Programming

Yuhuiii, we hit the can. This gives us 1 point.

The can hops when it gets hit.

```
when I receive hit
change points by 1
point in direction pick random 1 to 360
set y to y position + 100
change x by pick random -50 to 50
if direction < 90 then
    change x by join - direction / 50
if direction > 90 then
    change x by direction / 50
```
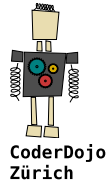
```
when I receive GameOver
hide
stop other scripts in sprite
```

When receiving GameOver message, the game is finished

Next steps

- Change points or speed
- Let the wizard move around
- Sometimes the can is small and sometimes big

CoderDojo Zürich

Shooting cans

Try to shoot down the cans with a lightning as many times as possible

| Zeit | 39 |
| Punkte | -32 |

Programming

When the lightning misses the target, it goes back to the wizard

```
hide
go to x: 200 y: -156
point in direction 0
```

After receiving GameOver message, the game is finished

```
when I receive GameOver
hide
stop other scripts in sprite
```
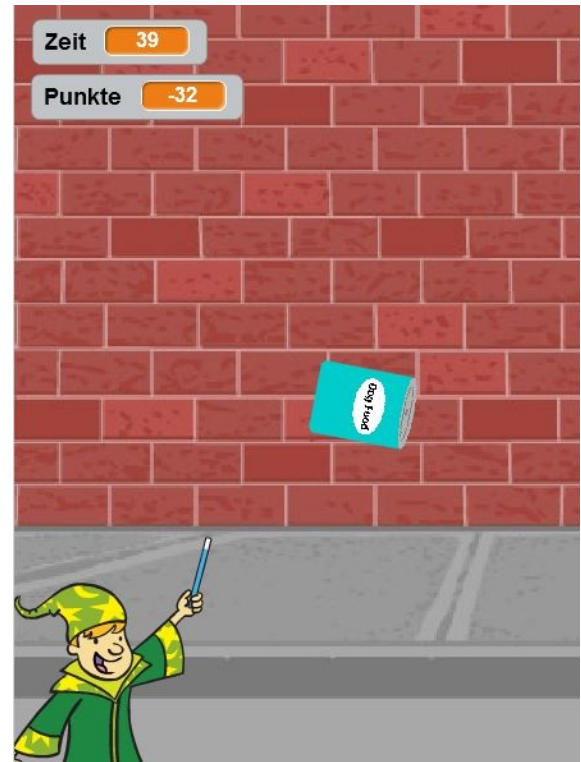
Next steps

• The wizard jumps up when the can is hit
• The wizard says "hit"
• The lightning changes to fire when it hits the can
• A music plays or an explosion is heard

Programming

When clicking on the green flag, we set the time, velocity, points and much more

```
Data          More Blocks

Make a Variable

☐ falling speed
☑ points
☐ shooting direction
☑ time
☐ velocity
```

```
when [green flag] clicked
set time to 60
show variable falling speed
set points to 0
show variable falling speed
set falling speed to -0.2
set velocity to 10
```

Then we set the countdown clock that goes from 60,59,58,57...to 0 (Game Over)

```
repeat 60
    wait 1 secs
    change time by -1
broadcast GameOver
```

Shooting cans

Aim at the can. Click the mouse button to shoot a lightning at the can

When you hit the can, then you receive one point.

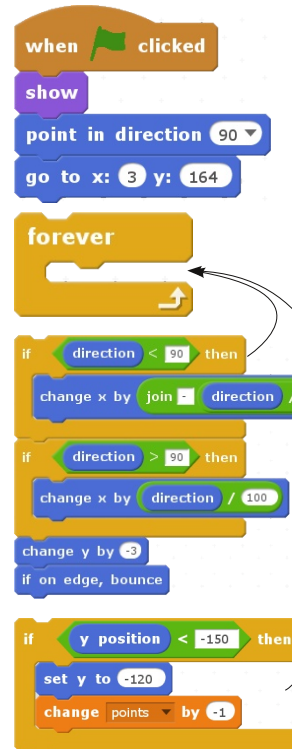Within 60 seconds hit as many cans as possible.

You will need for the game:
- a can,
- something to aim with,
- a wizard and a lightning
- and a cool background.

Pick the sprites from the library or draw them yourself.

First show the can on the screen

```
when [green flag] clicked
show
point in direction (90)
go to x: (3) y: (164)

forever
    if < direction < 90 > then
        change x by (join (-) (direction / 100))
    if < direction > 90 > then
        change x by (direction / 100)
    change y by (-3)
    if on edge, bounce
    if < y position < -150 > then
        set y to (-120)
        change [points] by (-1)
```

The can bounces around. Over and over again.

Hahaha, 1 point less when the can hits the floor

Programming

```
when [green flag] clicked
show
forever
    go to [mouse-pointer]
```
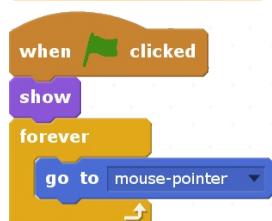
The targetting scope follows the mouse pointer

Here we go!

[green flag] [red stop]

Click on the green flag and the targetting scope will follow the mouse pointer in all directions.
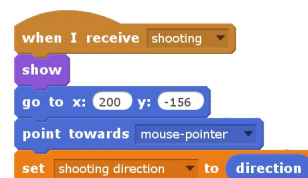
Programming

```
when this sprite clicked
broadcast [shooting]
```

```
when I receive [GameOver]
hide
stop [other scripts in sprite]
```

When the game is over, everything stops
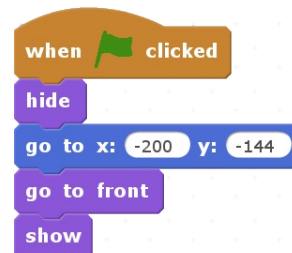
We are putting the wizard in the corner of the screen

```
when [green flag] clicked
hide
go to x: (-200) y: (-144)
go to front
show
```

The lightning starts at wizard's position and moves towards pointer

```
when I receive [shooting]
show
go to x: (200) y: (-156)
point towards [mouse-pointer]
set [shooting direction] to (direction)
```

When we miss the can, the lightning moves to the edge. When the lightning hits the can, we send a message.

```
repeat until < touching [edge] ? >
    if < touching [can] ? > then
        broadcast [hit]
        hide
        go to x: (-200) y: (-156)
        point in direction (0)
        stop [this script]
    else
        move (15) steps
```