Introducing Scratch

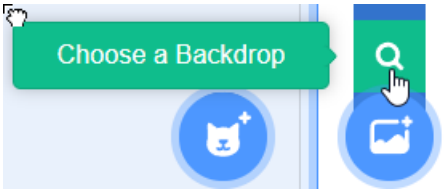
8 – Soccer

Time to get sporty. For this exercise, we'll make a 2 player game of Soccer / Football. It will feature 2 players. One moved using the WASD keys and the other moved using the ↑←↓→ keys. Each end of the field will have an area that the ball can be kicked in to for a goal. The field will have an edge so that the ball can go "out of bounds". We will also let players choose how long each game will last for.

Exercise 1. Create the Background

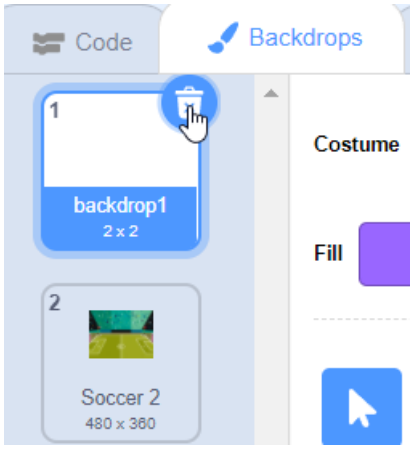
1. Start by removing the default Cat sprite.

2. From the sprites area choose a backdrop.



3. Choose the **Soccer 2** backdrop from the **Sports** category.

4. If you like, you can remove the default blank backdrop.



Exercise 2. Create the First Player

1. Now to import our first sprite. Import the sprite called **Ben** from the People category.

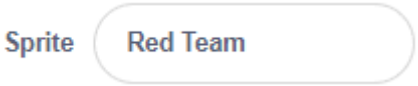
We've chosen this sprite because it has 4 costumes that we will be using.

Player moving

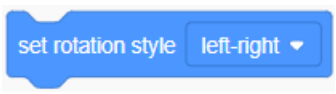
Kicking the ball

Celebrating a goal

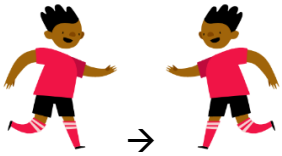
2. Rename the sprite as **Red Team**.



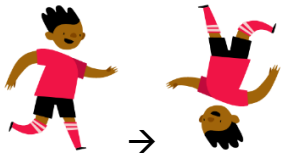
When our player is moving sometimes, they will be facing left and sometimes they will be facing right. Normally when a sprite is turning the other way it will rotate so that it ends up upside down. Instead, we want the sprite to simply face either left or right without rotating. Using the Set Rotation Style block makes this possible.



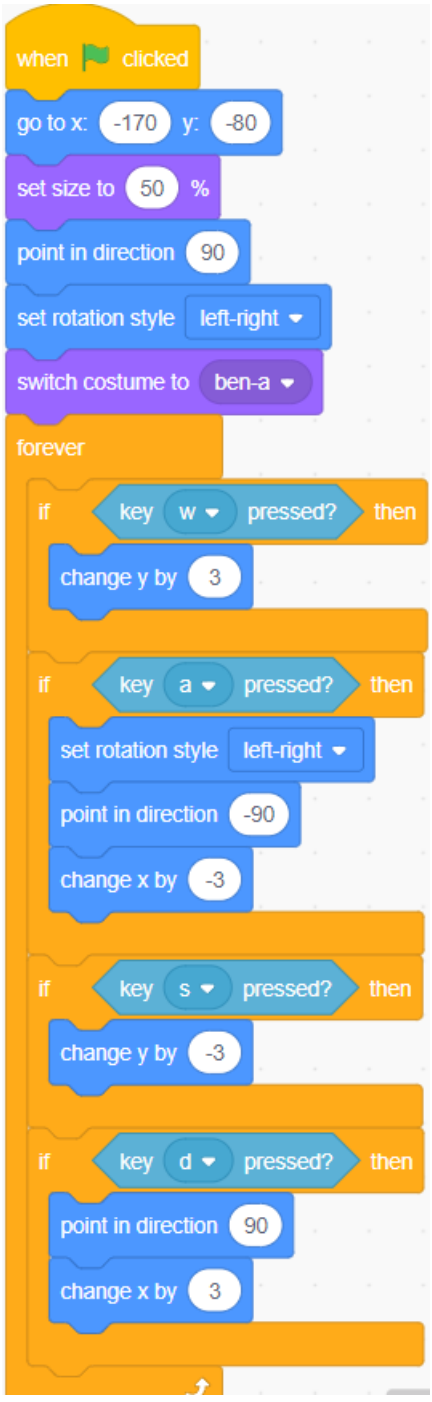
We want this



Not this



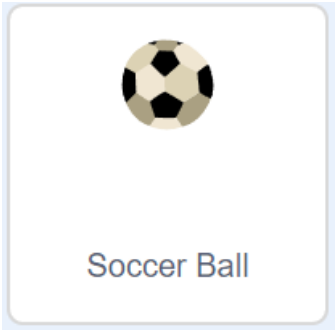
3. Add the following code blocks to set up the starting state for the sprite.



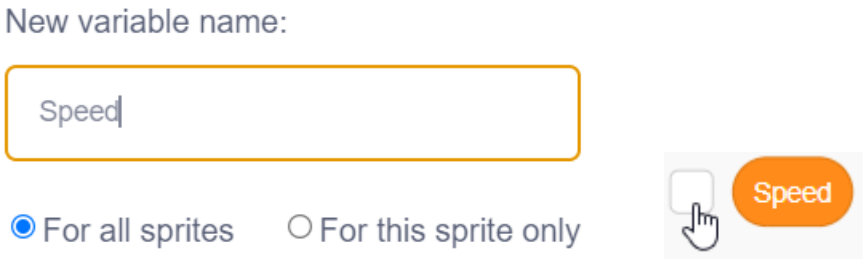
4. Test the game. Click the **Go** button and then test the movement keys.
W to move up.
A to move left.
S to move down.
D to move right.

Exercise 3. Add the Ball

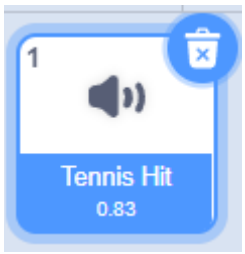
1. Import the **Soccer Ball** sprite from the **Sports** category.



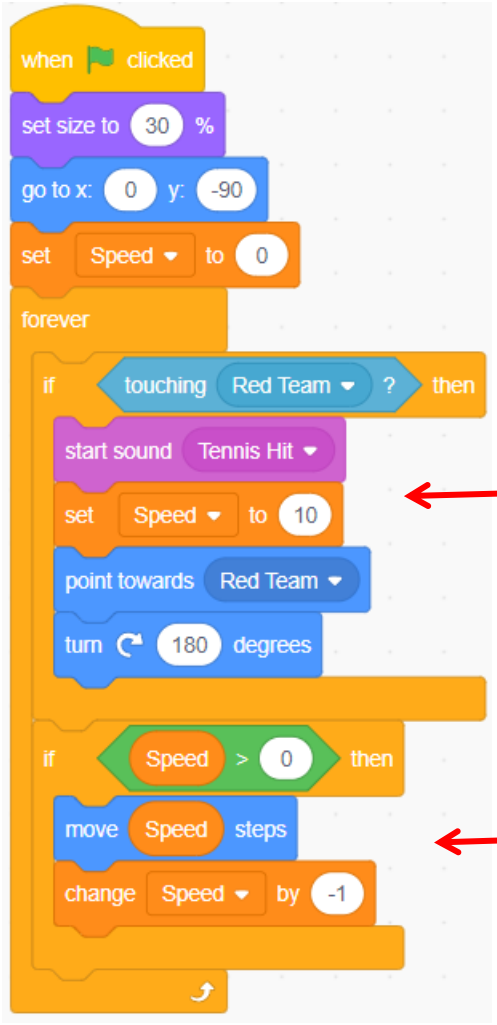
Create a variable called Speed and make sure it is not showing on the stage. This will be used to make the ball move when it has been kicked and then slow down until it is no longer moving.



2. Import a sound that sounds like a ball being hit (such as the **Tennis Hit** sound in the Sports category).



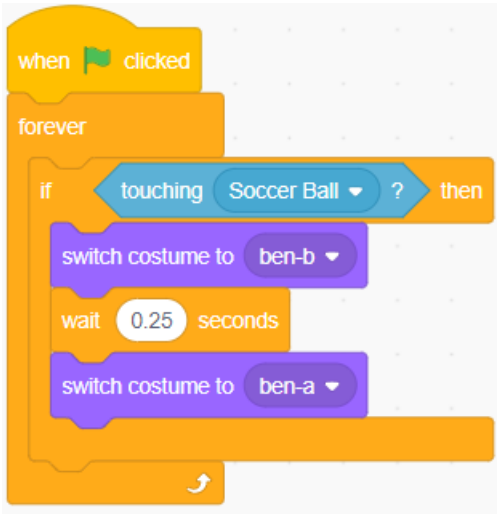
3. Add the following code blocks.



When the red player "kicks" the ball the speed variable is set, then the ball turns so that it is directed away from the red player.

While the speed variable is more than 0, the ball will move while slowing down until the speed is no longer above 0.

4. Add the following blocks to the **Red Team** sprite.



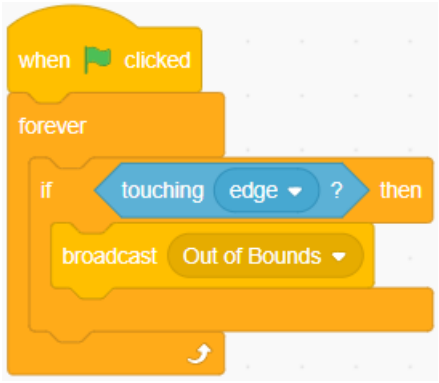
You could put this IF block inside the existing forever block if you prefer.

5. Test the game by moving the players so that they "kick" the ball.

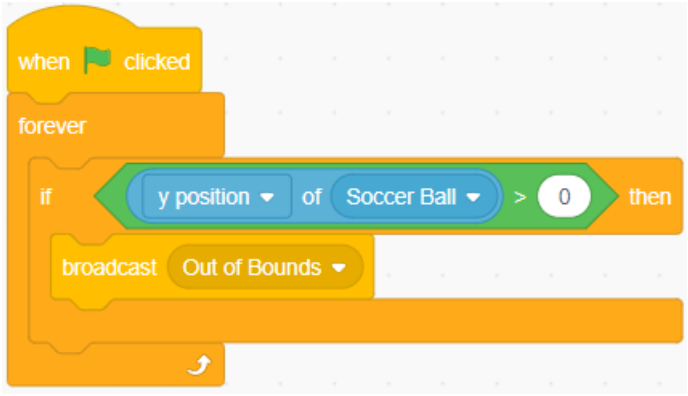
Exercise 4. Out of Bounds

If the ball touches the edge of the stage is will be considered out of bounds. However, the soccer field in our backdrop doesn't go all the way to the top of the stage. So the ball will also be considered out of bounds if it goes further up the stage than where the field in the backdrop goes up to. Since the edge of the field is in the middle of the screen, that position will be 0 on the Y axis. So if the ball goes above 0 on the Y axis, it will be out of bounds.

1. Add the following code blocks inside the **Soccer Ball** sprite.

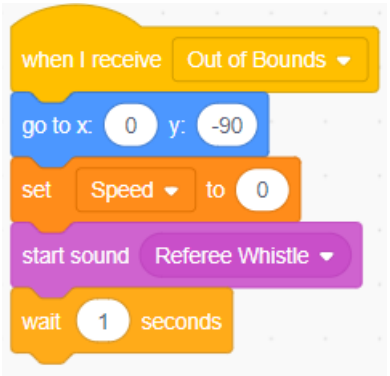


2. Add the following code blocks inside the **Backdrop**.



We've set out 2 out of bounds conditions. When the ball is out of bounds, play will briefly stop and a referee whistle sound will play as the ball returns to the centre of the field.

3. Select the Soccer Ball sprite and import the **Referee Whistle** sound from the **Sports** category.
4. Add the following code blocks to the **Soccer Ball** sprite.



Exercise 5. Adding the Green Team

It's not much of a soccer game without an opposing team. Much of the code for the second player will be similar to the player we have already done. Remember that you can copy code from one sprite to another by dragging code blocks on to another sprite. Just be careful to change all the parts that are meant to be different if you do it this way.

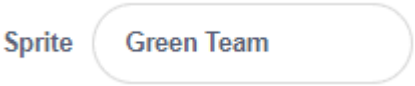
1. Import the sprite called **Jordyn** from the **People** category in the sprite library. This sprite has 4 costumes that we will use.

Player moving

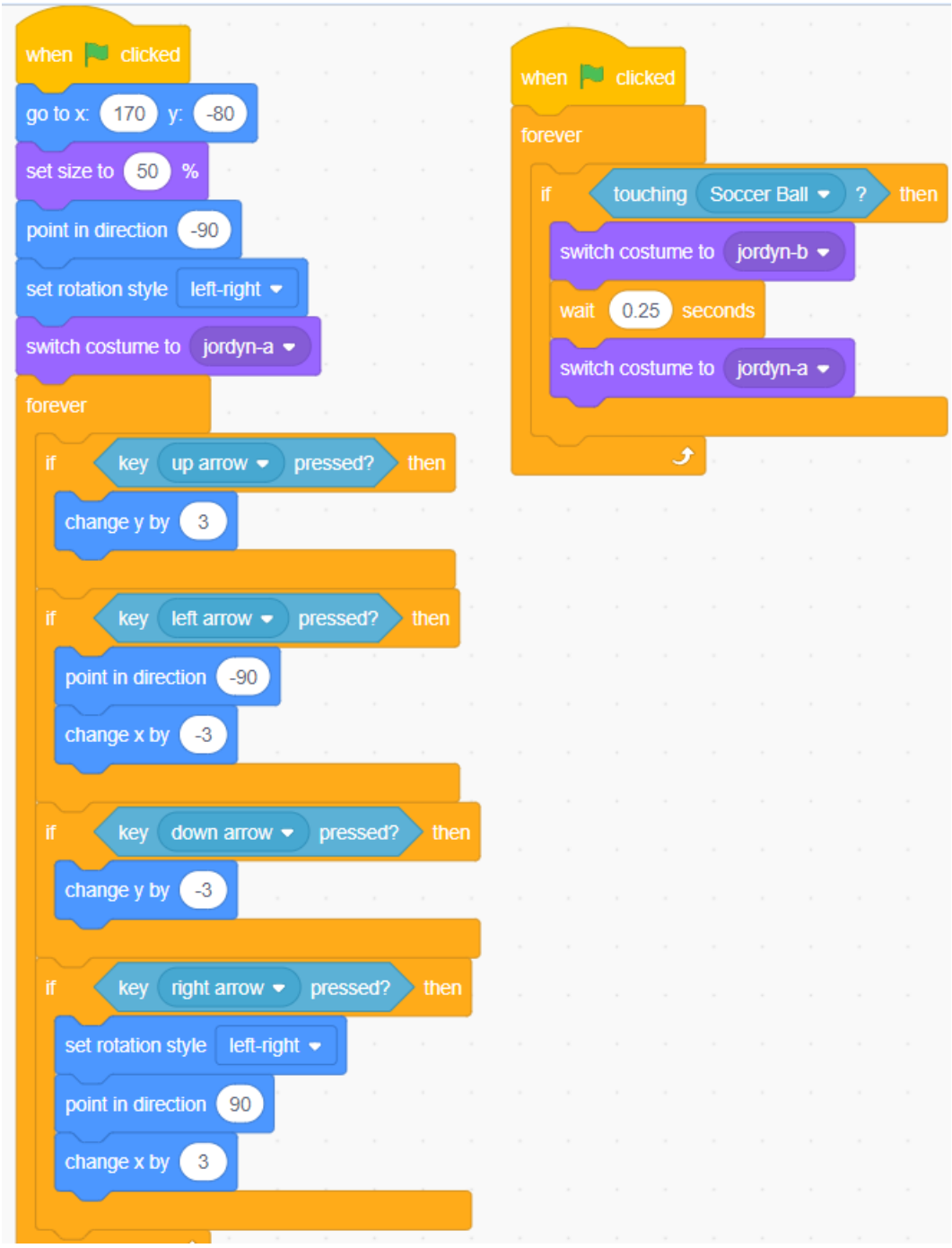
Kicking the ball

Celebrating a goal

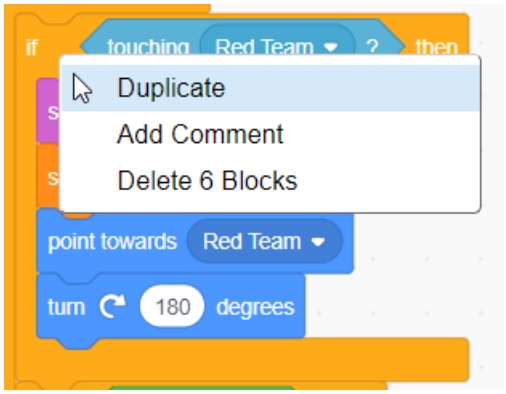
2. Rename the sprite as **Green Team**.



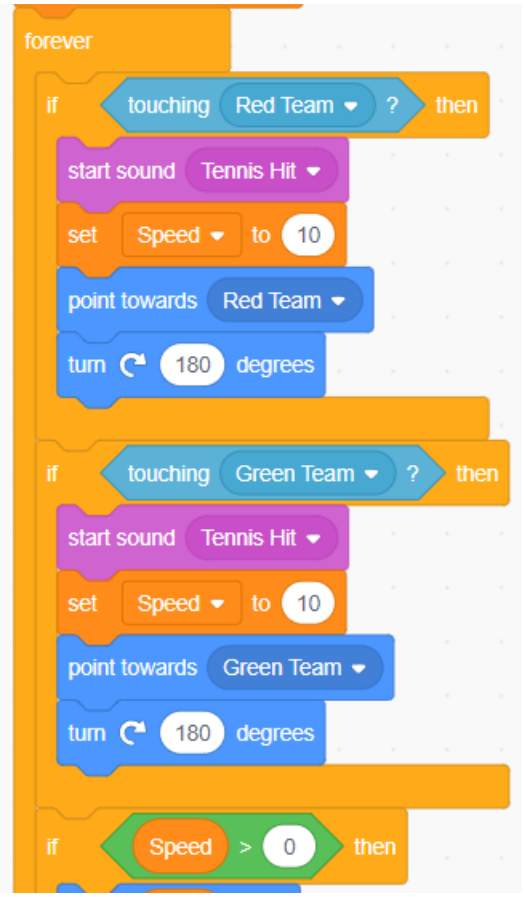
3. Add the following code blocks. These are similar to the Red Team code blocks but keep an eye out for the differences.



4. Select the **Soccer Ball** sprite.
5. **Right-click** on the section of code that handles the ball touching the red team player and select Duplicate. If more code than you need is copied, you can move the extra bits to the code bank on the left to remove it.



6. Place a copy of that same code underneath and change each Red Team reference to Green Team.

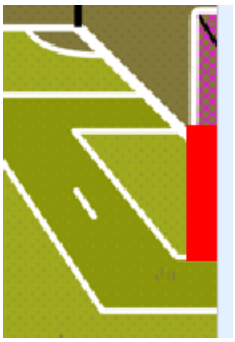


7. Test the game so far. Now both players should be able to move and kick the ball.

Exercise 6. Goal!!!!!!

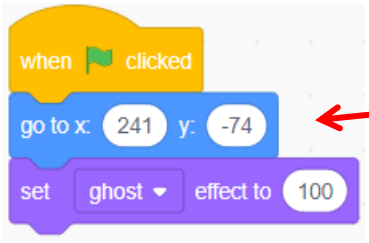
Each goal will have an invisible sprite sitting in front of it so that the ball can detect when it is in the goal area. Once the ball touches one of the goals it will broadcast that a goal has been scored at that end. A score variable will be increased, the ball will return to the centre, and the player on the scoring team do a little victory animation.

1. Create a new sprite that is a rectangle shape. Since it won't be visible the colour won't be important, but we'll make it red just to show that it's the red team goal.
2. Position it on the stage so that it is over the red team's goal and resize it in the sprite drawing area so that it is just the right size to cover the goal.


3. Rename the sprite to **Red Goal**.

Add the following code blocks to the Red Goal sprite.

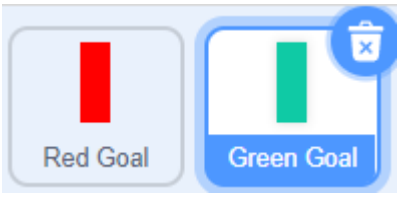


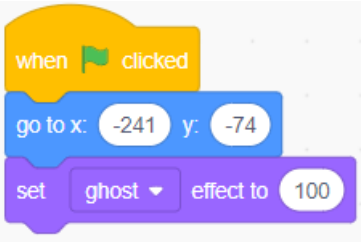
These coordinates should position the sprite right over the goal area

Note You could use a 

to make the sprite invisible but that also makes it impossible for other sprites to touch it. Instead, we've used a ghost effect to make sure it is still there, just not visible.

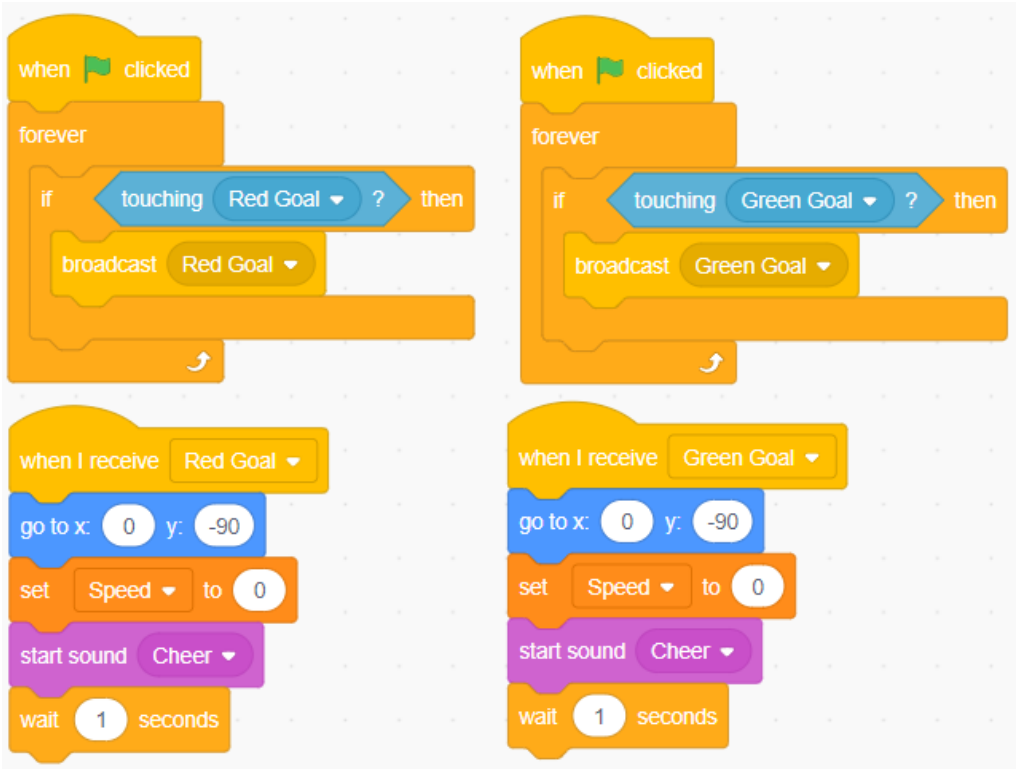
4. Create a new sprite (or duplicate and modify the one you just made) so that there is a **Green Goal** at the other end of the field.



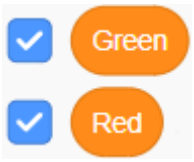


5. Import the **Cheer** sound from the **Sports** category of the Scratch library to the **Soccer Ball** sprite.

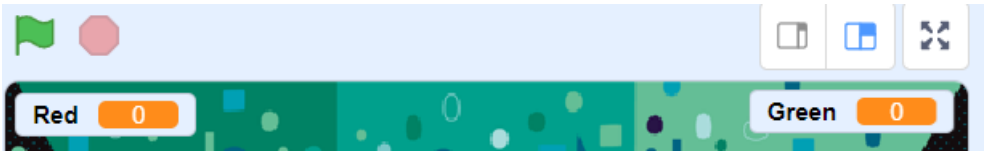
6. Add the following code blocks to the Soccer Ball sprite.



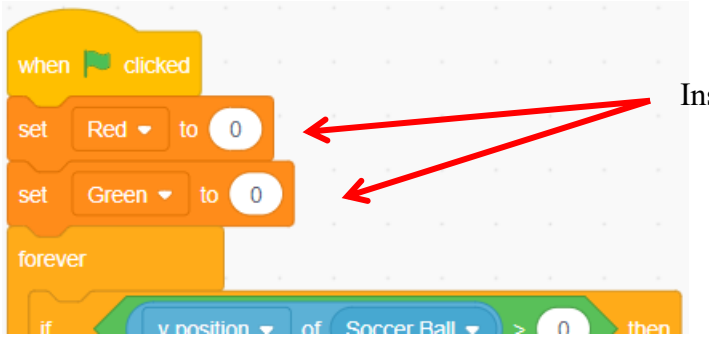
7. Select the **Backdrop** from the sprites area.
8. Create 2 new variables **Red** and **Green**.



9. Position them on the corners of the stage.

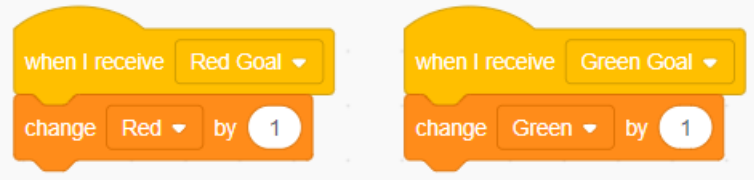


10. Add the following code blocks in the **Backdrop**.



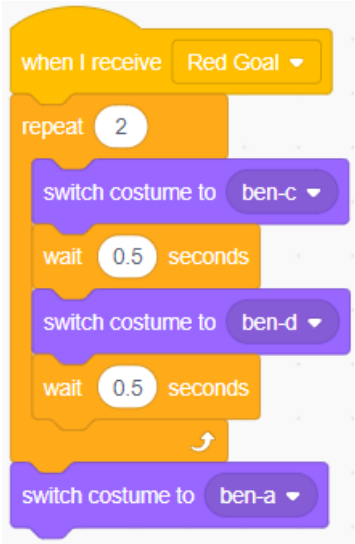
Insert into your existing blocks

11. Add these blocks to the **backdrop** also.

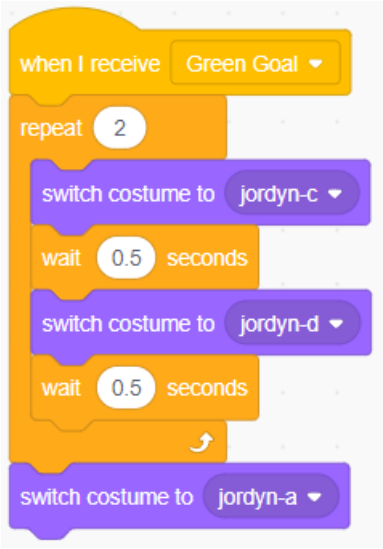


12. Add the following blocks.

Red Team sprite



Green Team sprite



13. Test your game and make sure both teams can score a goal properly. Check the score for the correct team updates as well as other goal conditions (such as the player's victory dance, crowd cheer sound and returning the ball to the centre).

Exercise 7. Ending the Game

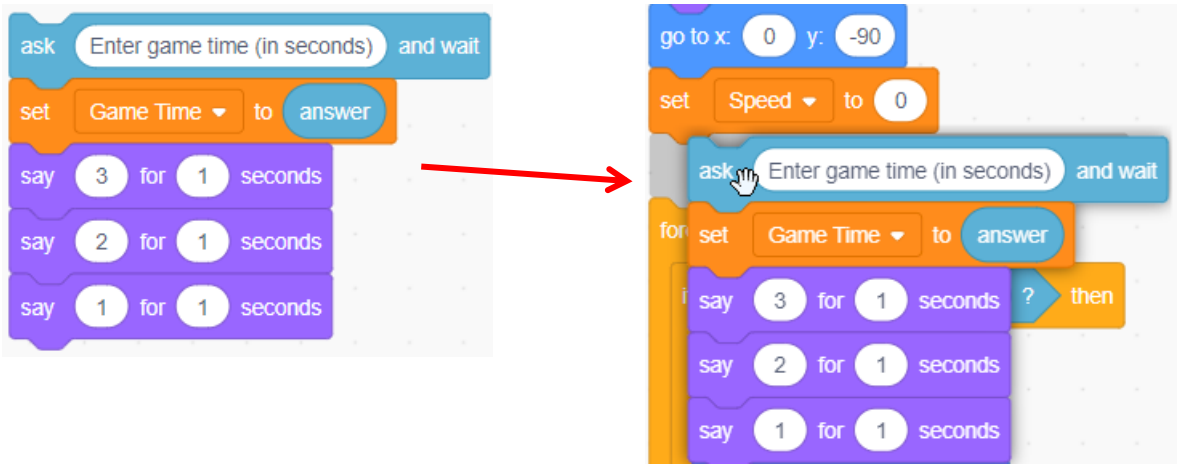
We will add a game timer that will count down the seconds until the game ends. We will also add an option for players to choose how long they want to game to last for before they start.

1. Make sure you are on the **Backdrop** and 2 new variables called add a new variable called **Game Time** and **Remaining**. Set the **Game Time** variable to **hidden**.

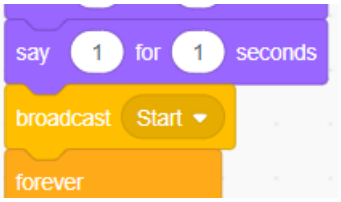
2. Position the **Remaining** variable at the top of the stage in the centre.
3. Right click on it and change it to **Large Readout**.

Now, we'll add some code blocks that will ask a player how long the game will go for by using the **Ask** and **Answer** blocks.

4. Add the following blocks to the Soccer Ball sprite.
5. Put them into the main section of code blocks.

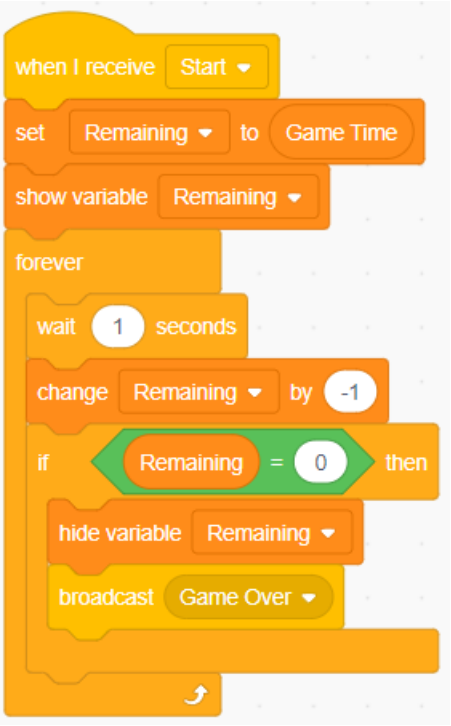


6. Place a **Start** broadcast under the blocks you just added.



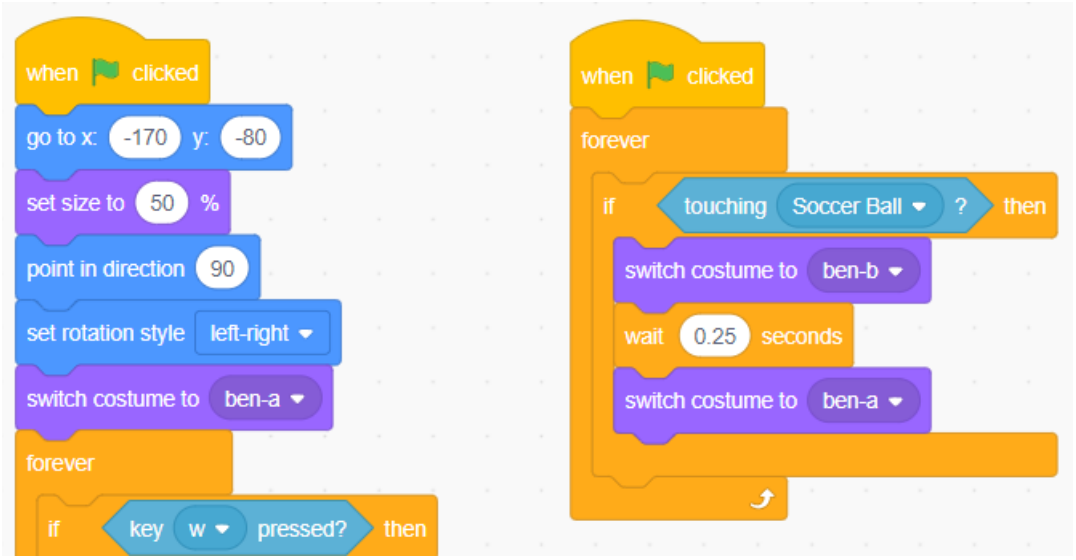
This broadcast will be used to start the timer. We will also modify the code for both players so that they start when they receive this broadcast instead of starting as soon as the go button is clicked.

7. Add the following code to the **Backdrop**.

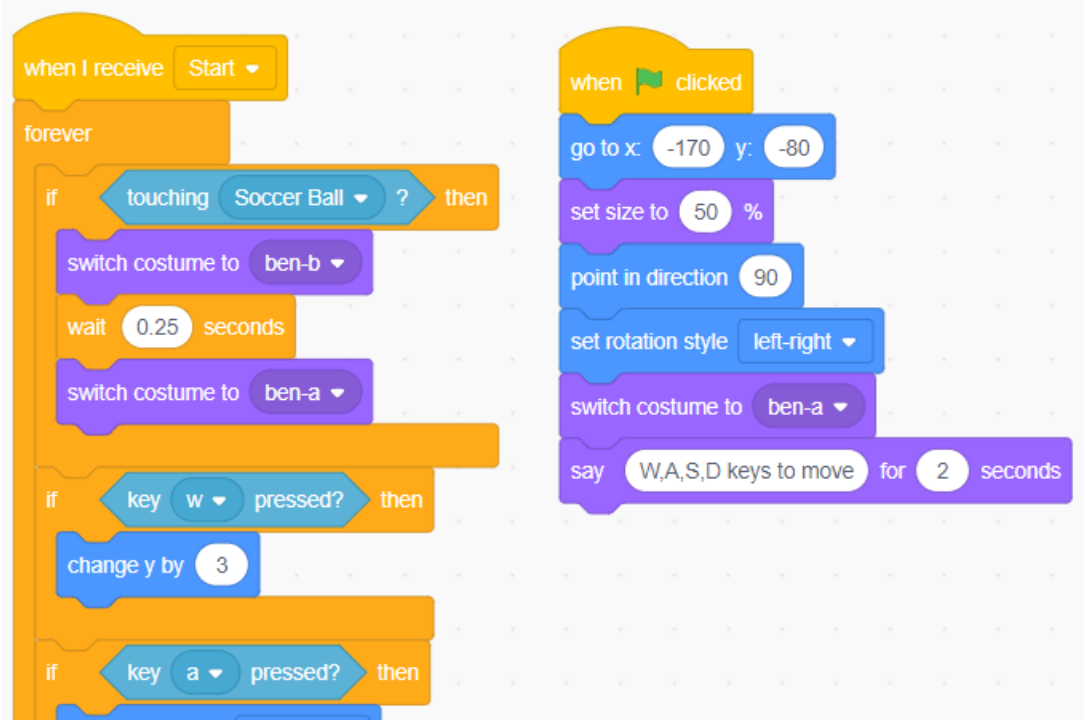


8. Add the blocks circled below into your **Go** button clicked code block.

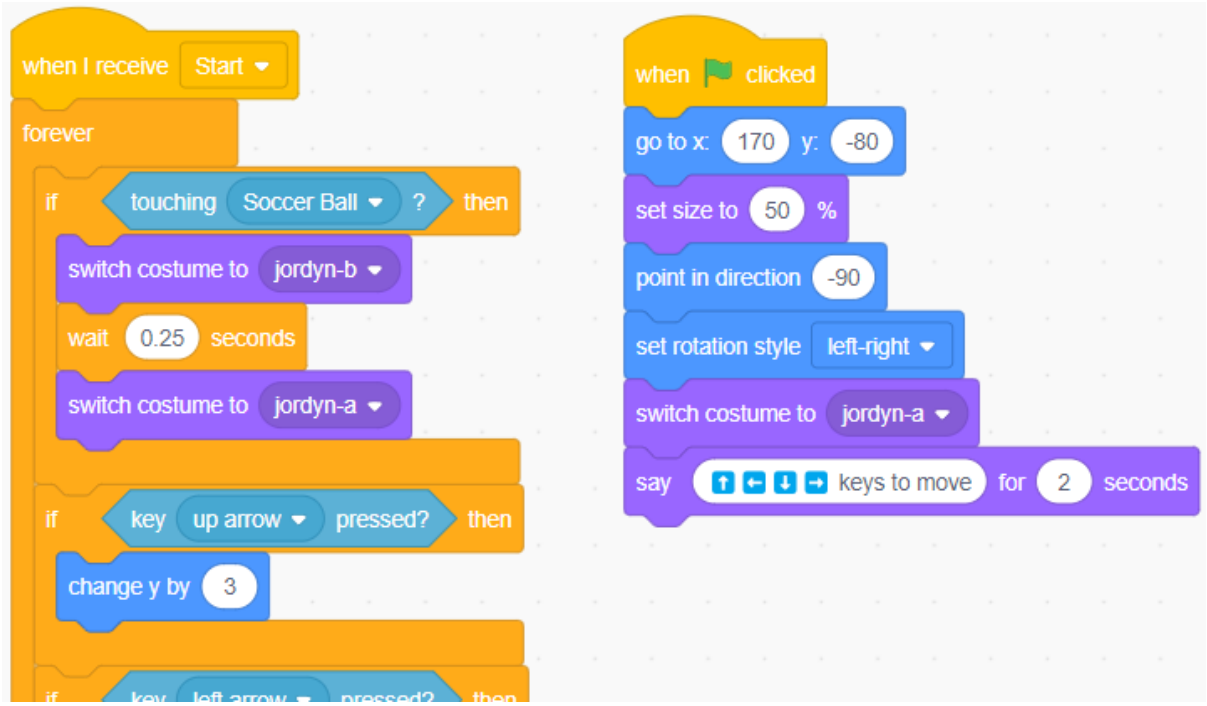
9. In the **Red Team** sprite, modify the code so that the blocks for the movement keys only begin when the Start Game message is received. We'll also add a small instruction message.



Change To

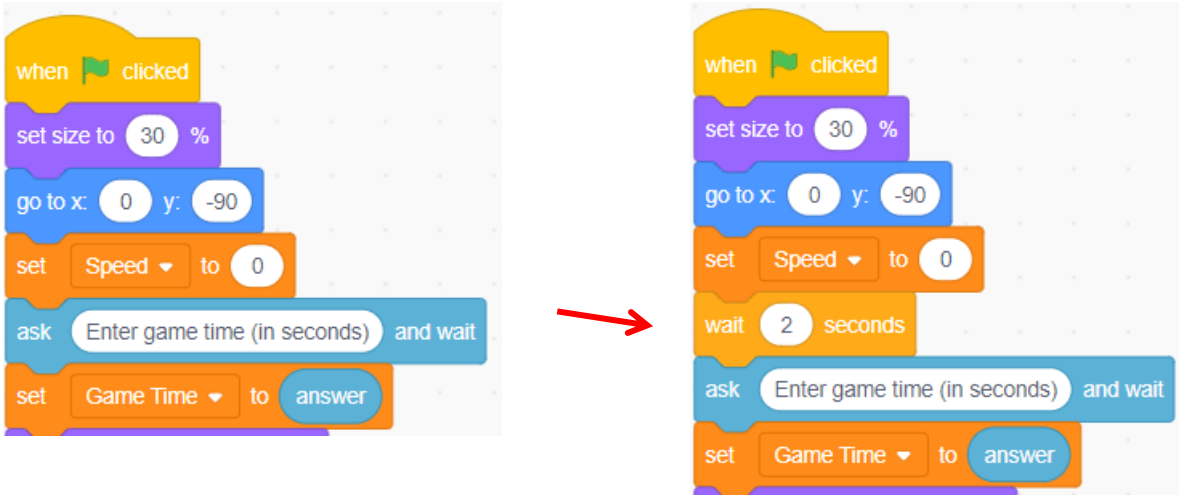


10. Make similar changes to the **Green Team** sprite.



Tip On Windows you can hold down the ⊞ key and press **.** (full stop or period key) to open the emoji menu and search for arrow symbols to insert.

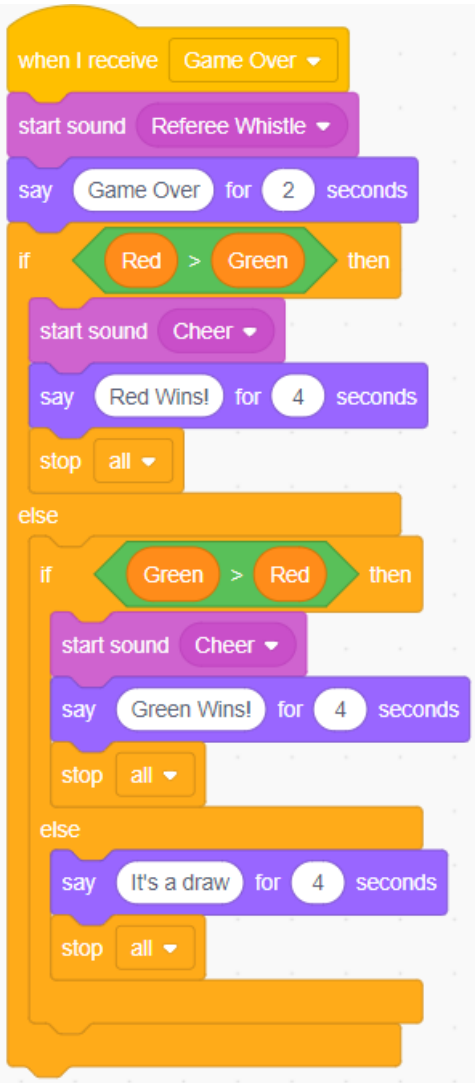
11. Since the game will be starting with both player sprites showing instructions for 2 seconds, we'll add a 2 second pause to the code blocks in the **Soccer Ball** sprite while that is happening. Prompting the player for game time and then starting the game will happen after that.



Exercise 8. Game Over

We already have a Game Over broadcast when the game time remaining reaches 0. Now we just need to add code blocks that will say what happens when the game ends.

1. Add the following code blocks to the **Soccer Ball** sprite.



Now to give it a good test.

What other features or changes can you add to make this game even better?