

SEAN MCMANUS

Sean McManus is a Code Club volunteer, and wrote the book *Scratch Programming in Easy Steps*. He co-wrote *Raspberry Pi For Dummies*. **sean.co.uk twitter.com/musicandwords**

ADD A TITLE SCREEN TO YOUR SCRATCH GAME

You'll Need

One of your own Scratch games, or you can use our simple example here To make a professional-looking game, follow these steps to add a title screen with instructions and a fun animation

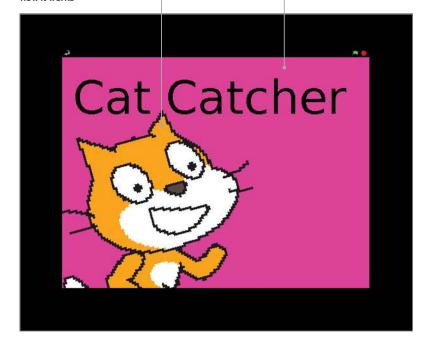
book has a cover, a film has its credits, and an album has its artwork. Only with the right presentation do these things feel professional and complete. In the same way, a great game starts with a title screen that draws players in and provides instructions. It's especially important if you want to share your game, as you won't be there to explain it when it's played. In this article, you'll see how you can add a title screen to a basic game. The same techniques will work for most simple games, so why not try adding a title screen to your own games, too?

>STEP-01

Write your game

We recommend you try adding a title screen to our example game Cat Catcher before you add one to your own game. To make Cat Catcher, first bring in the sprite Gravity Marble from the Things folder. It comes with some scripts for controlling it with the cursor keys. Add **Listing 1** to your cat sprite. Together, these two sprites make a game where you're challenged to see how quickly you can catch the cat ten times with the marble. We've added the playing field background.

Add an animated sprite to your title screen and use 'say' blocks to tell players how it works Black text on hot pink: a timeless background design!



>STEP-02

Create your title screen background

Create a new background image that you will be using for your game's title screen. Ours is just a bright colour with the game title on it, but you could make something more elaborate if you like. On the background, add the scripts shown in **Listing 2**. They change the background between the title screen and the in-game background, and tell all the sprites to go into 'title screen' mode when the green flag is clicked. Ultimately, this should be the only time you use a **when green flag clicked** script.

>STEP-03

Create your title screen sprite

This is the sprite that will tell the player how to play, and it can be animated, too. For our game, we've brought in another cat sprite. Add **Listing 3** to it. There are three parts to this: one part displays the title animation and instructions; another part starts the game when the sprite is clicked; and a third part hides the sprite when the game begins. You'll need to make a variable called **game status**, which all sprites will use

TITLE SCREENS IN SCRATCH

to tell whether the game is running or the title screen is on. You can add more sprites to your title screen. Include the when I receive play game script from Listing 3 to hide them when the game begins. Use a when I receive title screen script to show them on the title screen.

>STEP-04

Replace your green flag scripts

Now, you need to go through your game sprites (the game cat and the marble in our example) and change their scripts so they don't start when the green flag is clicked any more. For each sprite and each of its scripts, replace the block when green flag clicked with the block when I receive play game. Add

A great game starts with a title screen that draws players in

Listing 4 to your game sprites to make them hide when the title screen is on, and appear when the game begins. If a sprite shouldn't be there at the start of the game, you can leave out the **show** script.

>STEP-05

Replace the forever loops

Some of your in-game sprites might have **forever** loops. These will keep running, even when the title screen is showing and the sprite is hidden. To avoid this causing unwanted results, replace the **forever** block on your in-game sprites with the **forever if** block. Give the block the condition **game status = game** (using your variable **game status** and the **= Operator** block). You might also have events that are triggered, such as when there is a key press. To stop these working on the title screen, wrap an **if** block around the entire script after the **when [space] key pressed** block and give it the condition **game status = game**, too.

>STEP-06

Start a new game

When your game finishes, you can show the title screen again by adding a Control block to broadcast **title screen**. For example, you could add it to the end of **Listing 1** in our game. Players can once again start a new game from the title screen. That will keep them in the game and encourage them to keep playing until they've got a score they can brag about! You might need to make some other tweaks for your game (each one is different, after all), but following these steps should enable you to add a title screen to most simple games, to make them look more polished.

```
LISTING 1
when / clicked
reset timer
  go to x: (pick random -200) to 200) y: (pick random -150) to 150
  wait until touching gravity_marble v ?
set time v to timer
go to x: 0 y: 0
say It took you for 2 secs
 ay join time seconds for 2 secs
                                                           LISTING 2
when I receive play game v
switch to background playing-field
when I receive title screen v
switch to background title screen 🔻
when 🎮 clicked
set game status v to title
                                                           LISTING 3
when I receive title screen v
point in direction (75 v
go to x: -90 y: -90
set size to 350 %
say How fast can you catch me ten times? for 2 secs
say Use the cursor keys to move the ball for 2 secs
say Click me to play for 2 secs
 repeat until not (game status) = title
  turn (15) degrees
     ait (0.15) secs
  turn (15) degrees
   wait (0.15) secs
set game status v to game
broadcast play game v
 when I receive play game v
hide
                                                          LISTING 4
when I receive play game v
when I receive title screen v
```