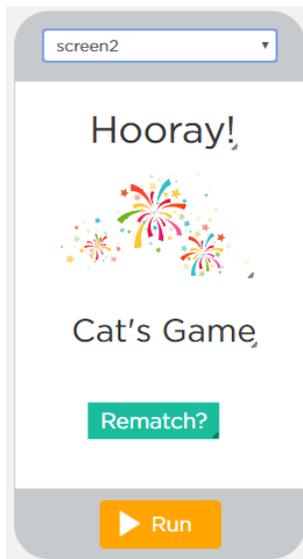


B6 Tic-Tac-Toe



Widget	Description
	I made my buttons 70 x 70. It is very important that you name them as shown.
	Label; id = whoseTurn



Widget	Description
	Label; id = winner
	Button; id = rematch

You need a variable (outside all the blocks) to track the turn.

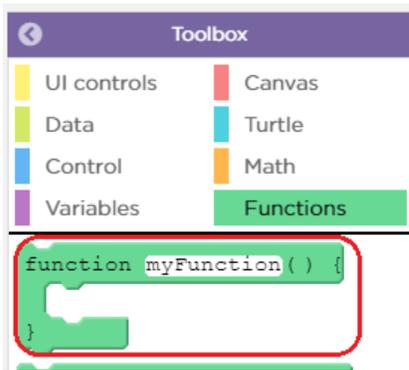
```
var turn = "X";
```

Start button a's onEvent off like this:

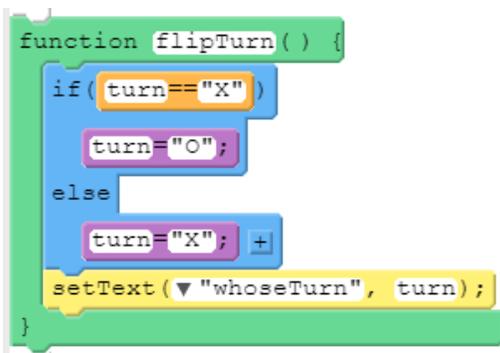
```
onEvent (▼"a", ▼"click", function(event) {
  if (getText(▼"a") == "?") {
    setText(▼"a", turn);
  }
});
```

If the button doesn't have a piece in it yet (it == "?"), then we place whatever turn is inside it.

Now, we need to switch the turn. However, that is going to be a common thing that I want to do EVERYWHERE. I don't want to cut and paste it each time because that will make my onEvent blocks big. Instead, I am going to make something called a function to do the work for me.

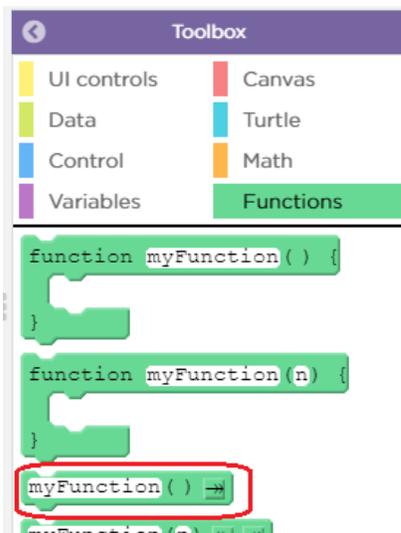


Then, add the blocks to flip the turn inside it.



Watch the quotation marks. Make sure that X and O are capitals.

Now, we can call this function inside a's onEvent and it will call the code for us.



Then rename it to match the function.

```

onEvent (▼"a", ▼"click", function(event) {
  if ( getText (▼"a") == "?" ) {
    setText (▼"a", turn);
    flipTurn ();
  }
});

```

OK... Now make one for all nine buttons. (a through i). Button b's onEvent will look like this:

```

onEvent (▼"b", ▼"click", function(event) {
  if ( getText (▼"b") == "?" ) {
    setText (▼"b", turn);
    flipTurn ();
  }
});

```

Switch in to text mode to do the cutting and pasting. It will be much quicker.

Winning Condition

Now we need to test the winning condition. We are going to make another function for this because it is very large and needs to be repeated many times.

```

function checkWinner (n) {

```

Switch over to text mode and add this if:

```

  if (getText("a")==getText("b") && getText("b")==getText("c") && getText("a")!= "?" ) {
    setText("winner", getText("a")+ " wins!");
    setScreen("screen2");
  }

```

The top row is what is being tested:

a	b	c
d	e	f
g	h	i

Go back to the buttons a through i. Add in a function to each. When a was complete, it looked like this:

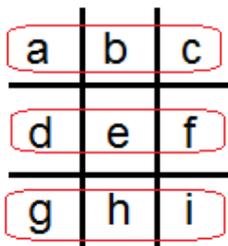
```

onEvent(▼"a", ▼"click", function(event) {
  if (getText(▼"a")=="?") {
    setText(▼"a", turn);
    flipTurn(); →
    checkWinner(); →
  }
});

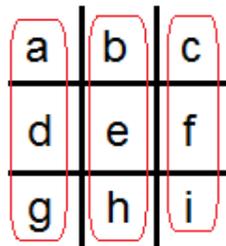
```

Test that it works. Make sure that there is a winner in the top row.

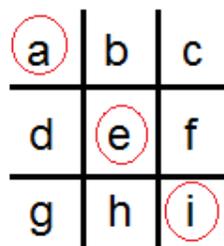
Now, you need to make sure all possible win conditions (all 8) are completed.



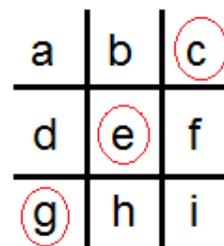
3 horizontal



3 vertical



down diagonal



up diagonal

Go back to the checkWin function and add if statements. Here I have added the second horizontal win to the original checkWinner function.

```

function checkWinner() {
  if (getText("a")==getText("b")&& getText("b")==getText("c") && getText("a")!= "?" ) {
    setText("winner", getText("a")+ " wins!");
    setScreen("screen2");
  }
  else if (getText("d")==getText("e")&& getText("e")==getText("f") && getText("d")!= "?" ) {
    setText("winner", getText("d")+ " wins!");
    setScreen("screen2");
  }
}

```

Verify that your other win methods run. Run your program.

The other situation that might occur is a cat's game (tie). In this case all pieces are filled and there is no winner. Add the else if to the bottom of your checkWinner function.

```

  else if(getText("a")!= "?" && getText("b")!= "?" && getText("c")!= "?" && getText("d")!= "?" &&
  getText("e")!= "?" && getText("f")!= "?" && getText("g")!= "?" && getText("h")!= "?" &&
  getText("i")!= "?") {
    setText("winner", "Cat's Game");
    setScreen("screen2");
  }
}

```

Take a look at this in the blocks editor. You really want to do this in text mode and NOT in blocks mode. Yikes.

Rematch Button

```
onEvent(▼"rematch", ▼"click", function(event) {  
  setText(▼"a", "?");  
  setText(▼"b", "?");  
  setText(▼"c", "?");  
  setText(▼"d", "?");  
  setText(▼"e", "?");  
  setText(▼"f", "?");  
  setText(▼"g", "?");  
  setText(▼"h", "?");  
  setText(▼"i", "?");  
  setScreen(▼"screen1");  
});
```

Set all of the squares (a through i) back to "?" and turn the screen back to the first screen.