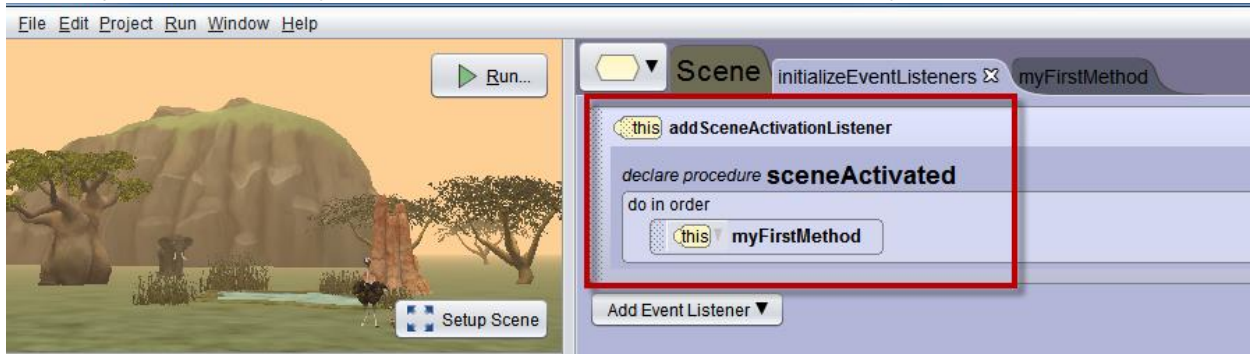
2. Tab initializeEventListeners

Đây là một thủ tục của lớp Scene. Thủ tục này giúp lắng nghe các "Sự kiện". Sự kiện là gì? Sự kiện là những tương tác người dùng sẽ được chương trình lắng nghe khi chương trình chạy. Ví dụ khi người lập trình lập trình cho chiếc máy bay điều khiển sang trái, phải bằng các nút mũi tên trái, phải trên bàn phím, thì đến khi người dùng chạy chương trình và nhấn nút mũi tên trái, phải

trên bàn phím thì chương trình sẽ lắng nghe và thực hiện theo những gì đã lập trình và điều khiển máy bay sang trái và phải. (Việc lập trình cho thủ tục này rất quan trọng, nếu muốn Sự kiện diễn ra đúng người dùng phải lập trình Sự kiện đúng)

Theo mặc định, tab initializeEventListeners của lớp Scene có một trình nghe sự kiện được tích hợp sẵn là addSceneActivationListener (Hình 4), cho phép khi người dùng chạy chương trình thì thủ tục myFirstMethod được gọi (thực hiện) và các lệnh ở trong myFirstMethod sẽ được chạy. Ta cũng có thể xóa sự kiện này.

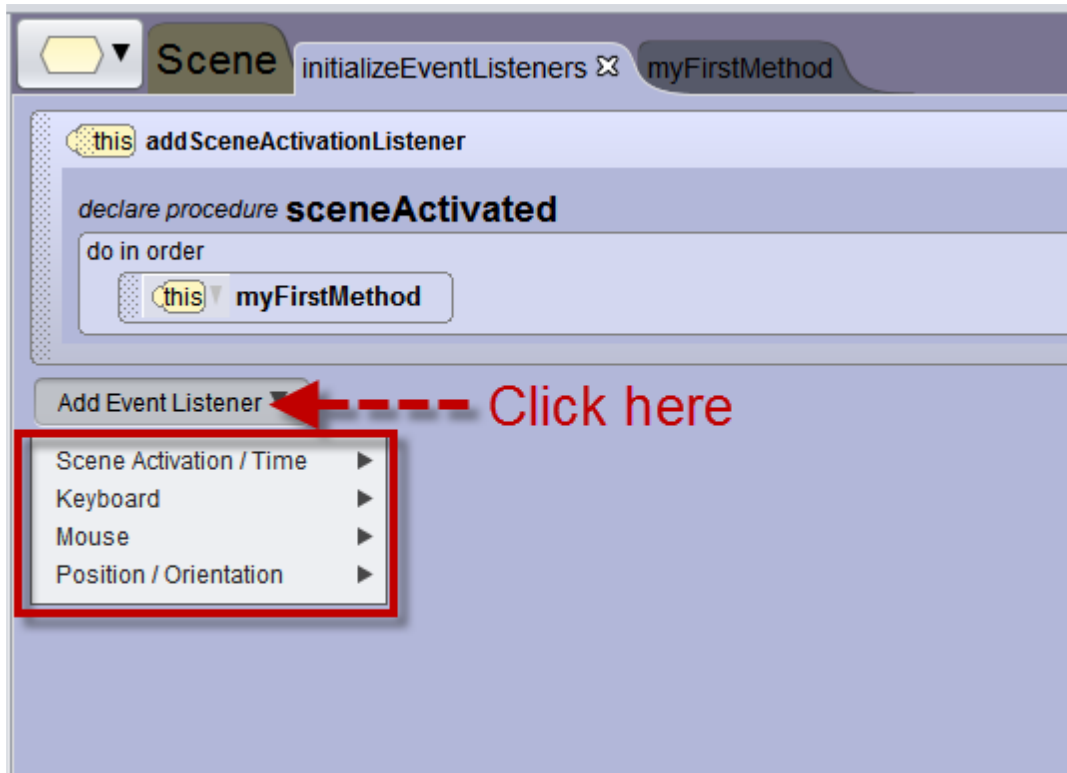
Tại đây ta có thể chạy một thủ tục khác hoặc xóa thủ tục myFirstMethod.



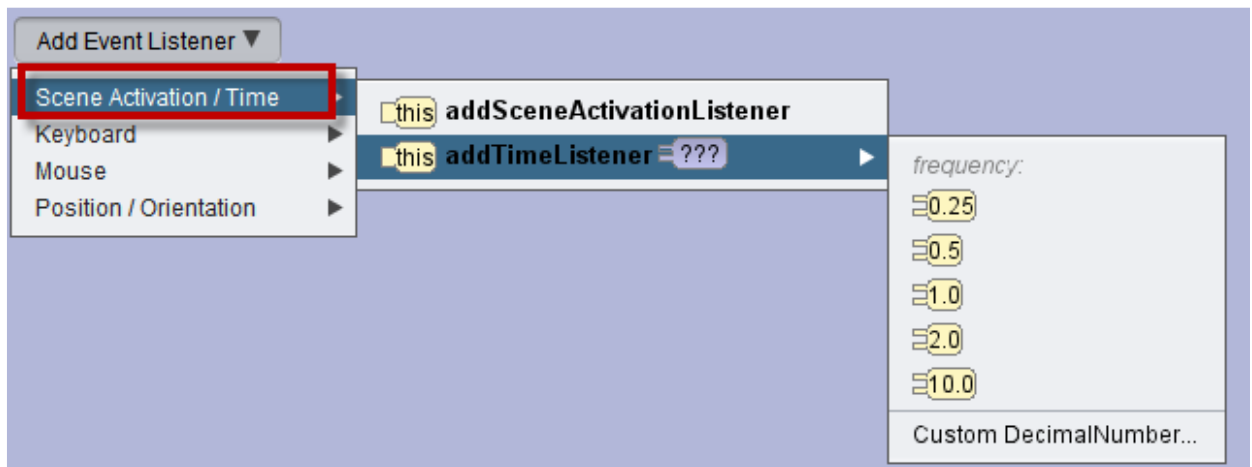
Hình 4: Sự kiện được tích hợp sẵn là addSceneActivationListener

Để thêm sự kiện mới, hãy nhấp vào nút AddEventListener. Khi chọn vào AddEventListener sẽ xuất hiện danh mục sự kiện khác nhau (Hình 5). Các loại sự kiện:

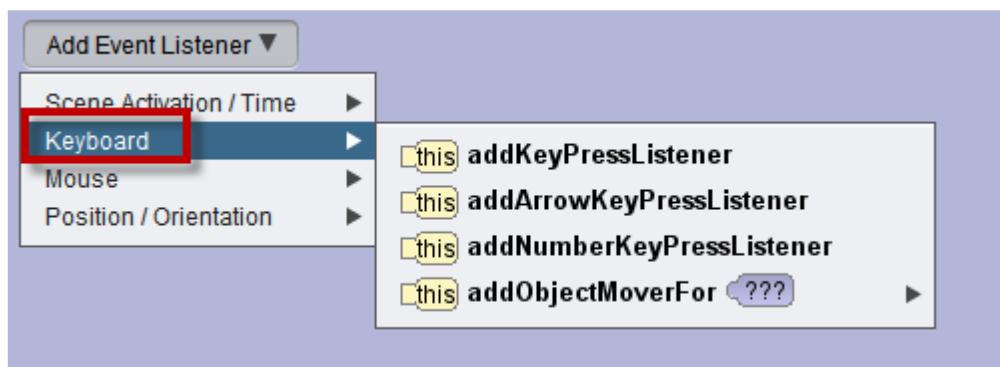
- ☆ **Khởi động sự kiện/ Thời gian bắt đầu sự kiện:** Khởi động sự kiện: Các sự kiện bên trong sẽ được thực hiện ngay khi bấm Run. Thời gian bắt đầu sự kiện: Sau một khoảng thời gian thì sự kiện sẽ bắt đầu (Hình 6)
- ☆ **Sự kiện nhập từ bàn phím:** Sự kiện nhập từ bàn phím: Khi chương trình chạy, chương trình sẽ lắng nghe người dùng nhập gì từ bàn phím và thao tác theo lệnh đã lập trình (Hình 7)
- ☆ **Sự kiện chuột (Mouse):** Sự kiện chuột: Khi chương trình chạy, chương trình sẽ lắng nghe click chuột trái từ người dùng và chạy theo những gì đã lập trình. (Hình 8)
- ☆ **Sự kiện vị trí và định hướng:** Sự kiện vị trí và định hướng: Sự kiện này sẽ xét về vị trí của vật hoặc khoảng cách của 2 vật (chạm nhau, di chuyển lại gần ...) để thực hiện lệnh đã được lập trình. (Hình 9)



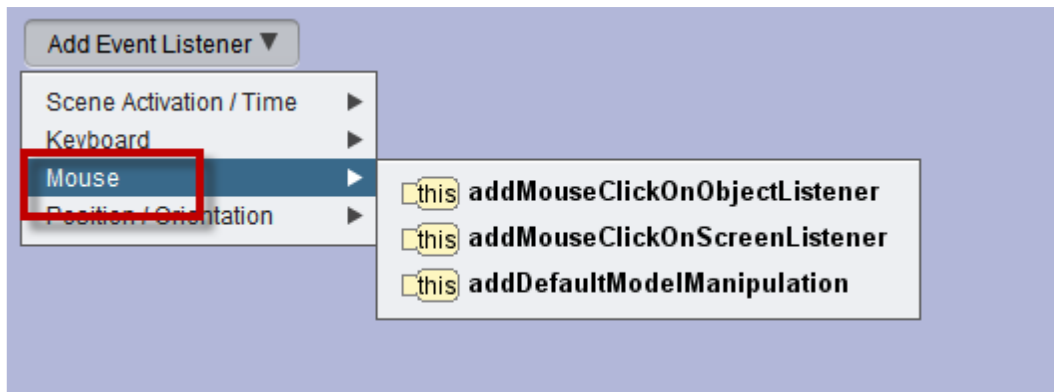
Hình 5: Danh sách sự kiện



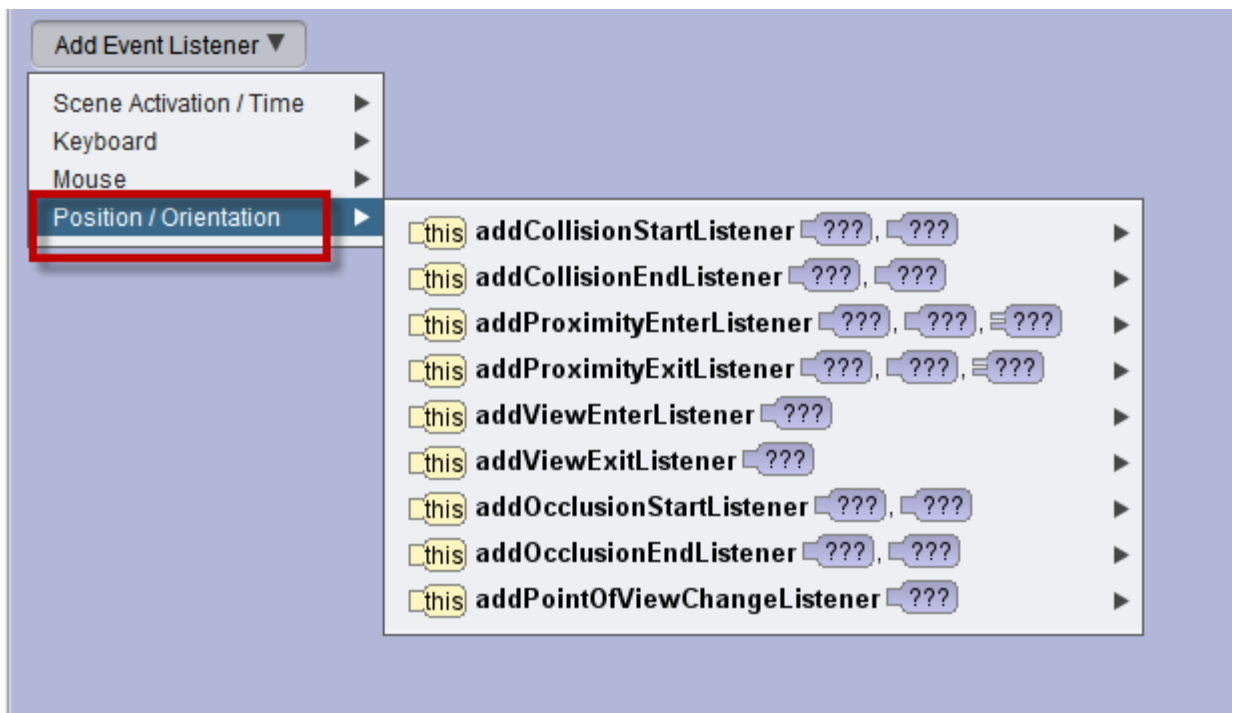
Hình 6: Khởi động sự kiện/ Thời gian bắt đầu sự kiện



Hình 7: Sự kiện nhập từ bàn phím



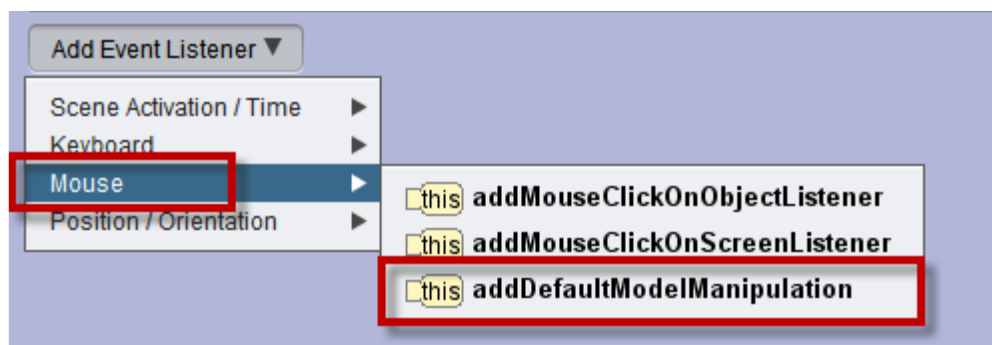
Hình 8: Sự kiện chuột (Mouse)



Hình 9: Sự kiện vị trí và định hướng

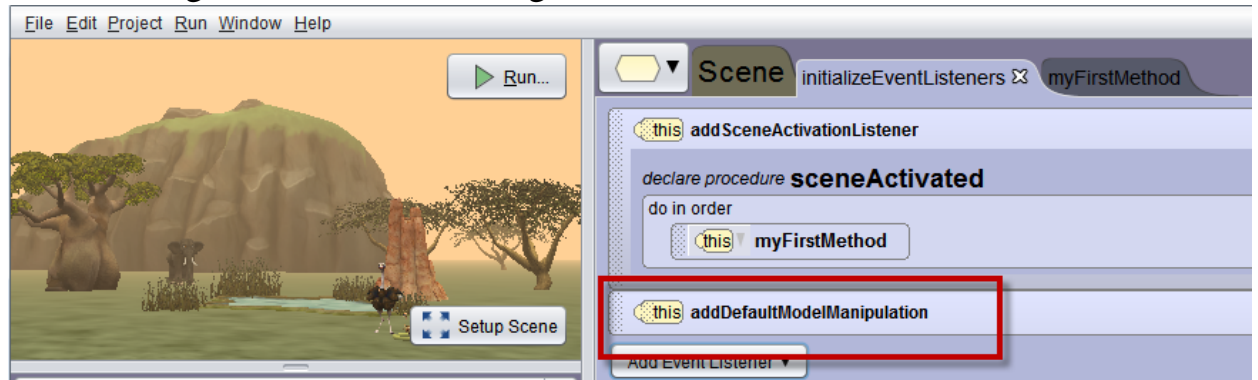
Ví dụ thêm sự kiện:

Giả sử khi chạy chương trình cho phép người dùng di chuyển các đối tượng xung quanh trong cảnh thì chọn danh Sự kiện chuột (Mouse) và sau đó chọn addDefaultModelManipulation trong danh sách sự kiện. (Hình 10)



Hình 10: Sự kiện `addDefaultModelManipulation`

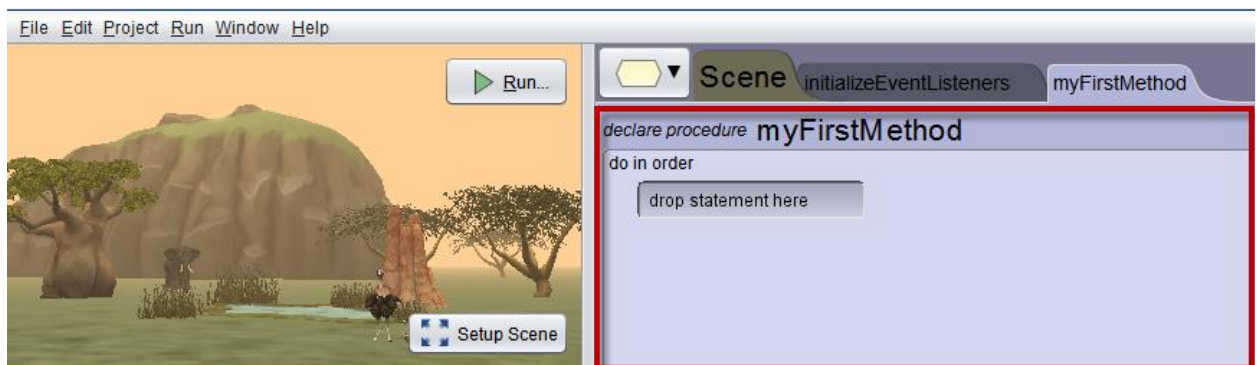
Kết quả: Sự kiện đã được thêm vào. Và 2 sự kiện được xếp song song nhau nên khi chạy chương trình 2 sự kiện này sẽ chạy cùng một lúc. (Hình 11). Đối với sự kiện vừa thêm vào thì bất kỳ đối tượng nào trong cảnh có thể được ta dùng chuột kéo và thả trong hoạt cảnh.



Hình 11: Kết quả

3. Tab `myFirstMethod`

Tab `myFirstMethod` là nơi ta có thể tạo chương trình bằng các câu lệnh mà ta mong đợi sẽ được thực hiện khi nút Run là đã nhấp. Bên trong thủ tục `myFirstMethod` có 1 mã đầu tiên là “do in order”, đoạn mã này giúp cho các câu lệnh được xây dựng trong thủ tục chạy theo thứ tự từ trên xuống. (Hình 12).

Hình 12: Tab `myFirstMethod`