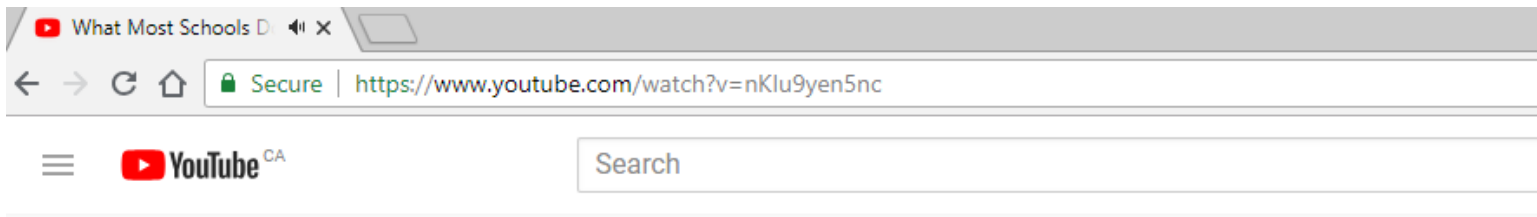


Introduction to AppLab

Widgets & Graphical User Interfaces



“Everybody in this country should learn how to program a computer... because it teaches you how to think.”
- Steve Jobs

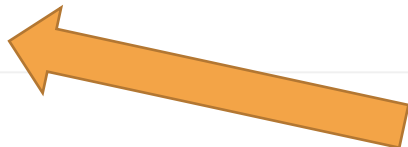
In 2013, the tech industry was concerned about the shortage of good programmers.

This video started circulating.

<https://www.youtube.com/watch?v=nKlu9yen5nc>

What Most Schools Don't Teach

14,415,893 views



What Most Schools Don't Teach - YouTube



<https://www.youtube.com/watch?v=nKlu9yen5nc>

Feb 26, 2013 - Uploaded by Code.org

Learn about a new "superpower" that isn't being taught in 90% of US schools. Starring Bill Gates, Mark ...



To train the needed programmers, many tech leaders have given money and time to get code.org's on-line courses running.



[My Dashboard](#)[Course Catalog](#)[Projects](#)[Professional Learning](#)

Hi Gorski ▾





Projects

Over 24 million projects created

[My Projects](#)[Public Projects](#)

Start a new project

[Play Lab](#)[Artist](#)[App Lab](#)[Game Lab](#)[▾ View full list](#)

	Name	Updated ▾	
	Sudoku	6/26/18 4:03 PM	Rename Publish Delete
	SomethingSomething	6/26/18 4:01 PM	Rename Publish Delete

Design Mode – setting up the widgets on the screen

The screenshot shows the 'Design Mode' interface for an app named 'Suduko'. The top teal header contains the app name 'Suduko' (saved 2 months ago) and buttons for 'Rename', 'Share', 'Remix', and 'Create New'. On the right are 'My Projects' and a user profile 'Hi Gorski'. Below the header, a tab bar shows 'Code' and 'Design' (highlighted with a red circle). The 'Design' view is split into three main sections:

- Left Panel (Preview):** Displays a 4x4 grid for the Sudoku game. The grid contains question marks and sea creature icons (pink fish, orange star, blue starfish, green octopus). Below the grid are 'Check' and 'Reset' buttons, and a large orange 'Run' button at the bottom.
- Design Toolbox:** A purple sidebar with the title 'Design Toolbox' and a gear icon. It contains the instruction 'Drag the elements into your app!' and a grid of 12 widget icons: Button, Text Input, Label (abc), Dropdown, Radio Button, Checkbox, Image, Canvas, Screen, Text Area, Chart, and Slider.
- Workspace:** A large area with a purple header 'Workspace' and a 'Version History' button. It contains the instruction 'Click on an element or choose it from the dropdown below to edit its properties. Create code in Events.' Below this is a 'PROPERTIES' tab (selected) and an 'EVENTS' tab. A dropdown menu shows 'a'. The properties list includes: id (a), width (px) (65), height (px) (65), x position (px) (12), y position (px) (94), image (b0.png with a 'Choose...' link), fit image (contain), and hidden (checkbox).

On the right side of the workspace, there are 'Delete' and 'Duplicate' buttons. The bottom of the screen has a footer with 'Privacy Policy | Copyright | More'.

Code Mode – setting up blocks for the program

The screenshot displays the Code Mode interface of a web-based IDE. The top navigation bar includes a logo with 'C O D E' and the project name 'Sudoku', along with buttons for 'Rename', 'Share', 'Remix', and 'Create New'. On the right, there are links for 'My Projects' and the user 'Hi Gorski'. Below the navigation bar, the 'Code' tab is selected and highlighted with a red circle. The interface is divided into three main sections: a preview window on the left, a toolbox in the middle, and a workspace on the right.

The preview window on the left shows a 4x4 grid for the Sudoku game. The grid contains question marks and sea creatures (a pink fish, a starfish, a blue starfish, and a green octopus). Below the grid are 'Check' and 'Reset' buttons, and a large orange 'Run' button at the bottom.

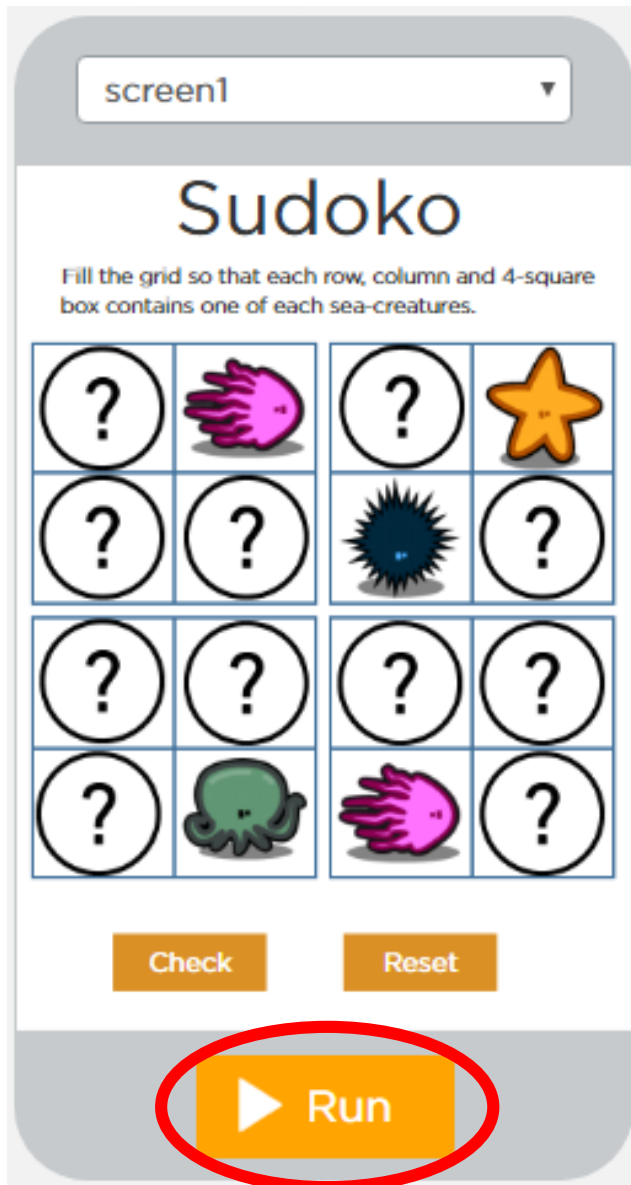
The toolbox in the middle lists various UI controls and functions, categorized by color: UI controls (yellow), Data (green), Control (blue), Variables (purple), Canvas (red), Turtle (cyan), Math (orange), and Functions (green). The functions listed include `onEvent(id, type, callback)`, `button(id, text)`, `textInput(id, text)`, `textLabel(id, text)`, `dropdown(id, option1, etc)`, `getText(id)`, `setText(id, text)`, `getNumber(id)`, `setNumber(id, number)`, `checkbox(id, checked)`, `radioButton(id, checked)`, and `getChecked(id)`.

The workspace on the right shows the code blocks for the program. It contains three event listener functions for buttons 'a', 'c', and 'e'. Each function increments a counter (a, c, or e) and updates the image URL of the corresponding button. The code blocks are color-coded to match the toolbox categories: yellow for UI controls, green for Data, blue for Control, and purple for Variables.

```
13 onEvent(▼ "a", ▼ "click", function(event) {
14   a++;
15   if(a>4){
16     a=1;
17   }
18   setImageURL(▼ "a", "b"+a+".png");
19 }
20 onEvent(▼ "c", ▼ "click", function(event) {
21   c++;
22   if(c>4){
23     c=1;
24   }
25   setImageURL(▼ "c", "b"+c+".png");
26 }
27 onEvent(▼ "e", ▼ "click", function(event) {
28   e++;
29   if(e>4){
```

The bottom of the interface features a 'Debug Console' section with a 'Show Debug Commands' button, a 'Clear' button, and a 'Watch' button.

Run Mode – seeing if your code works



Not Knots

Only one of these ropes would make a knot if you pulled both ends. Which one?

A



B



C



D



Answer

Click the above button to answer.

Built on Code Studio <

How many
widgets are on
this screen?

Which widget
types are on
this screen?

How many
widget types
are on this
screen?

Widget Types:

Button

Button

abc

Label



Radio Button



Image



Screen



Chart



Text Input



Dropdown



Checkbox



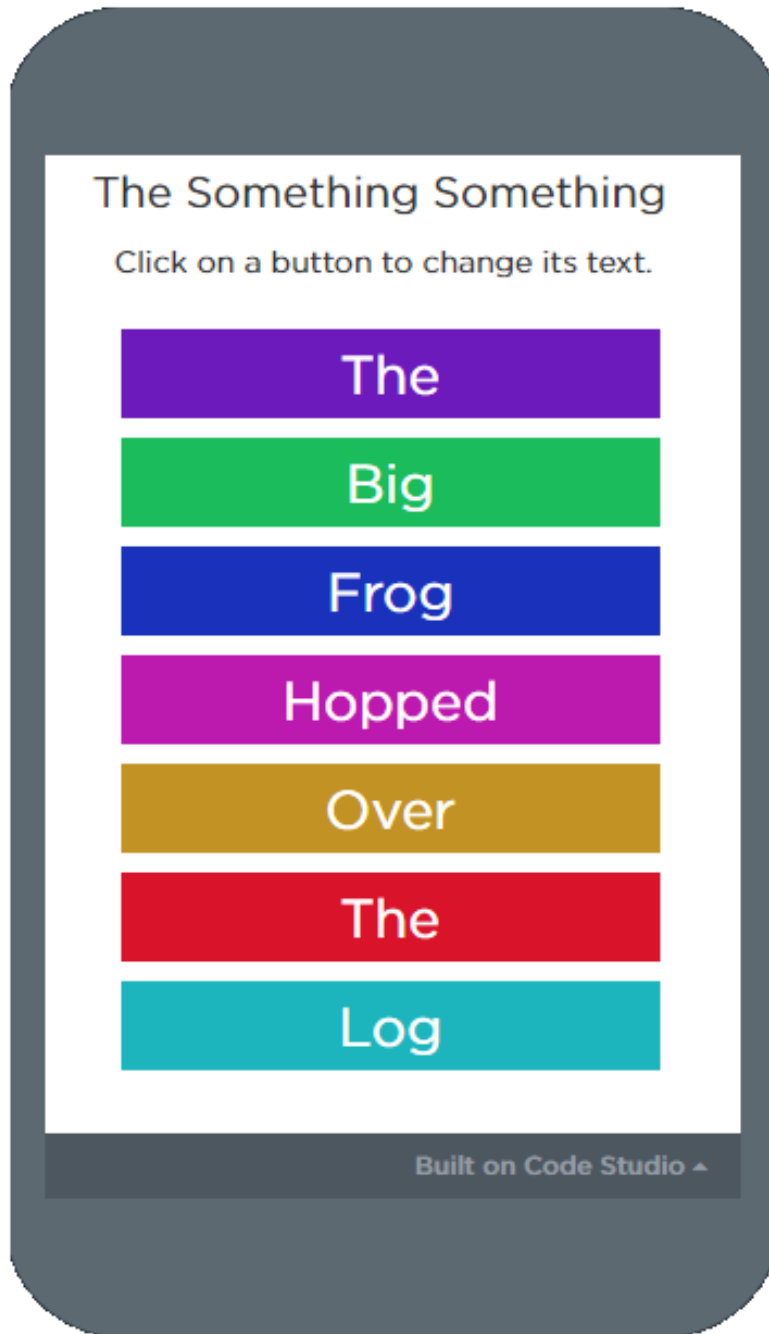
Canvas



Text Area



Slider



How many widgets are on this screen?

Which widget types are on this screen?

How many widget types are on this screen?

Widget Types:



Button



Text Input



Label



Dropdown



Radio Button



Checkbox



Image



Canvas



Screen



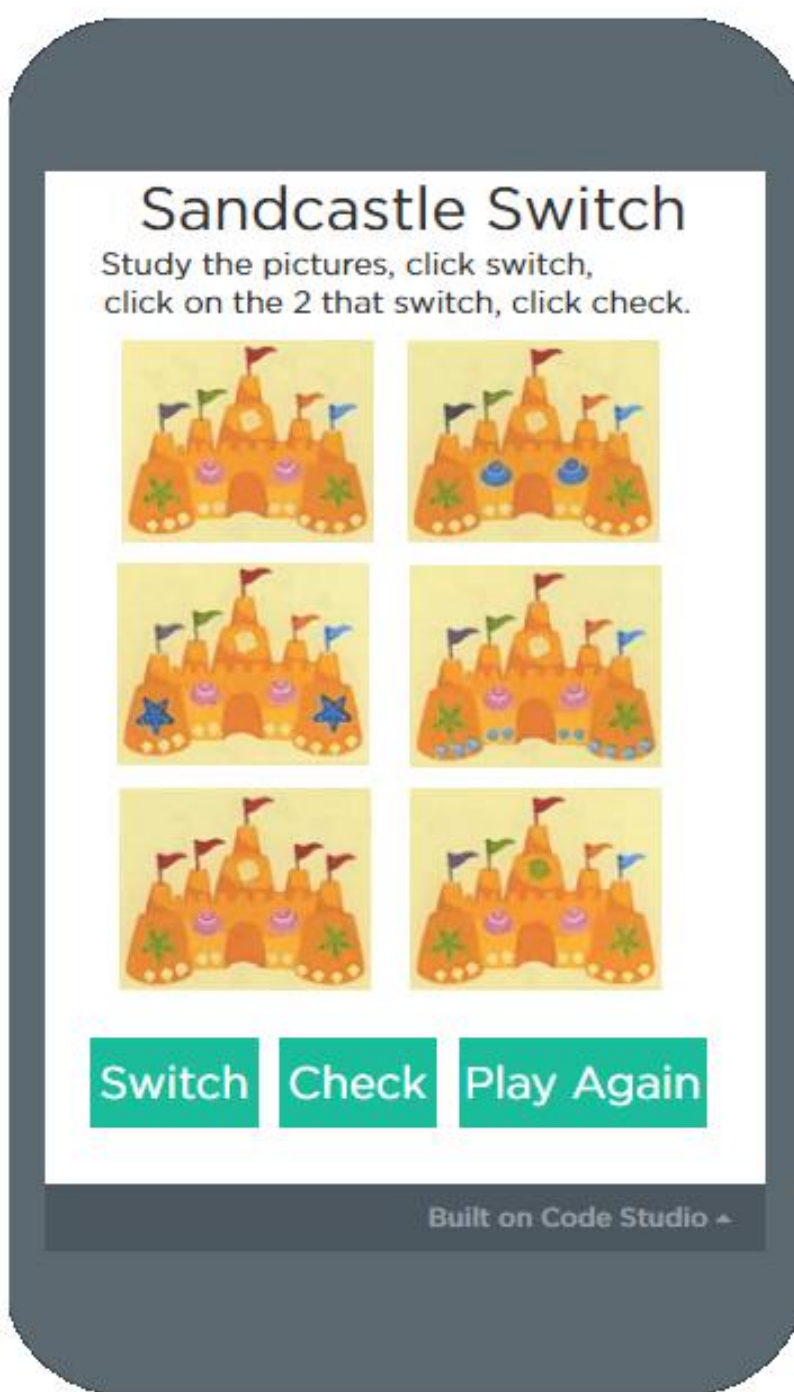
Text Area



Chart



Slider



How many widgets are on this screen?

Which widget types are on this screen?

How many widget types are on this screen?

Widget Types:



Button



Text Input



Label



Dropdown



Radio Button



Checkbox



Image



Canvas



Screen



Text Area



Chart



Slider

screen1

CE

<-

x^y

$x!$

7

8

9

/

4

5

6

X

1

2

3

-

0

.

=

+

▶ Run

PROPERTIESEVENTS

id

b8

text

8

width (px)

60

height (px)

55

x position (px)

95

y position (px)

180

text color

#ffffff

background color


#1abc9c

Each widget has properties that can be altered.

Identify a button properties on this screen.

What widget
type is the
spaceship?

What is
its widget
id?



screen1

PROPERTIES

EVENTS

id

ship

width (px)

48

height (px)

48

x position (px)

135

y position (px)

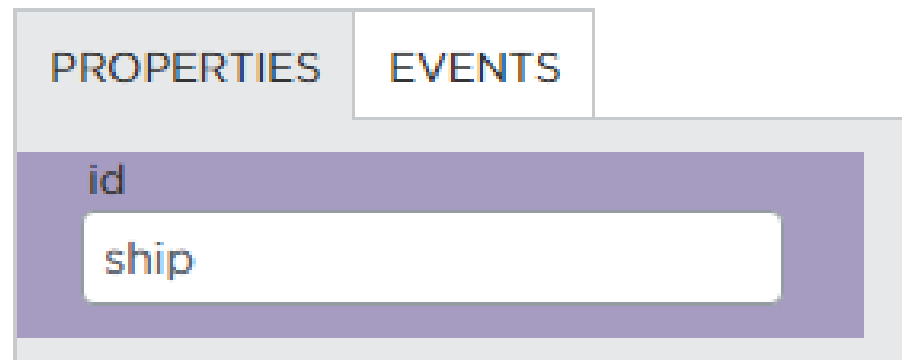
345

image

spaceship.png [Choose...](#)

fit image

fill



The id is an important property.

It is the name of the widget inside the code.

If the widget is a button OR it will change, it needs a meaningful id.
A meaningful id make the code easier to read.

```
onEvent(▼ "Right", ▼ "click", function(event) {  
  var shipX = getXPosition(▼ "ship") + 15;  
  var shipY = getYPosition(▼ "ship");  
  setPosition(▼ "ship", shipX, shipY, 48, 48);  
});
```


Not all ids are possible choices.
Which of the following are valid?

multiply

@school

Kitty_cat

2Price

First name

Last.name

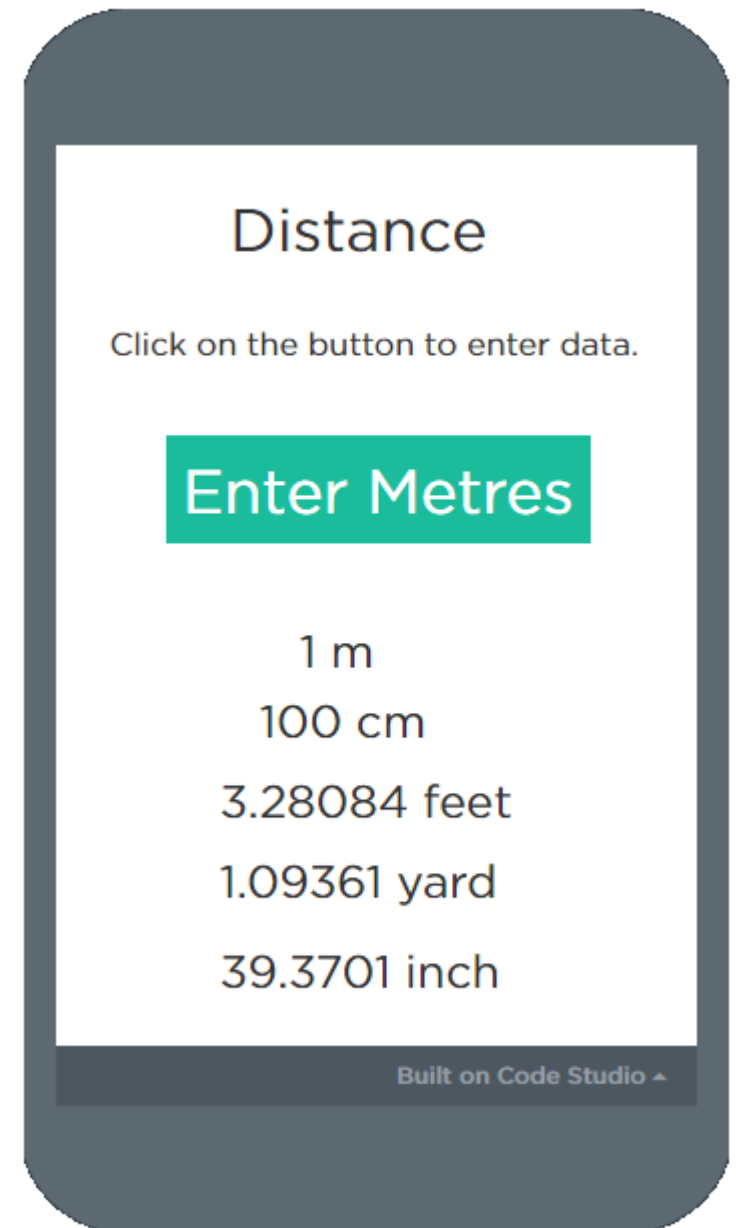
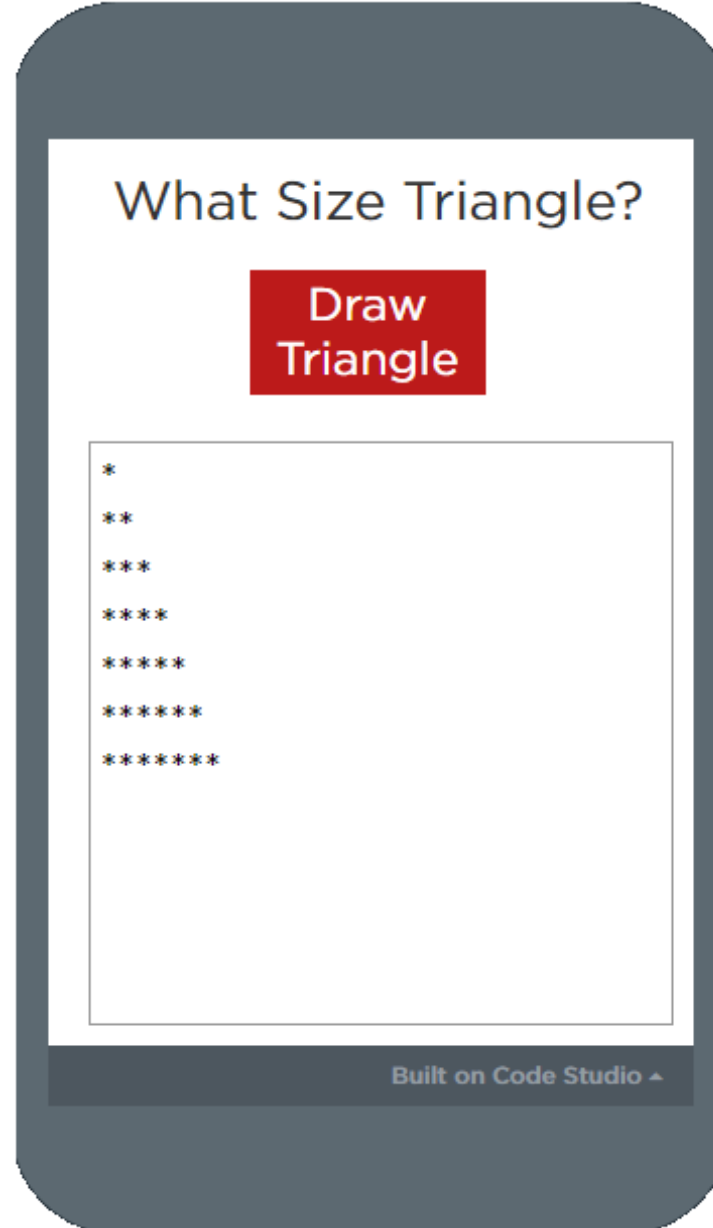
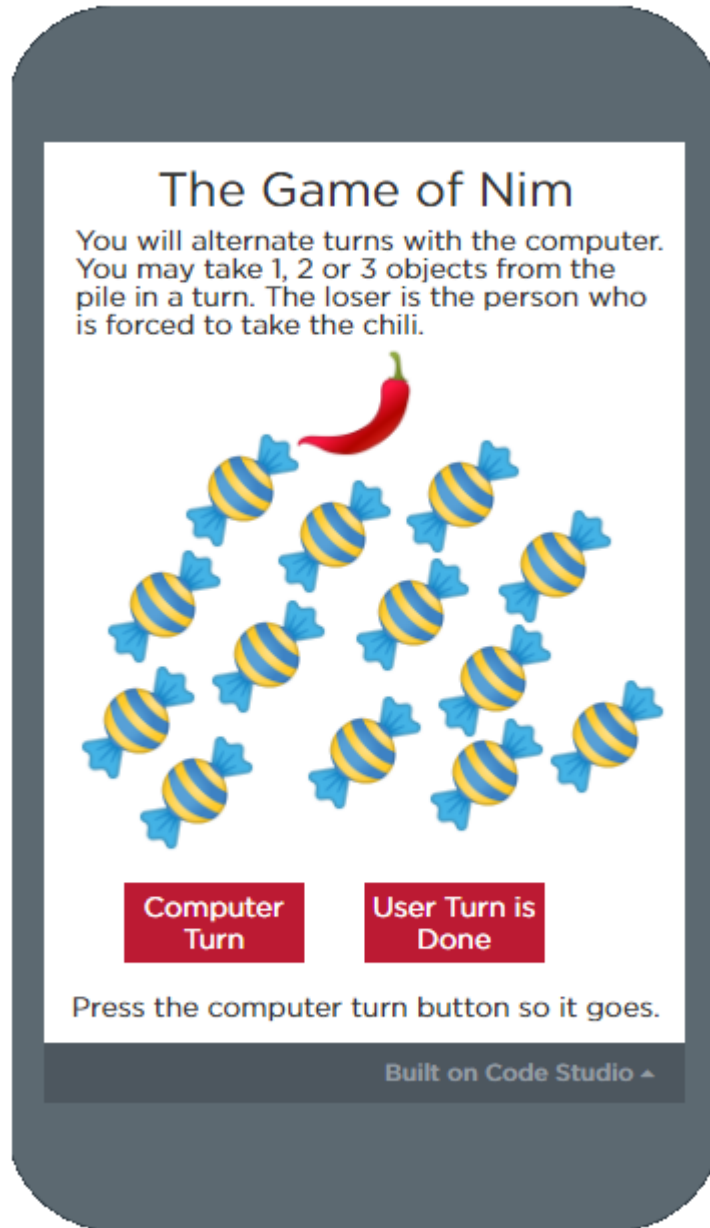
Price\$

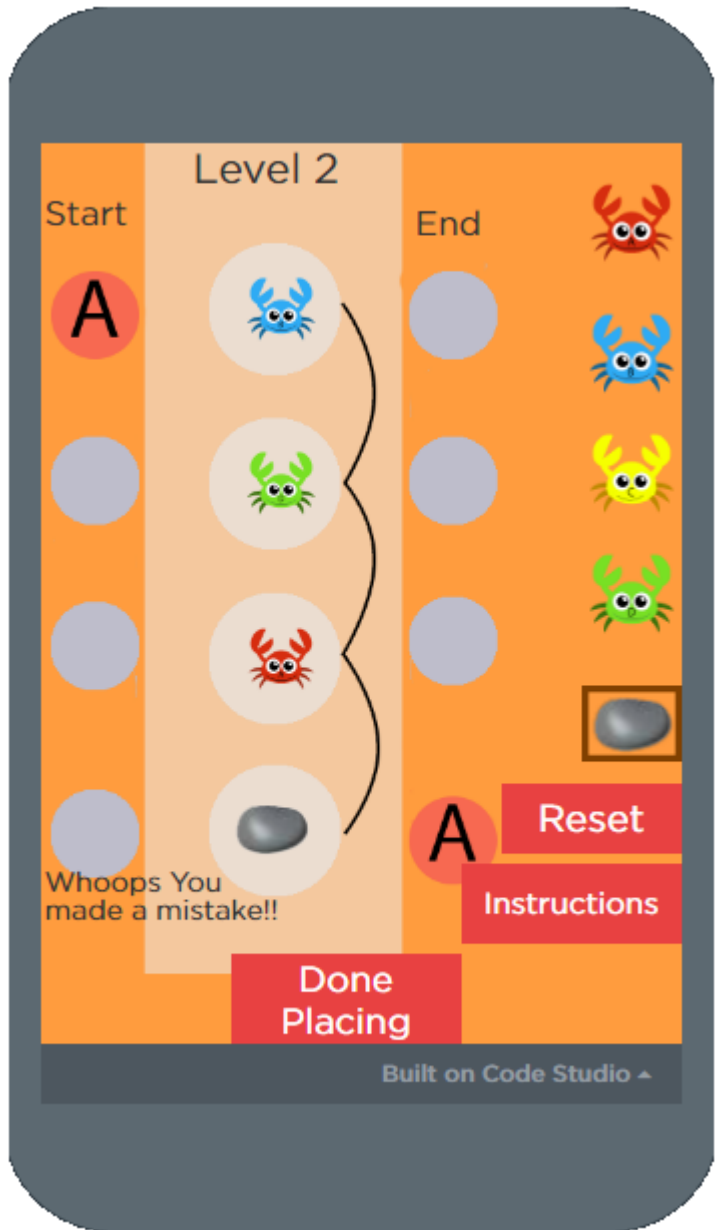
Price2

Id Naming Rules

1. Can't begin with a number
2. Can't have a space
3. Can't have odd characters. _ is NOT an odd character.
4. Should be meaningful

Some other AppLab Programs we will make:





Still more
AppLab Programs
we will make:

What
types of
widgets?

