Project 41

DP EDUCATION

Coding School

41

Lucky LOTTERY

Make your own luck

PRESS

Built on Code Studio ▲

How It Works

T Here

See App

Built on Code Studio ▲

❖ First of all you need to create the screen as above. Go to the design toolbox for that. Here the playtime theme is used. Use the dropdown below to select a theme that suits you.

❖ Use the label tool in the design toolbox to display the "Lucky Lottery" header at the top of the screen. By changing its properties, you can create a beautiful title as above.

❖ Set its id as "label_h1", text property as "Lucky LOTTERY" and font size as 36. Change the x position and y position properties to match the values of the label to position it where you want it.

❖ Let's use the same label to create the second slogan "Make your own luck". There, the above properties should be changed and placed in the relevant place. Change the properties as shown in this image.

❖ To create the numbered spinner to rotate when the button is clicked, you must first add a numbered spinning gif using the image tool. For this, select the file "spinnerAnimated.gif" by the image property. It can be placed in the right place by changing its properties as follows. Change its id to "spinnerGif". Change its size and position as shown in the image below. Then tick the hidden property.
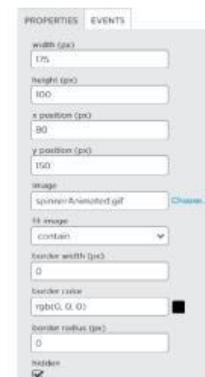
❖ Then use the image tool to create the image where the spinner is stopped below. Select the relevant image for its image property. Select the spinnerImg.jpg file as the image. Change the properties of that image as follows.

❖ Then let's create the display to show the winning number. For that, let's use the same label in the design toolbox. Change its id to "label_winning_number". Change its properties as shown in the image below.
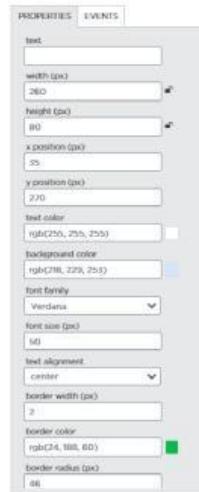
❖ After that, let's create the "PRESS" button. Give the value "button" as its id and change its properties as shown in the image below.

❖ Because the layout of the screen has been created, now let's turn to coding. For that you have to go to the code tab.

❖ First of all, the characters needed to be randomly auto generated should be saved in a variable. For that, you have to drag & drop the following block in the variables section of the toolbox. Delete the name x and type a suitable name in it. We are now storing English letters in this variable, so let's give it the name "characters". Enter the English alphabet letters in capital as its values.

```
1  var characters = "ABCDEFGHIJKLMNOPQRE";
```

❖ In the same way, create another variable and give it the name "numbers" and enter the numbers from 0-9 as its values..

❖ Use the code block below to change the color of the text on the screen where the winning number is displayed. For that, get the setProperty block from the UI controls section. For its "id", give the value "label_winning_number" which is the id of our display label. For the next value, select the "text-color" value from the dropdownd. Select a desired color from the last dropdown.

```
setProperty(▼ "label_winning_number",  ▼ "text-color",  ▼ "blue");
```

❖ Then let's code what happens when the button is clicked. For that, let's use the "on event" block in the UI controls. Give the value "button" which is the button id to its id. After that, the image where the spinner has stopped disappears and the spinning gif is shown, which should be clicked. Therefore, the two related blocks must be included inside the on event.

```
hideElement(▼ "spinnerImg");
showElement(▼ "spinnerGif");
```

❖ Then use the following codes to display a randomly generated character and 5 numbers on our display label. Use the playsound block to play a sound when the winning number is displayed and select a suitable sound. Since the winning number should be generated only once, when the button is clicked once, let's use the hide element block to hide the button.

```
var char = randomNumber(0, 51);
var num1 = randomNumber(0, 9);
var num2 = randomNumber(0, 9);
var num3 = randomNumber(0, 9);
var num4 = randomNumber(0, 9);
var num5 = randomNumber(0, 9);

var letter = characters.substring(char, char + 1);
var number1 = numbers.substring(num1, num1 + 1);
var number2 = numbers.substring(num2, num2 + 1);
var number3 = numbers.substring(num3, num3 + 1);
var number4 = numbers.substring(num4, num4 + 1);
var number5 = numbers.substring(num5, num5 + 1);
playSound(▼ "sound://category_bell/bells_win_high.mp3", ▼ false);
setText(▼ "label_winning_number", letter + number1 + number2 + number3 + number4 + number5);
hideElement(▼ "button");
```

Choose the correct answer.

1. Suppose we want to change the id of the label where the winning number is displayed. If so, what should be changed in the code section?

   hideElement,setProperty හා setText යන blocks

   setProperty හා setText යන block

   onEvent,showElement හා setProperty යන block

2. Which answer is correctly explained by these two code blocks?

   ```
   hideElement(▼"spinnerImg");
   showElement(▼"spinnerGif");
   ```

   An element with a gif called spinnerGif will start to appear.

   The spinnerImg element disappears and the spinnerGif element appears.

   Starting to see the image of the spinner that has stopped spinning.

3. Which block is suitable for displaying one randomly generated letter and 5 numbers?

   ```
   setText(▼"label_winning_number", letter + (number1 + (number2 + (number3 + (number4 + ◯)))));
   ```

   ```
   setText(▼"label_winning_number", letter + (number1 + (number2 + (number3 + (number4 + number5)))));
   ```

   ```
   setText(▼"label_winning_number", number5 + (number1 + (number2 + (number3 + (number4 + letter)))));
   ```

4. Imagine you want to display the image of the spinner that has stopped instead of the spinner that spins again after a second. What are the required code blocks for that?

```
setTimeout( function() {
    showElement( ▼ "spinnerImg");
}               , 1000);
```

```
setTimeout( function() {
    showElement( ▼ "spinnerGif");
}               , 1000);
```

```
setTimeout( function() {
    showElement( ▼ "spinnerImg");
}               , 2000);
```

5. Which block should be removed to get repeated winning number at one time?

```
hideElement( ▼ "button");
```

```
hideElement( ▼ "spinnerImg");
```

```
showElement( ▼ "button");
```