CODAPPS
Making the cookie click

Clément Levallois

Version 1.0, last modified

Table of Contents

1. Recap of the app so far

Before we start, it is good to have an overview of where we stand so far:

1. We created a class called `Form1`, this is the screen of the app.

2. In it, we have created a special method called `Form1` (same name as the class). This method will execute when the screen is instantiated

3. In this method, we have created variables for our images.

4. We have also created containers and given them a layout. Our screen is organized in three regions.

5. We have created Labels, ScaleImageButton and MultiButtons and added them to the screen.

6. The interface is ready, it should look like:

In the preview, <u>clicking on the cookie</u> should show a bright star. Try it!

In the preview, you can <u>scroll down with your mouse</u> in this region of the screen to see the other tools available. Try it!
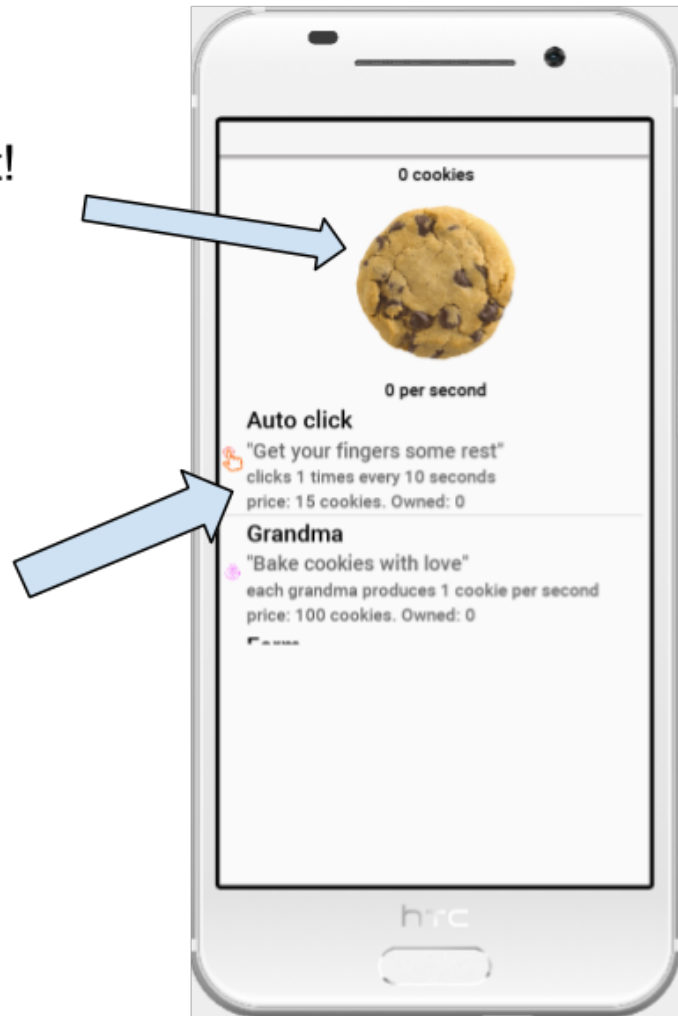
Figure 1. How your app should look like so far

If you are lost or unsure about the code you have written so far, here is a file with the complete code. You should of course place it in a file named `Form1` (to create this file, follow these instructions).

Now we need that the app "do something" when the player clicks on different parts of the screen. This is what we are going to write now.

2. Increasing the score when the cookie is clicked

First, we need a variable to store the value of the score. We will put this variable near the top of our file:

Adding a variable to keep track of the score

```
package net.clementlevallois.net.cookieclicker.ultimate;

public class Form1 extends com.codename1.ui.Form {

    Resources theme = Resources.getGlobalResources();
    float score = 0f; ①


}
```

① we create a new variable. The `f` after the zero is to indicate the number is a decimal, see this lesson.

Then we add lines of code **in the constructor method `public Form1 () {` where we have written everything so far**:

Adding an action listener to the cookie ScaleImageButton

```
centerRegion.add(autoclickButton); ①
centerRegion.add(grandmaButton); ①
centerRegion.add(farmButton); ①

cookieCentral.addActionListener(   ②
              (event) -> { }   ③
        ); ④
```

① The last lines we have written in the previous lesson.

② we use the method `addActionListener` for the variable `cookieCentral` (which is the picture of our cookie).

③ Anything written between the curly braces `{ }` will be executed when the user touches the cookie on sreen!

④ this parenthesis closes the `addActionListener` method, and the `;` ends the line.

The lines of code above are typically **hard to understand the first time**, because there are brackets, curly braces and semicolons to put at the right place.

You don't need to memorize everything but should still take some time to understand the logic, instead of copying this code without thoughts.

These lines of code **do nothing yet**, because we haven't written anything between the curly braces in the line `(event)  →  { }`.

Let's had lines of code which will increase the score by one (+ 1) when the player hits the cookie:

The lines will be:

```
score = score + 1;
```

```
scoreLabel.setText(score + " cookies");
```

And they should be placed between the curly braces. What they do is simple:

```
score = score + 1;
```

→ the variable score gets incremented by one

```
scoreLabel.setText(score + " cookies");
```

→ the text of the Label showing the score on screen gets updated and will show the score followed by the word "cookies".

Let's place these 2 lines where they should be:

Adding an action listener to the cookie ScaleImageButton

```
cookieCentral.addActionListener(
            (event) -> { ①
              score = score + 1;
              scoreLabel.setText(score + " cookies");
            } ①
      );
```

① don't forget the opening and closing curly braces!

Now, when the user will touch / press the picture of the cookie, the score will increase by one and this new score will be visible on screen:

▶ https://www.youtube.com/watch?v=pCo-wKIDc0M *(YouTube video)*

Congratulations, your app has started coming alive!

In the next lesson, we will learn how to make the other buttons of the app "do something" when the player clicks on them.

The end

Questions? Want to open a discussion on this lesson? Visit the forum here (need a free Github account).

Find references for this lesson, and other lessons, here.

Licence: Creative Commons, Attribution 4.0 International (CC BY 4.0). You are free to:

• copy and redistribute the material in any medium or format

• Adapt — remix, transform, and build upon the material

⇒ for any purpose, even commercially.

This course is designed by Clement Levallois.

Discover my other courses in data / tech for business: http://www.clementlevallois.net

Or get in touch via Twitter: @seinecle