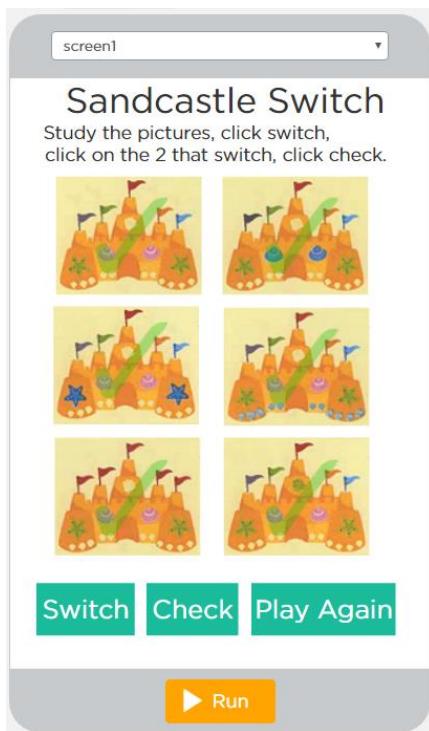
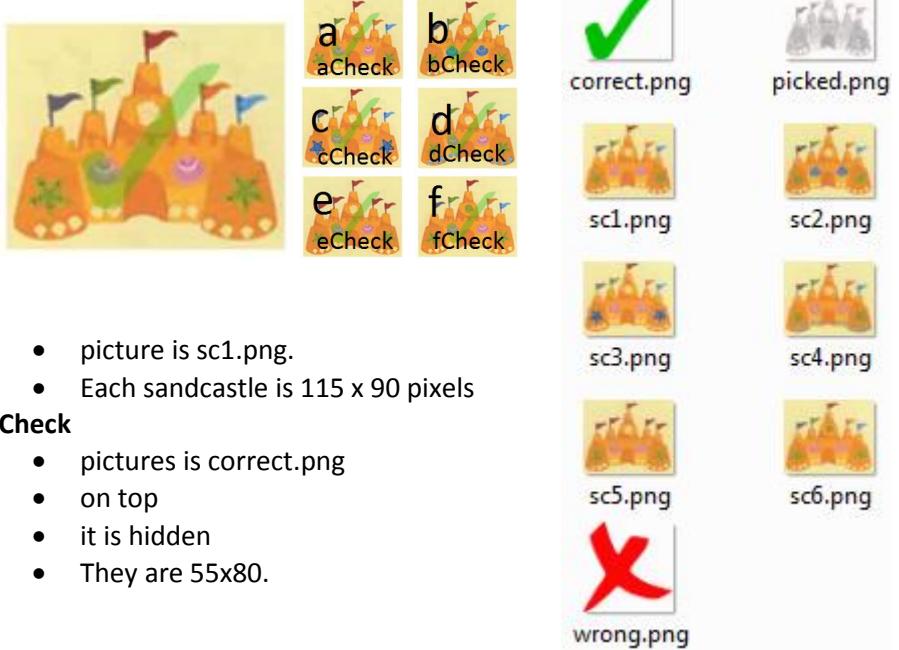


Sandcastle Switch



Each sandcastle picture is two pictures:



a

- picture is sc1.png.
- Each sandcastle is 115 x 90 pixels

aCheck

- pictures is correct.png
- on top
- it is hidden
- They are 55x80.

First, create these variables outside all of the onClicks.

```

1 var list = [1, 2, 3, 4, 5, 6];
2 var pic1 = 0;
3 var pic2 = 0;
4 var choice1 = 0;
5 var choice2 = 0;
```

Then, code the function MoveCastles, we will use it to display the values stored in the list on the screen:

```

function moveCastles() {
    setImageURL("a", "sc"+list[0]+".png");
    setImageURL("b", "sc"+list[1]+".png");
    setImageURL("c", "sc"+list[2]+".png");
    setImageURL("d", "sc"+list[3]+".png");
    setImageURL("e", "sc"+list[4]+".png");
    setImageURL("f", "sc"+list[5]+".png");
}
```

Here is one of the lines as text for you to cut and paste:

```
setImageURL("a", "sc"+list[0]+".png");
```

Then we can code the Switch button:

```
onEvent("switch", "click", function(event) {  
    pic1 = randomNumber(0, 5);  
    pic2 = randomNumber(0, 5);  
  
    while (pic1==pic2) {  
        pic2 = randomNumber(0, 5);  
    }  
    var swap = list[pic1];  
    list[pic1] = list[pic2];  
    list[pic2] = swap;  
    pic1++;  
    pic2++;  
    moveCastles();  
});
```

Here are the three swap lines so you can cut and paste them:

```
var swap = list[pic1];  
list[pic1]=list[pic2];  
list[pic2]=swap;
```

TEST YOUR CODE. Did two sand castles swap?

Now, we can code the images that the user clicks on to select their choices of the two they think switched.

```
onEvent("a", "click", function(event) {  
    setImageURL("a", "picked.png");  
    if(choice1==0)  
        choice1=1;  
    else  
        choice2=1;  
});
```

Here is b's code in text form:

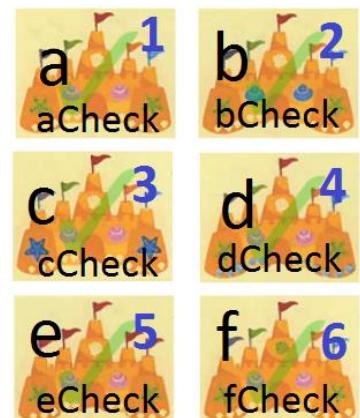
```
onEvent("b", "click", function(event) {  
    setImageURL("b", "picked.png");  
    if(choice1==0)  
        choice1=2;  
    else  
        choice2=2;  
});
```

You need to code all 6 buttons: a, b, c, d, e, f

TEST YOUR CODE. Can you click on two sandcastles?

The check button is next, it is quite complex because the marking piece is hard. One portion is cut and paste here, and the blocks for a + b follow. You need to do all 6.

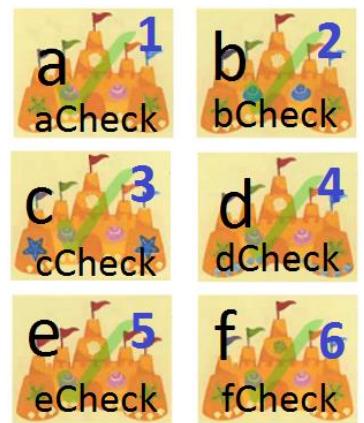
```
//for button a, position 1  
if(choice1==1 || choice2==1){  
    setImageURL("aCheck", "wrong.png");  
    showElement("aCheck");  
    if(pic1==1 || pic2==1)  
        setImageURL("aCheck", "correct.png");  
} else if(pic1==1 || pic2==1){  
    setImageURL("aCheck", "wrong.png");  
    showElement("aCheck");  
}
```



```

onEvent("check", "click", function(event) {
    //for button a, position 1
    if( choice1==1 || choice2==1 ) {
        setImageURL("aCheck", "wrong.png");
        showElement("aCheck");
        if( pic1==1 || pic2==1 )
            setImageURL("aCheck", "correct.png");
    } else if( pic1==1 || pic2==1 ) {
        setImageURL("aCheck", "wrong.png");
        showElement("aCheck");
    }
    //for button b, position 2
    if( choice1==2 || choice2==2 ) {
        setImageURL("bCheck", "wrong.png");
        showElement("bCheck");
        if( pic1==2 || pic2==2 )
            setImageURL("bCheck", "correct.png");
    } else if( pic1==2 || pic2==2 ) {
        setImageURL("bCheck", "wrong.png");
        showElement("bCheck");
    }
    //for button c
    if( choice1==3 || choice2==3 ) {
        setImageURL("cCheck", "wrong.png");
    }
}

```

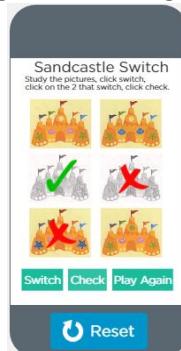


TEST YOUR CODE. Were your choices marked? Check all three cases shown below.

Both right:



One right, one wrong:



Both wrong:



The button again resets everything to hidden and back to the originals so that you can play again.

```
onEvent("again", "click", function(event) {
    hideElement("aCheck");
    hideElement("bCheck");
    hideElement("cCheck");
    hideElement("dCheck");
    hideElement("eCheck");
    hideElement("fCheck");

    list[0]=1;
    list[1]=2;
    list[2]=3;
    list[3]=4;
    list[4]=5;
    list[5]=6;
    choice1=0;
    choice2=0;
    moveCastles();
}
});
```

TEST your code. Everything should be working.