



Physique collégiale de 12e année

# SIMULATEUR DE TRANSFORMATIONS ÉNERGÉTIQUES

Lien pour les étudiants

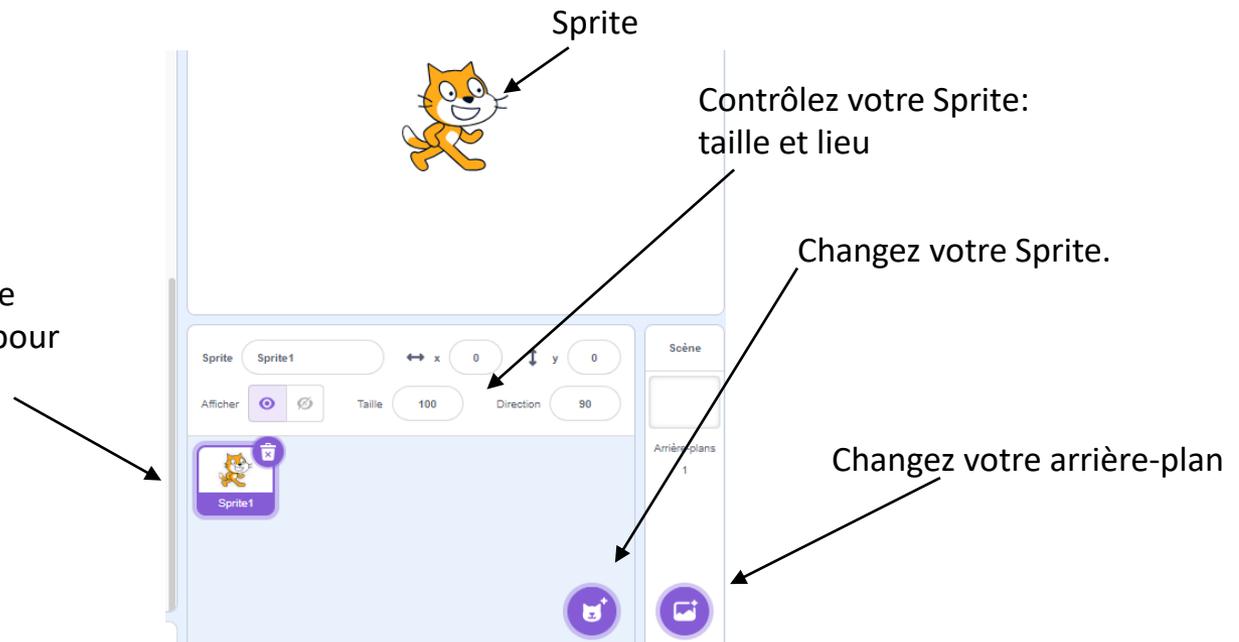
<https://scratch.mit.edu/>

Solution

<https://scratch.mit.edu/projects/882782185/>

# Apprendre à connaître Scratch !

Choisissez votre Sprite avant de coder. Cliquez sur la poubelle pour supprimer le Sprite.

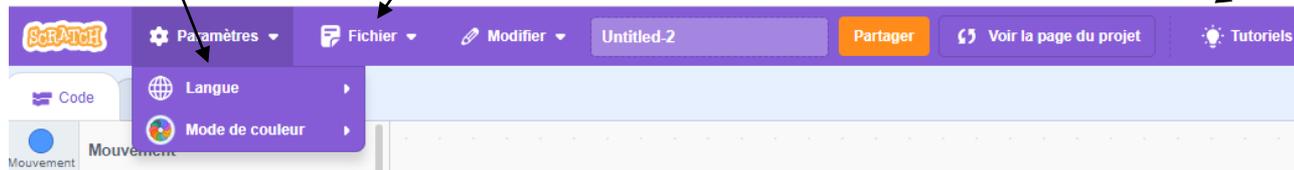


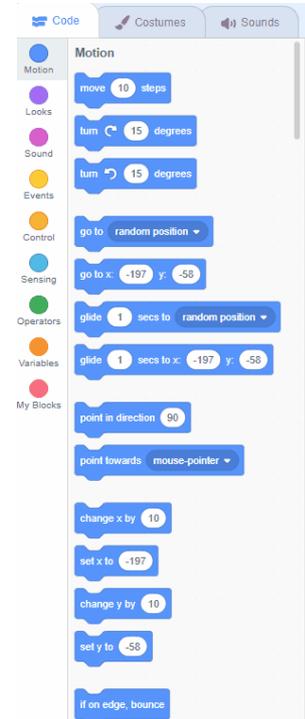
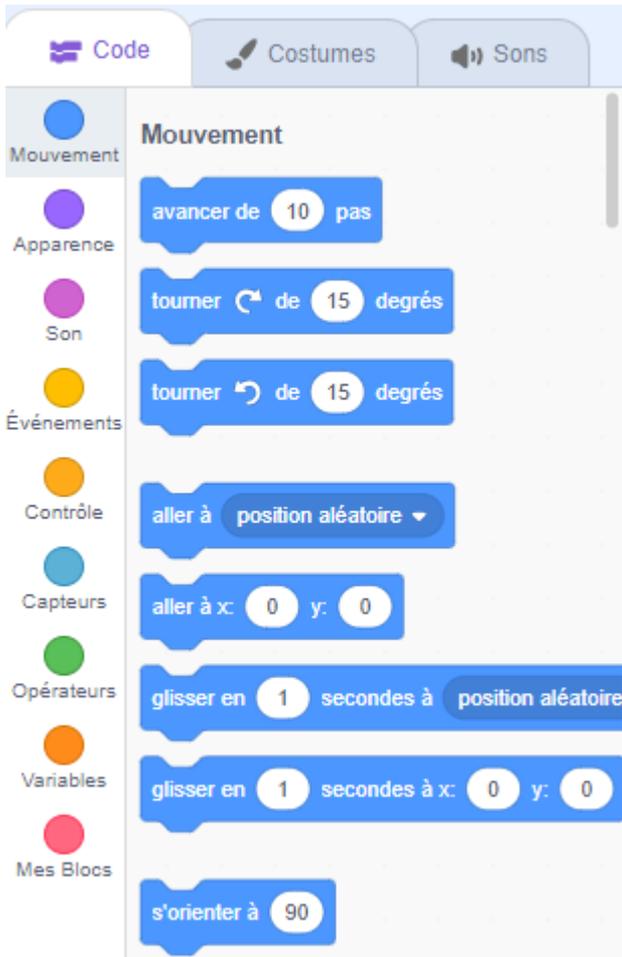
# Apprendre à connaître Scratch !

Cliquez ici pour enregistrer un projet ou ouvrir un projet depuis votre ordinateur

Cliquez ici pour changer la langue

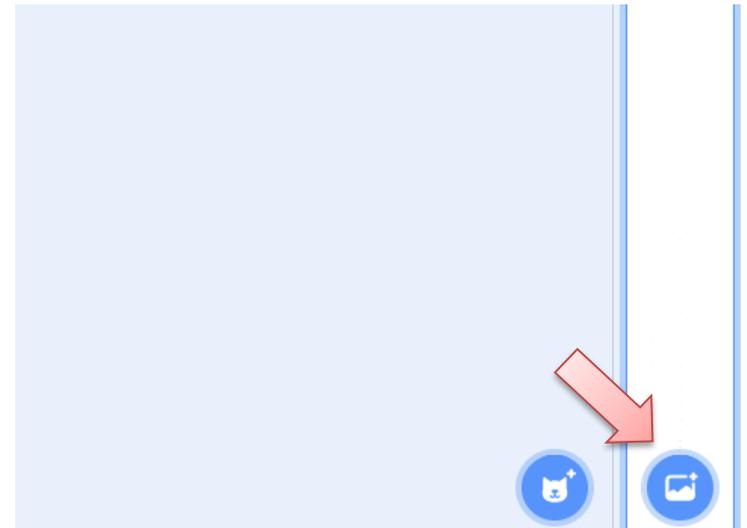
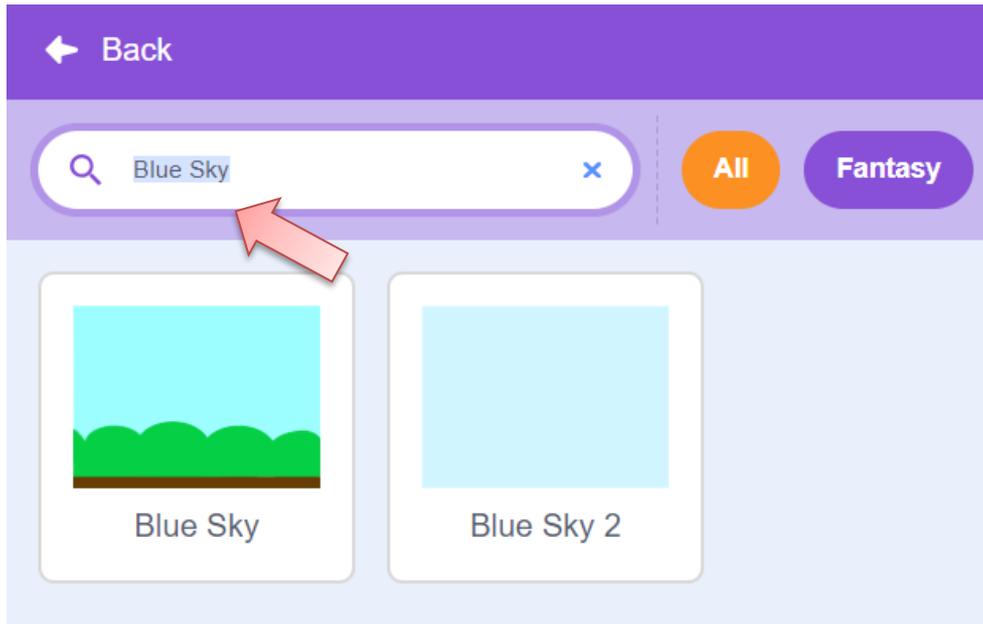
Utilisez les tutoriels pour apprendre



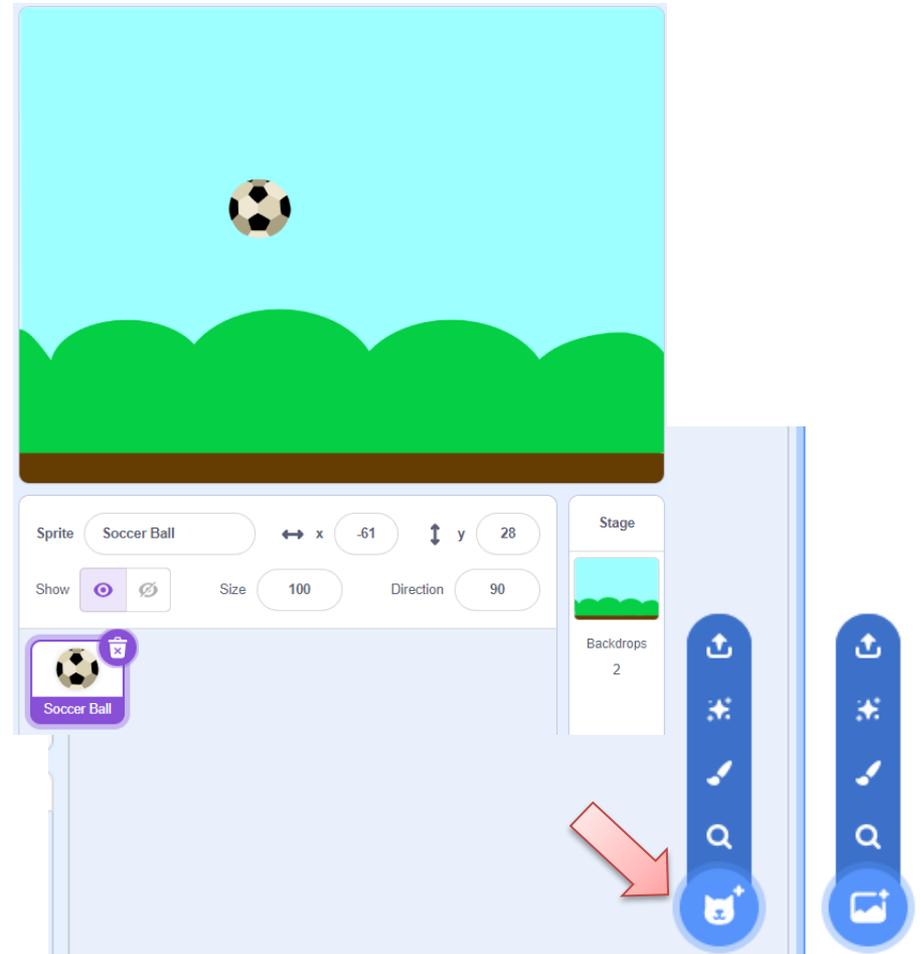
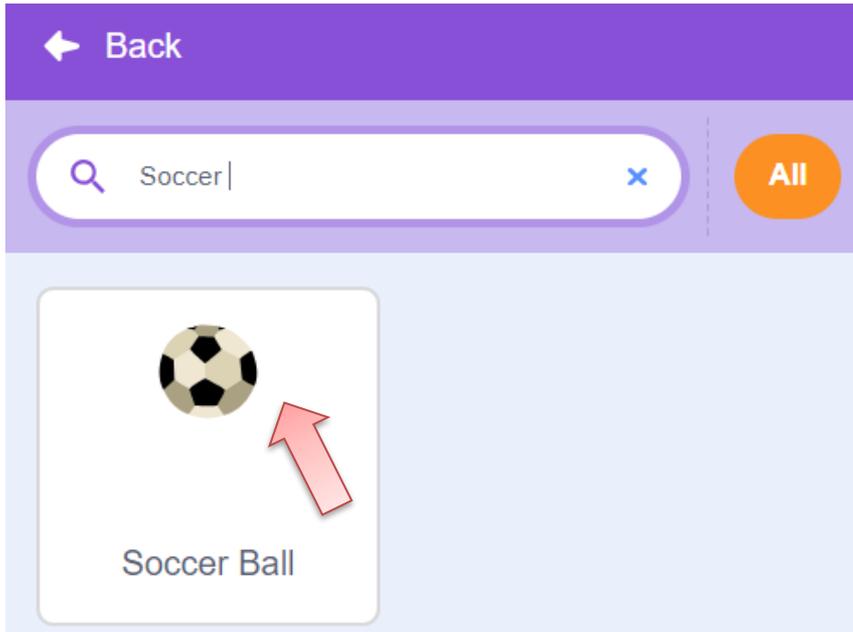


Cliquez et tirez les blocs de codage de la droite vers la zone ouverte de gauche pour créer votre code.

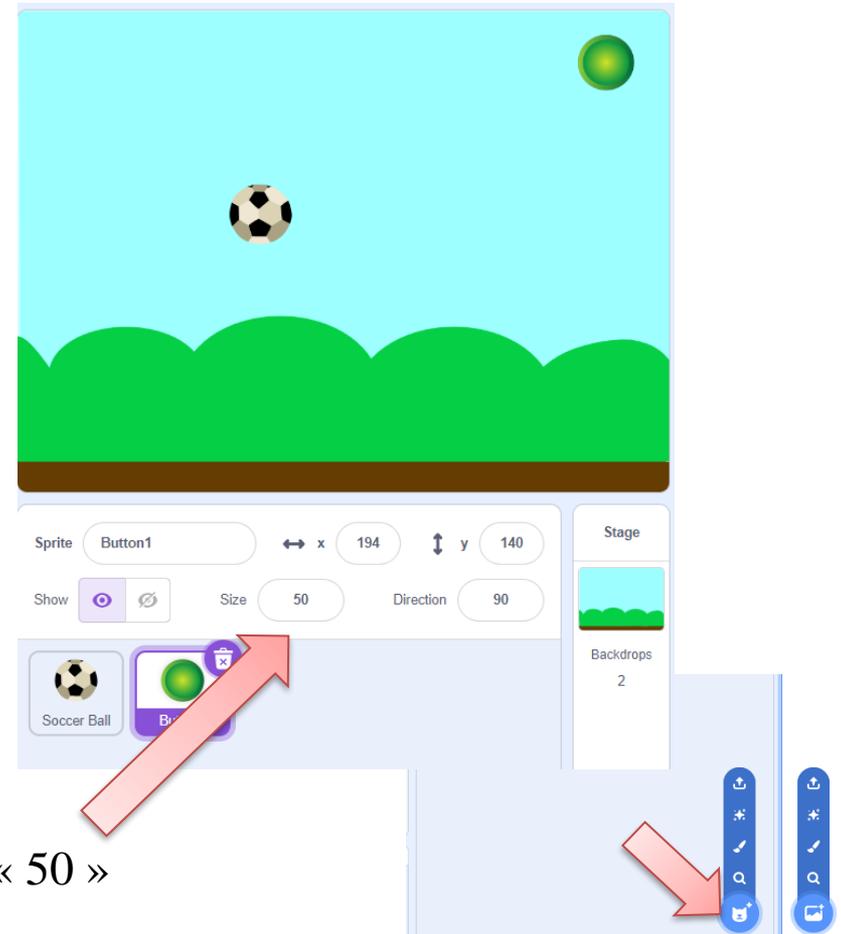
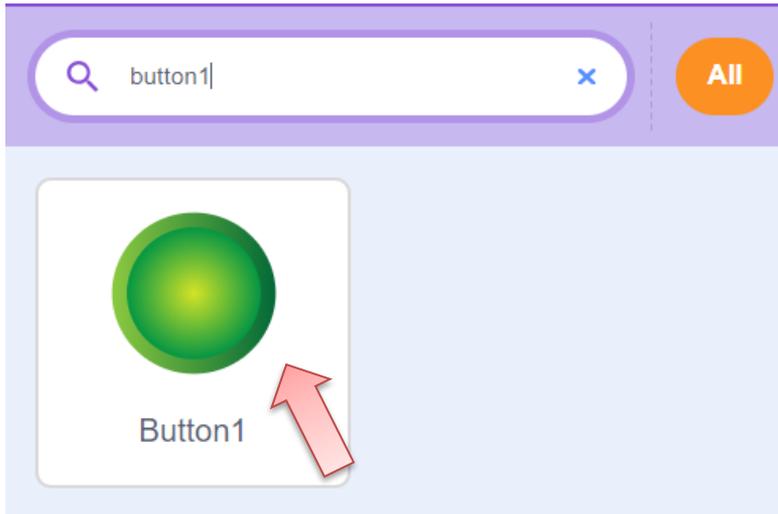
# Choisir une arrière-plan "Blue Sky"



# Choisir un Sprite “Soccer Ball”



# Choisir un Sprite "Button1"



Taille « 50 »

# Créer des variables pour le sprite "Soccer Ball".

Code Costumes Sons

Mouvement

Apparence

Son

Événements

Contrôle

Capteurs

Opérateurs

Variables

Mes Blocs

**Variables**

Créer une variable

Ep

hauteur

ma variable

masse

vitesse

mettre Ep à 0

ajouter 1 à Ep

montrer la variable Ep

cacher la variable Ep

Créer une liste

**Mes Blocs**

Créer un bloc

Énergie potentielle de la balle

La hauteur de chute

Sa masse

Vitesse au moment de l'impact au sol

# Créer un programme pour le sprite "Soccer Ball".

Notez l'ordre des opérations  
 $-150 + (\text{hauteur} * 10)$

```
quand le drapeau est cliqué  
mettre hauteur à 0  
mettre vitesse à 0  
mettre Ep à 0  
mettre masse à 0  
aller à x: -190 y: -70  
demander "De quelle hauteur (max. 30 m) la balle tombe-t-elle ? (entrez un nombre)" et attendre  
mettre hauteur à réponse  
glisser en 1 secondes à x: 0 y: -150 + hauteur * 10  
demander "Quelle est la masse(kg) de la balle?" et attendre  
mettre masse à réponse  
quand je reçois message1  
glisser en racine de 2 * hauteur / 9.8 secondes à x: 0 y: -150
```

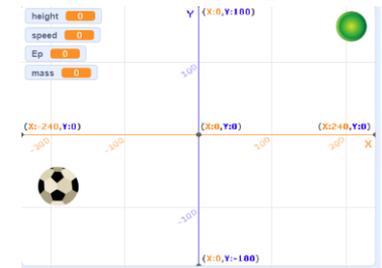
Ep 0  
hauteur 15  
masse 5  
vitesse 0

Sprite Soccer Ball x 0 y -150  
Afficher Taille 100 Direction 90

Scène  
Arière-plans 2

# Explication du programme pour le sprite "Soccer Ball".

coordinate plane



```
quand est cliqué
mettre hauteur à 0
mettre vitesse à 0
mettre Ep à 0
mettre masse à 0
aller à x: -190 y: -70

demander De quelle hauteur (max. 30 m) la balle tombe-t-elle ? (entrez un nombre) et attendre
mettre hauteur à réponse

glisser en 1 secondes à x: 0 y: -150 + hauteur * 10

demander Quelle est la masse(kg) de la balle? et attendre
mettre masse à réponse

quand je reçois message 1
glisser en racine de 2 * hauteur / 9.8 secondes à x: 0 y: -150
```

Nous mettons toutes les variables à zéro au début de la simulation et déplaçons la balle à sa position initiale.

Nous demandons la hauteur à partir de laquelle la balle tombera et nous la tapons dans le champ.

Nous déplaçons la balle dans le plan de coordonnées, dont le centre est 0,0.

Nous demandons la masse et l'enregistrons dans une variable.

Nous recevons un message de clic du bouton et nous simulons la chute de la balle conformément à la loi physique :

$$t = \sqrt{\frac{2h}{g}}$$

# Créer un programme pour le sprite "Button1".

The image shows a programming environment with a scene of a soccer field. A green ball is in the top right corner. A soccer ball sprite is on the field. A purple button sprite labeled "Button1" is in the bottom right. The code for "Button1" is:

```
quand ce sprite est cliqué  
envoyer à tous message1
```

The scene properties are:

- Sprite: Button1
- x: 200
- y: 145
- Afficher:
- Taille: 50
- Direction: 90

The scene background is a soccer field with green hills and a brown ground. The scene is named "Scène" and has 2 arrière-plans.

# Exécuter et tester le programme

The image displays the Scratch code editor and stage for a physics simulation of a falling soccer ball. The code is as follows:

```
when green flag clicked
  set height to 0
  set speed to 0
  set Ep to 0
  set mass to 0
  go to x: -190 y: -70
  ask "From what height ( max 30 meters) I fall? (enter only a number)" and wait
  set height to answer
  glide 1 secs to x: 0 y: -150 + height * 10
  ask "What is my mass? (kg)" and wait
  set mass to answer

when I receive message1
  glide sqrt of (2 * height / 9.8) secs to x: 0 y: -150
```

The stage shows a soccer ball sprite on a green field with a light blue sky. The following table shows the current values of the variables defined in the code:

| Variable | Value |
|----------|-------|
| height   | 15    |
| speed    | 294   |
| Ep       | 735   |
| mass     | 5     |

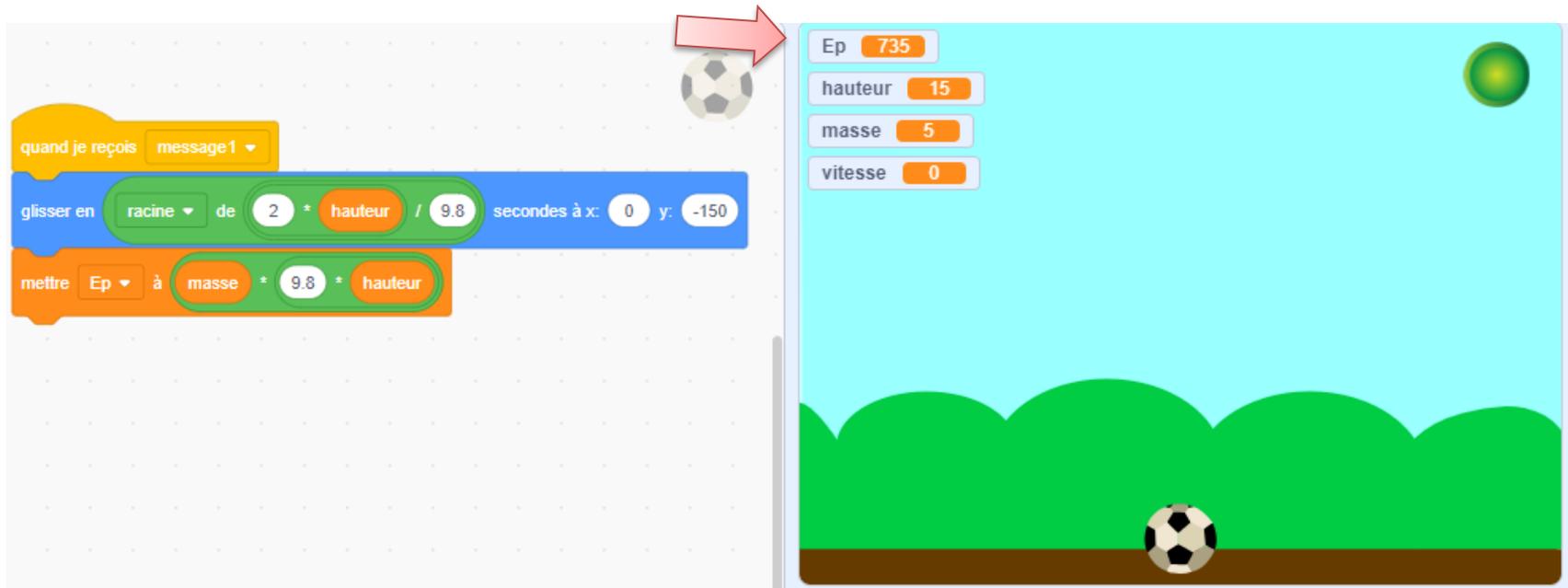
The sprite properties for the Soccer Ball are:

- Sprite: Soccer Ball
- x: 0
- y: -150
- Show:
- Size: 120
- Direction: 90

The stage also shows a backdrop and a list of sprites:

- Backdrops: 5
- Sprites: Soccer Ball, Button1

# Extension 1. Nous ajoutons un affichage de l'énergie potentielle



The image shows a Scratch script on the left and a stage on the right. A red arrow points from the script to the stage.

**Script:**

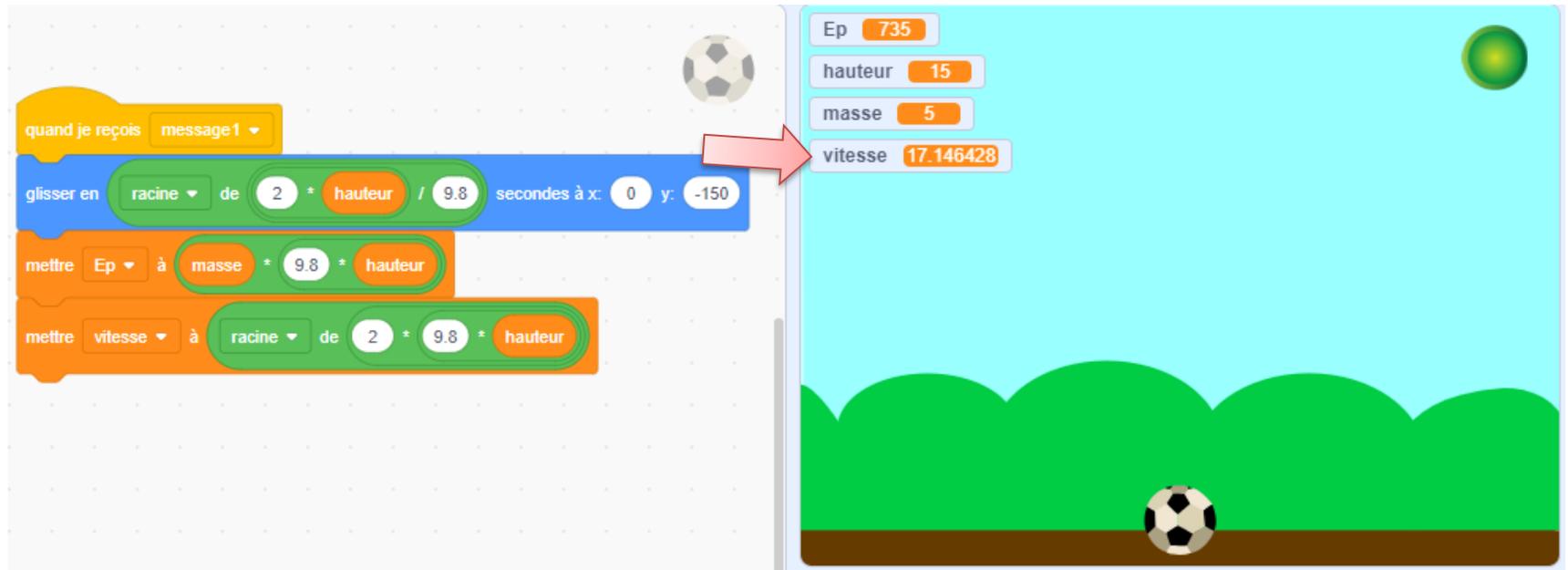
- when I receive message 1
- slide in from the bottom:  $\sqrt{2 \times \text{hauteur} / 9.8}$  seconds to x: 0 y: -150
- set Ep to  $\text{masse} \times 9.8 \times \text{hauteur}$

**Stage:**

- Ep: 735
- hauteur: 15
- masse: 5
- vitesse: 0

The stage features a soccer ball on a brown ground line, green hills, and a light blue sky with a sun in the top right corner.

# Extension 2. Nous ajoutons un affichage de la vitesse



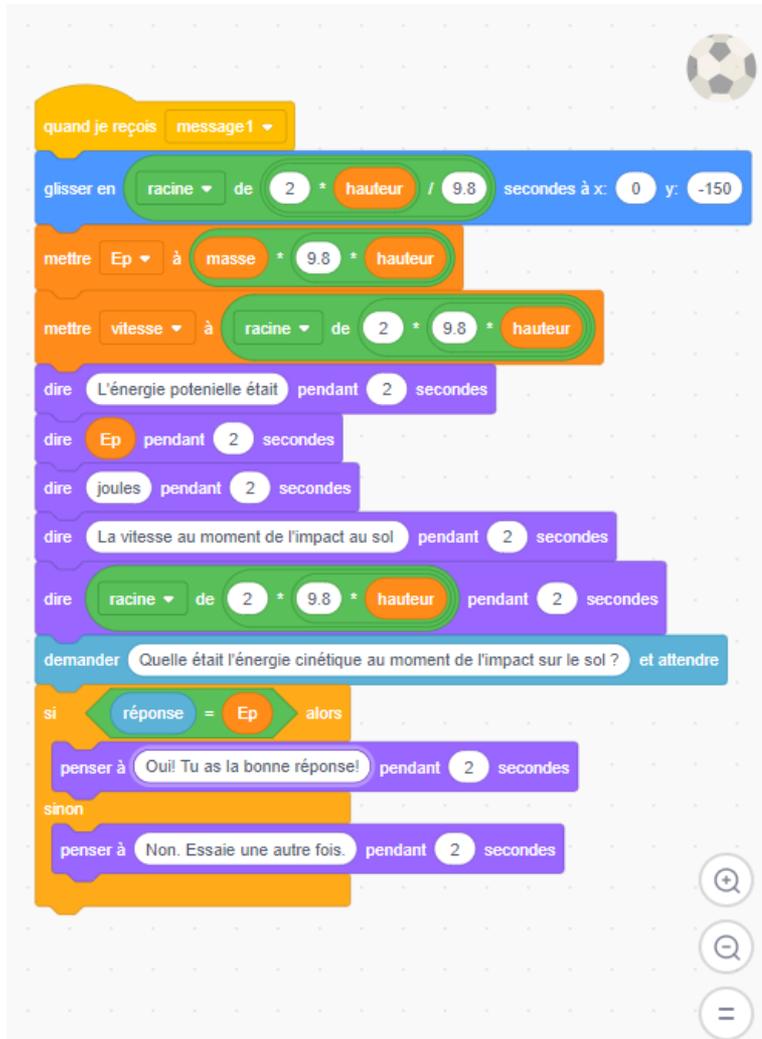
The image shows a Scratch script on the left and a stage on the right. The script consists of three blocks:

- quand je reçois message1** (yellow block)
- glisser en racine de  $2 * hauteur / 9.8$  secondes à x: 0 y: -150** (blue block)
- mettre vitesse à racine de  $2 * 9.8 * hauteur$**  (orange block)

The stage on the right shows a soccer ball on a green field with a brown ground line. A red arrow points from the 'mettre vitesse à' block in the script to the 'vitesse' property in the stage's information panel. The information panel displays the following values:

- Ep: 735
- hauteur: 15
- masse: 5
- vitesse: 17.146428

## Extension 3.



The image shows a Scratch script for a physics simulation. The script starts with a 'when I receive message 1' event. It then calculates the time of fall using the formula  $t = \sqrt{2 \cdot \text{hauteur} / 9.8}$  and moves the ball to x: 0, y: -150. It sets the potential energy  $E_p = \text{masse} \cdot 9.8 \cdot \text{hauteur}$  and the velocity  $v = \sqrt{2 \cdot 9.8 \cdot \text{hauteur}}$ . The script then displays several messages: 'L'énergie potentielle était', 'Ep', 'joules', 'La vitesse au moment de l'impact au sol', and the velocity formula, each for 2 seconds. It then asks the user 'Quelle était l'énergie cinétique au moment de l'impact sur le sol ?' and waits for a response. If the response equals  $E_p$ , it says 'Oui! Tu as la bonne réponse!' for 2 seconds. Otherwise, it says 'Non. Essaie une autre fois.' for 2 seconds.

```
quand je reçois message 1
glisser en racine de 2 * hauteur / 9.8 secondes à x: 0 y: -150
mettre Ep à masse * 9.8 * hauteur
mettre vitesse à racine de 2 * 9.8 * hauteur
dire L'énergie potentielle était pendant 2 secondes
dire Ep pendant 2 secondes
dire joules pendant 2 secondes
dire La vitesse au moment de l'impact au sol pendant 2 secondes
dire racine de 2 * 9.8 * hauteur pendant 2 secondes
demander Quelle était l'énergie cinétique au moment de l'impact sur le sol ? et attendre
si réponse = Ep alors
  penser à Oui! Tu as la bonne réponse! pendant 2 secondes
sinon
  penser à Non. Essaie une autre fois. pendant 2 secondes
```

Nous ajoutons de l'interactivité et démontrons le temps de chute de la balle.

# Programme complet

The image displays a complete Scratch program for a physics simulation. The code is organized into three main sections:

- Initialization (Left):** A 'when clicked' event triggers the following actions:
  - Set `hauteur` to 0.
  - Set `vitesse` to 0.
  - Set `Ep` to 0.
  - Set `masse` to 0.
  - Go to x: -190, y: -70.
  - Ask: "De quelle hauteur (max. 30 m) la balle tombe-t-elle ? (entrez un nombre) and wait".
  - Set `hauteur` to `answer`.
  - Glide 1 sec to x: 0, y:  $-150 + hauteur * 10$ .
  - Ask: "Quelle est la masse(kg) de la balle? and wait".
  - Set `masse` to `answer`.
- Simulation (Middle):** A 'when I receive message1' event triggers:
  - Glide  $\sqrt{2 * hauteur / 9.8}$  secs to x: 0, y: -150.
  - Set `Ep` to  $masse * 9.8 * hauteur$ .
  - Set `vitesse` to  $\sqrt{2 * 9.8 * hauteur}$ .
  - Say "L'énergie potentielle était" for 2 seconds.
  - Say `Ep` for 2 seconds.
  - Say "joules" for 2 seconds.
  - Say "La vitesse au moment de l'impact au sol" for 2 seconds.
  - Say  $\sqrt{2 * 9.8 * hauteur}$  for 2 seconds.
  - Ask: "Quelle était l'énergie cinétique au moment de l'impact sur le sol? and wait".
  - If `answer = Ep` then:
    - Think "Oui! Tu as la bonne réponse!" for 2 seconds.
  - Else:
    - Think "Non. Essaie une autre fois." for 2 seconds.
- Game Scene (Right):** The visual interface shows:
  - Stage with a green ground and blue sky.
  - A soccer ball sprite at x: 0, y: -150.
  - Control panels for `Ep` (735), `hauteur` (15), `masse` (5), and `vitesse` (17.146428).
  - A speech bubble containing the number 735.
  - Sprite and Stage panels at the bottom.