KUAS
KYOTO UNIVERSITY of ADVANCED SCIENCE
京都発世界人財

京都先端科学大学

Introduction to Design (Track 3)

4.2 Hands-on: Counter App and BMI Calculator App

Zilu Liang

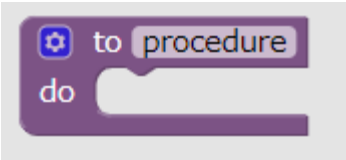
www.zilu-liang.net/id3

Important!!

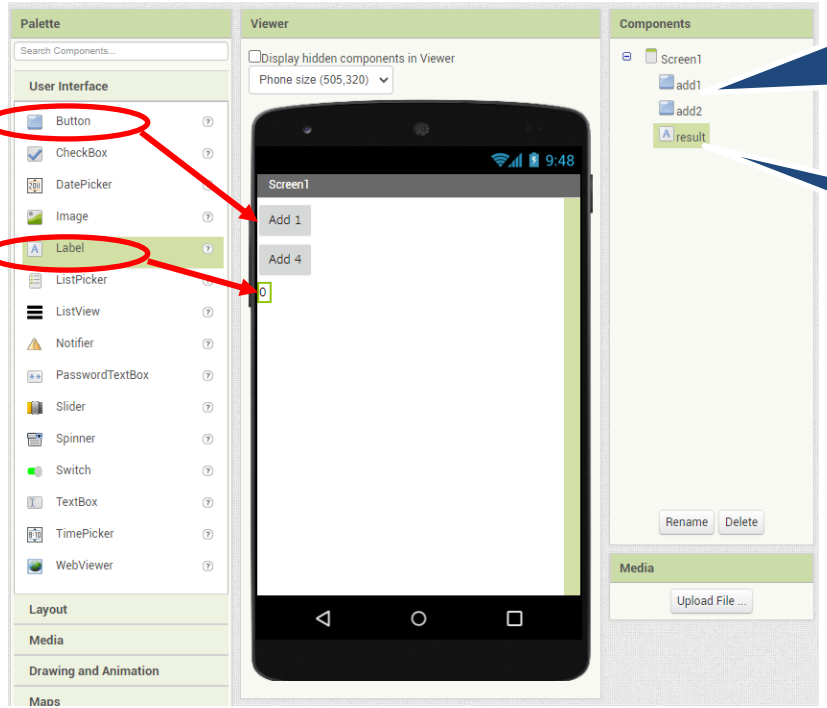
Please disinfectize your hands before entering the classroom!
入室前にアルコールを使用して手指消毒を行ってください。

Please disinfectize your chair and table!
①ペーパーにアルコールを噴霧してください。
②アルコールが噴霧されたペーパーで、使用箇所（テーブル、椅子など）を拭き取ってください。
③使用済のペーパーは廊下のごみ箱に捨ててください。

I. Counter app with ‘to-do’ procedure



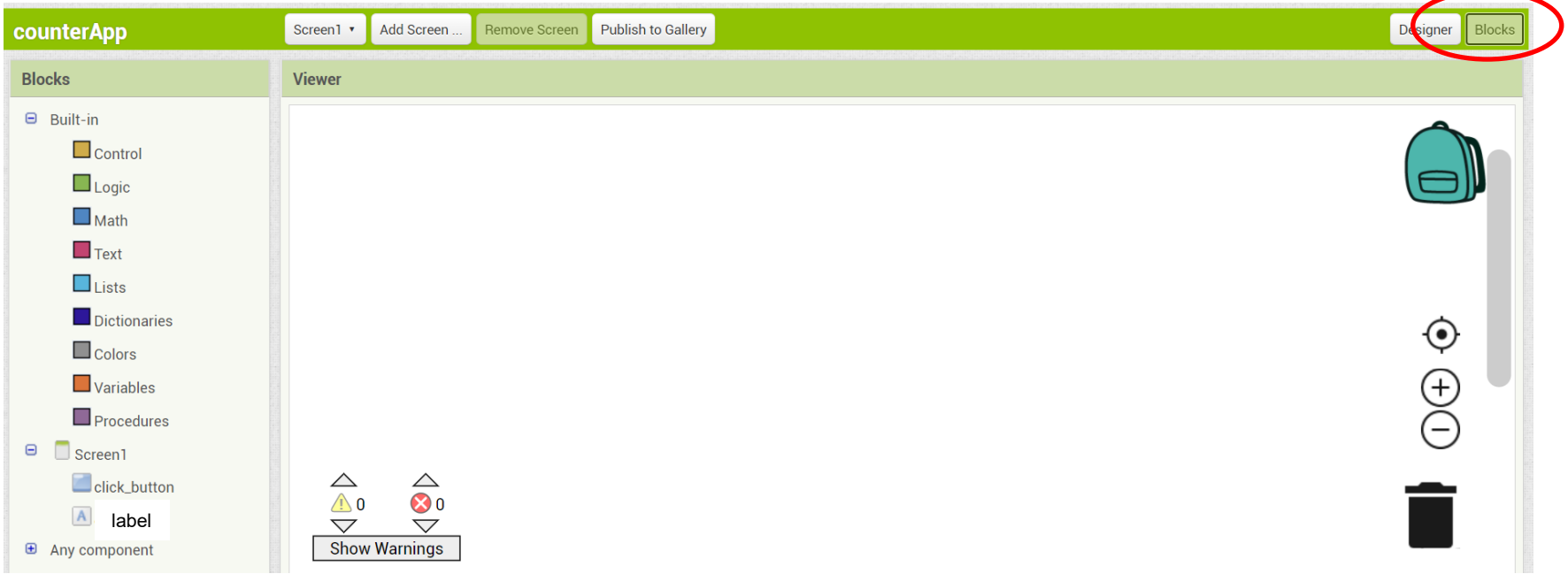
Build UI in designer view



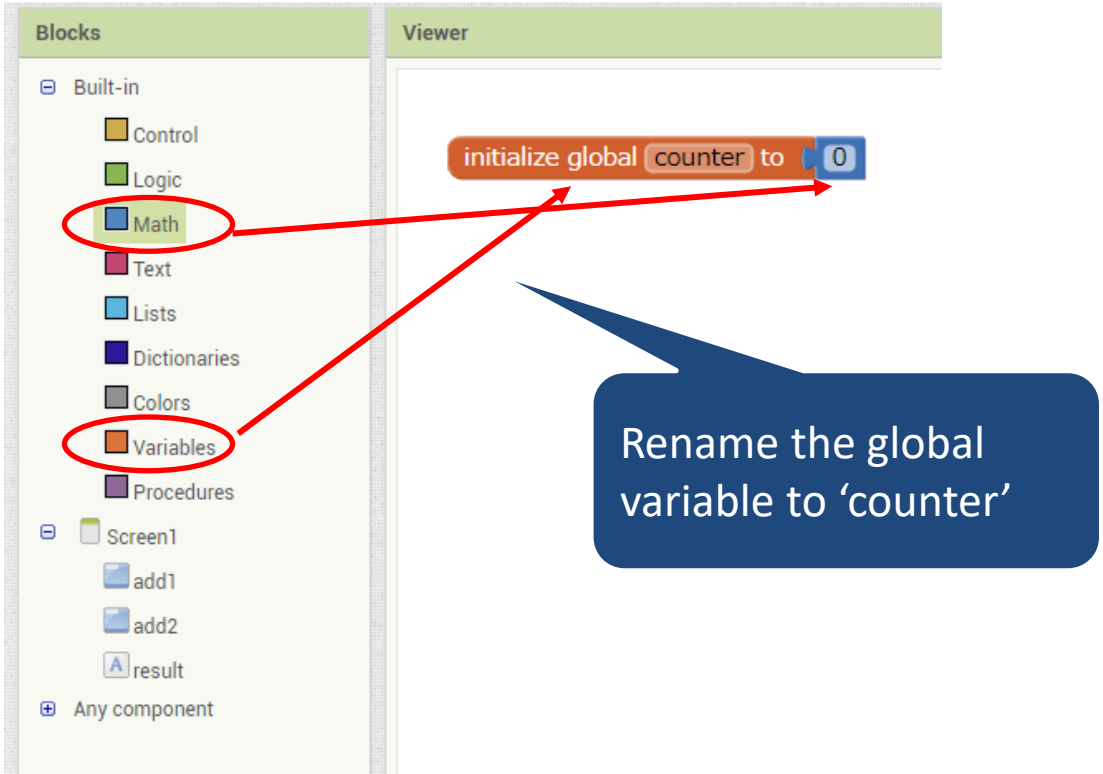
- Drag and drop two buttons
- Rename the buttons to ‘add1’ and ’add2’

- Drag and drop a label
- Rename the label to ‘result’

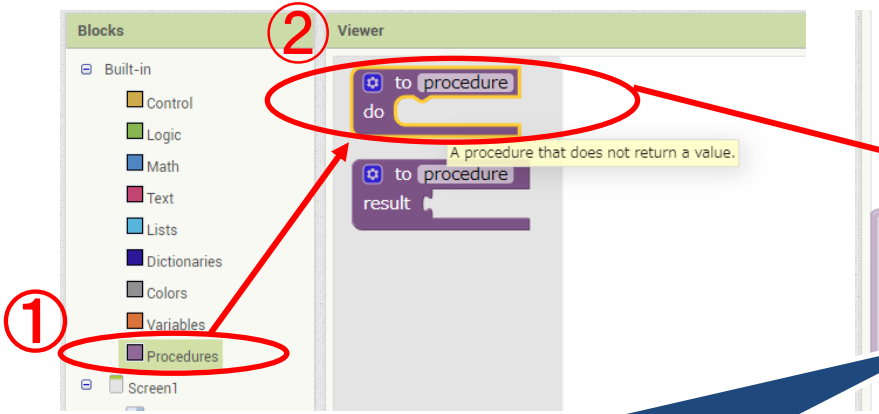
Switch to blocks view

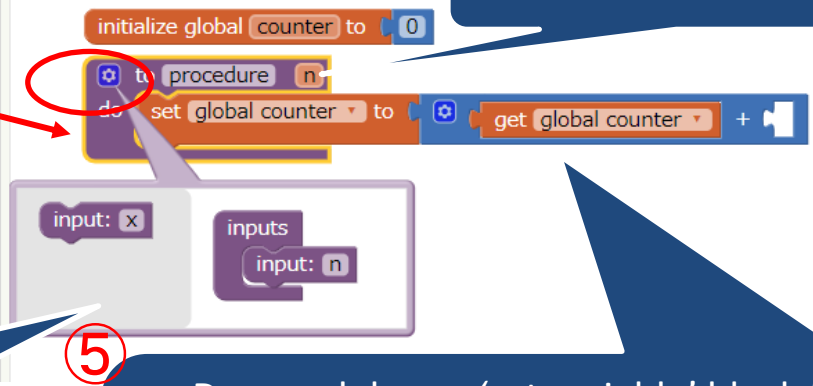


Initialize a global variable



Create a ‘to-do’ procedure



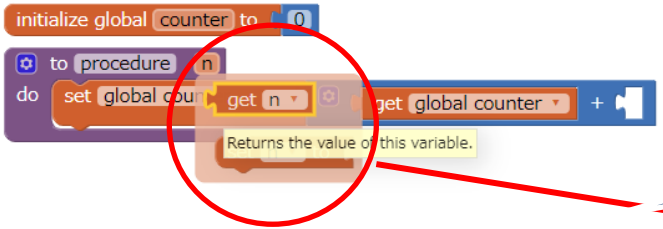


④ Rename the argument to ‘n’

③
- Click on the mutator
- Inside the mutator, add an argument by dragging an input block on the left side and dropping it inside the inputs block on the right side

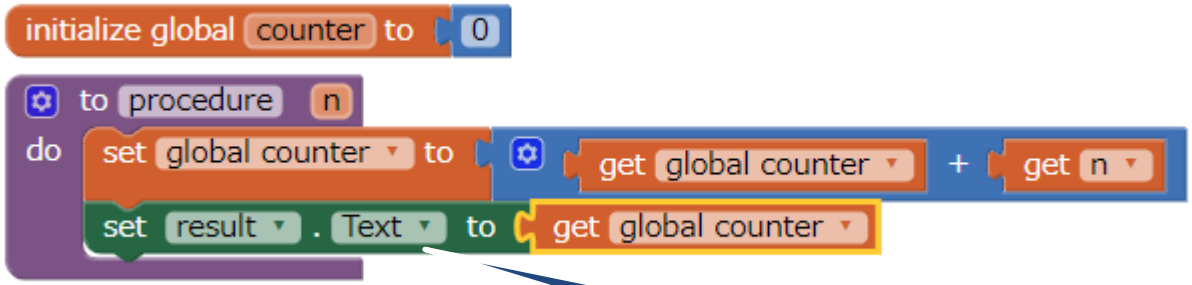
⑤
- Drag and drop a ‘set variable’ block from ‘Variables’
- Drag and drop an ‘add’ block from ‘Math’
- Drag and drop a ‘get ’variable value’ block from ‘Variables

Create a ‘to-do’ procedure



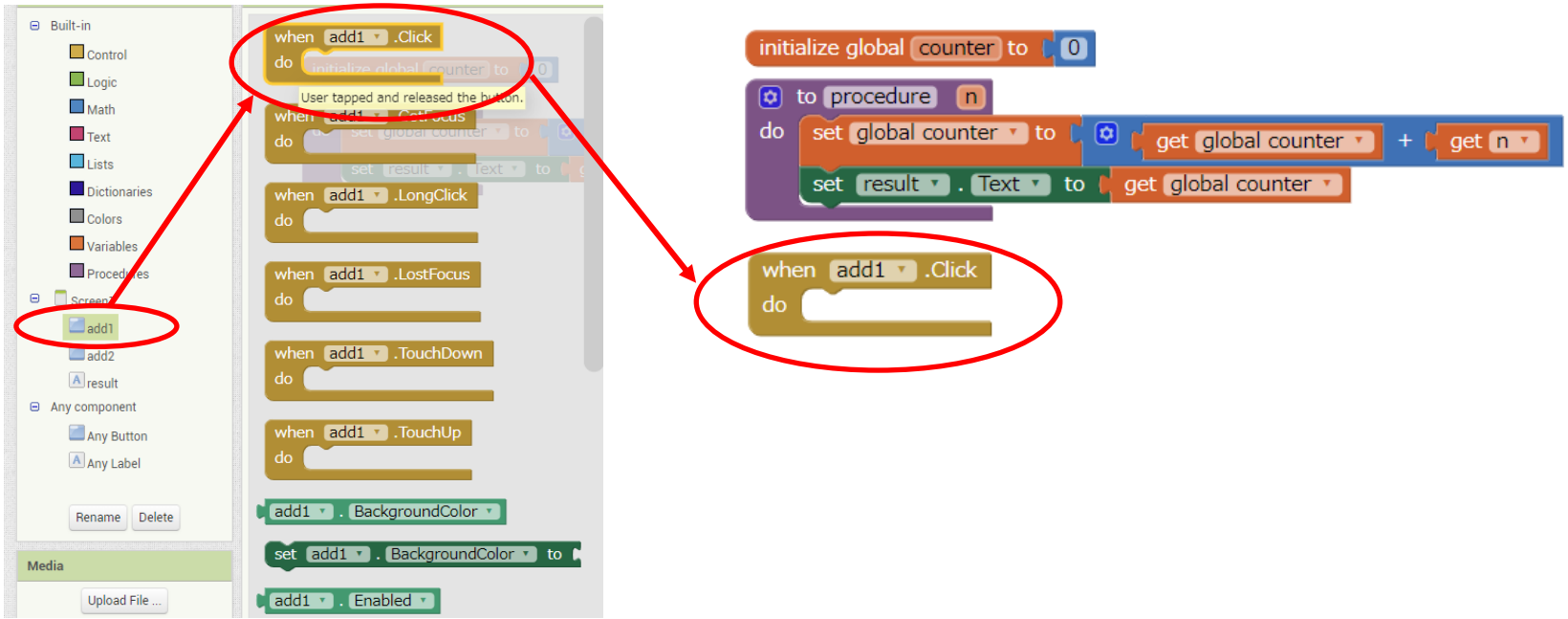
- Hover mouse on the argument ‘n’
- Drag the ‘get’ block and drop it inside the ‘add’ block

Create a ‘to-do’ procedure

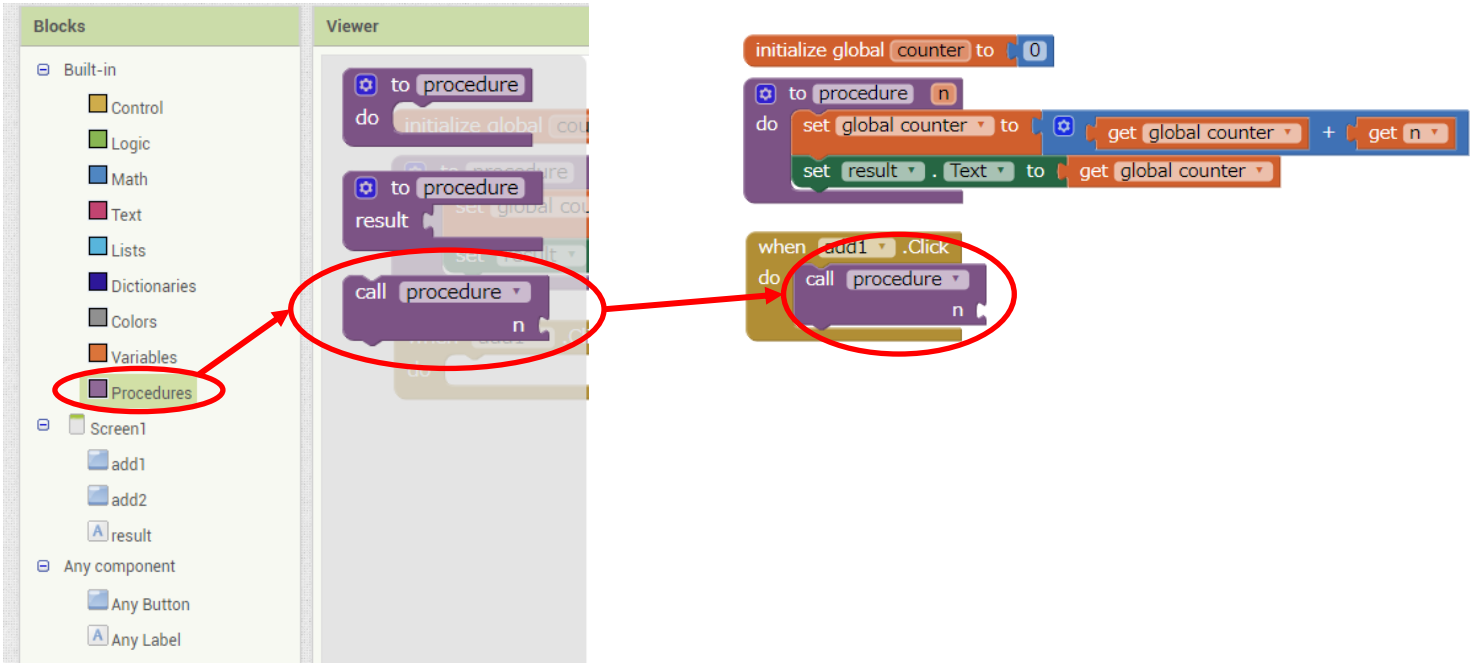


Set the result.text to the value of the global counter

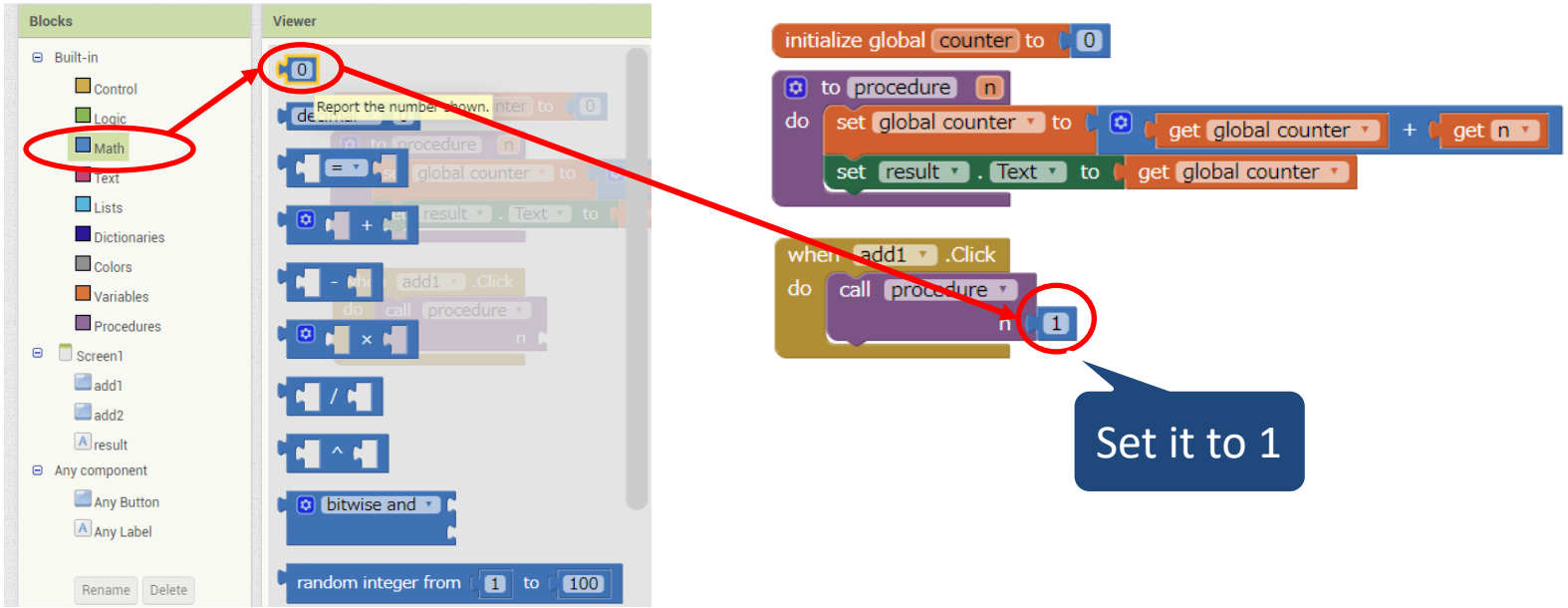
Add click handler to button ‘add1’



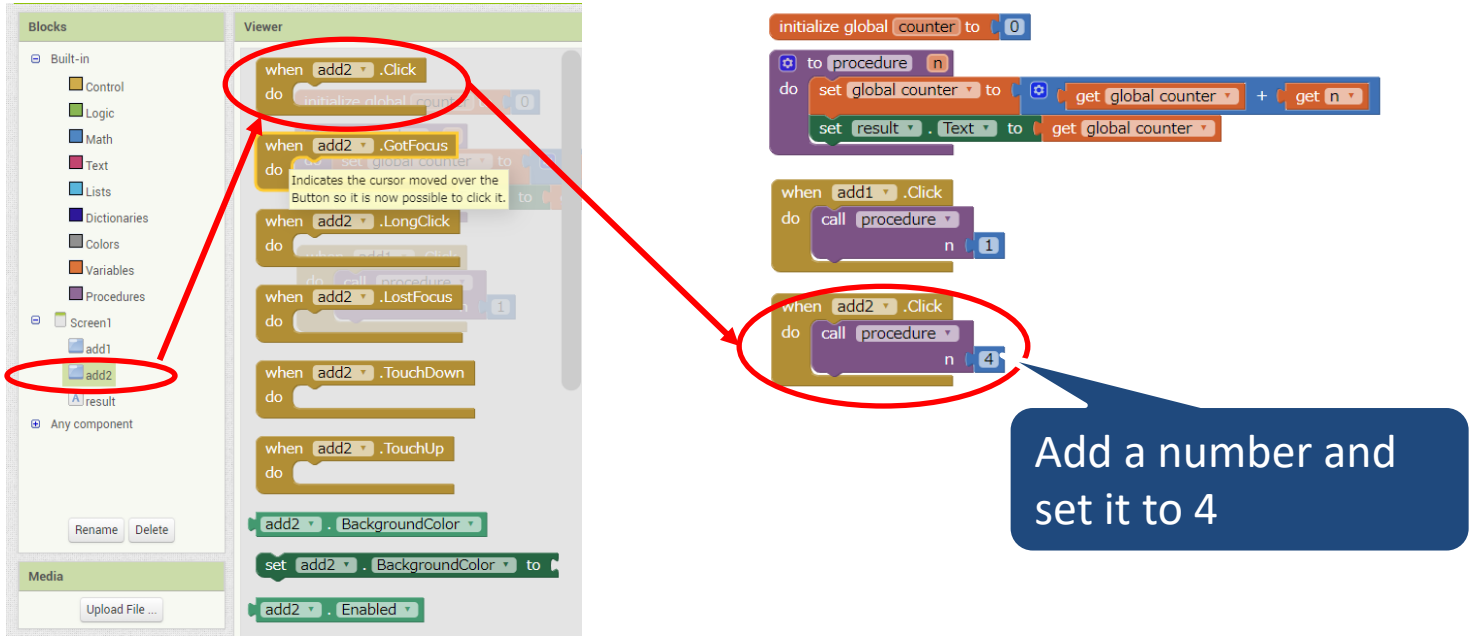
Call the procedure in click handler



Set the procedure argument

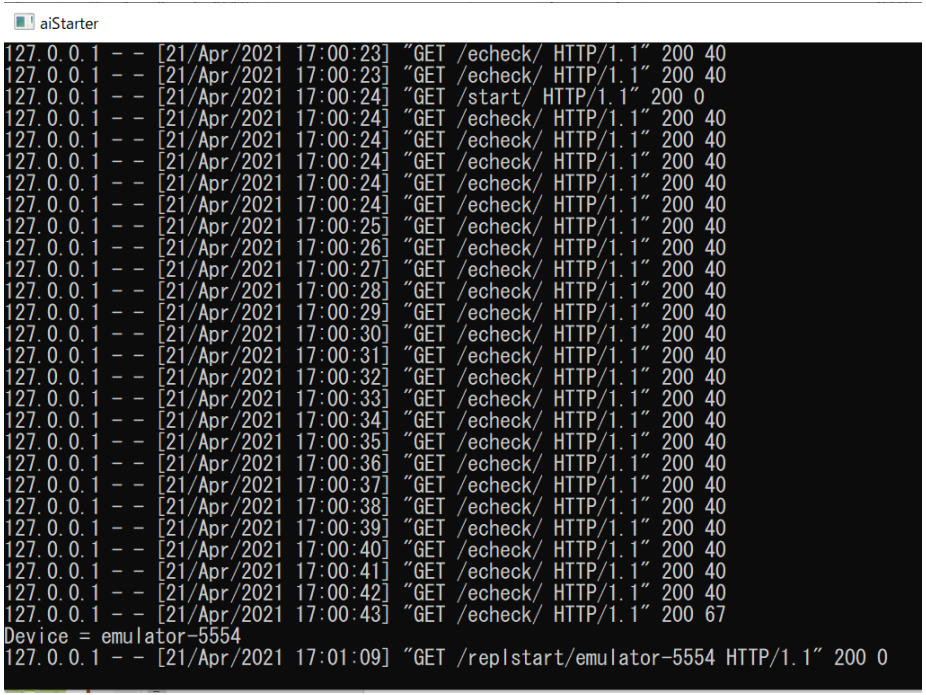


Add click handler to button ‘add2’

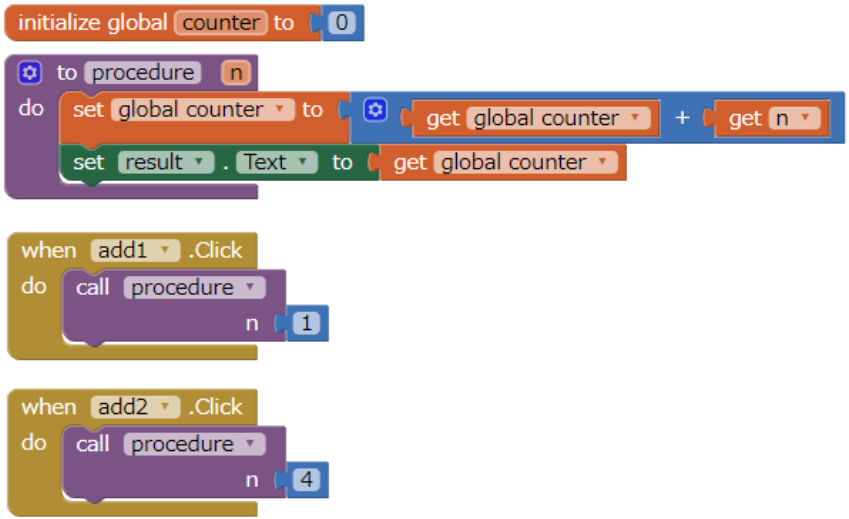


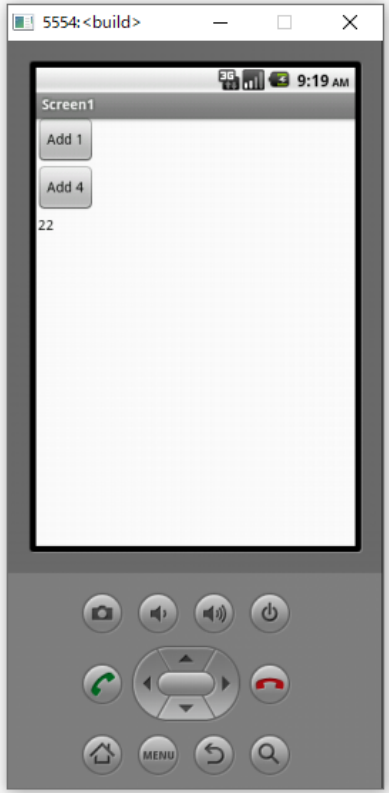
Test App on Emulator

Click on the aiStarter on your computer, you should see the following window open.

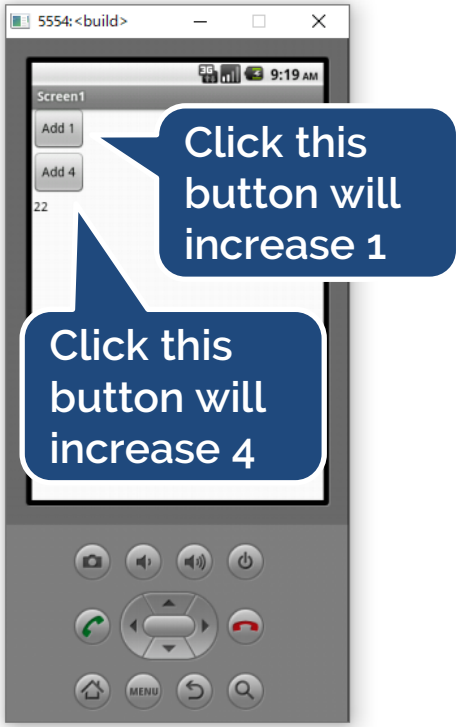


Test App on Emulator



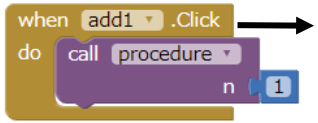


Code Anatomy: Counter App

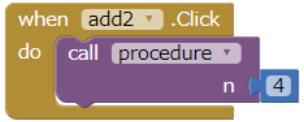


Initialize the global variable ‘counter’ and set it to 0

Define a ‘procedure’ that takes in the argument ‘n’, increase the counter by ‘n’, and show the result in the label

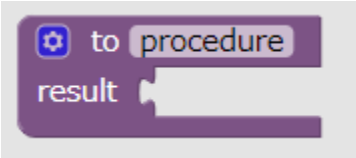


When ‘add1’ button is clicked, call ‘procedure’ and pass the argument ‘n=1’

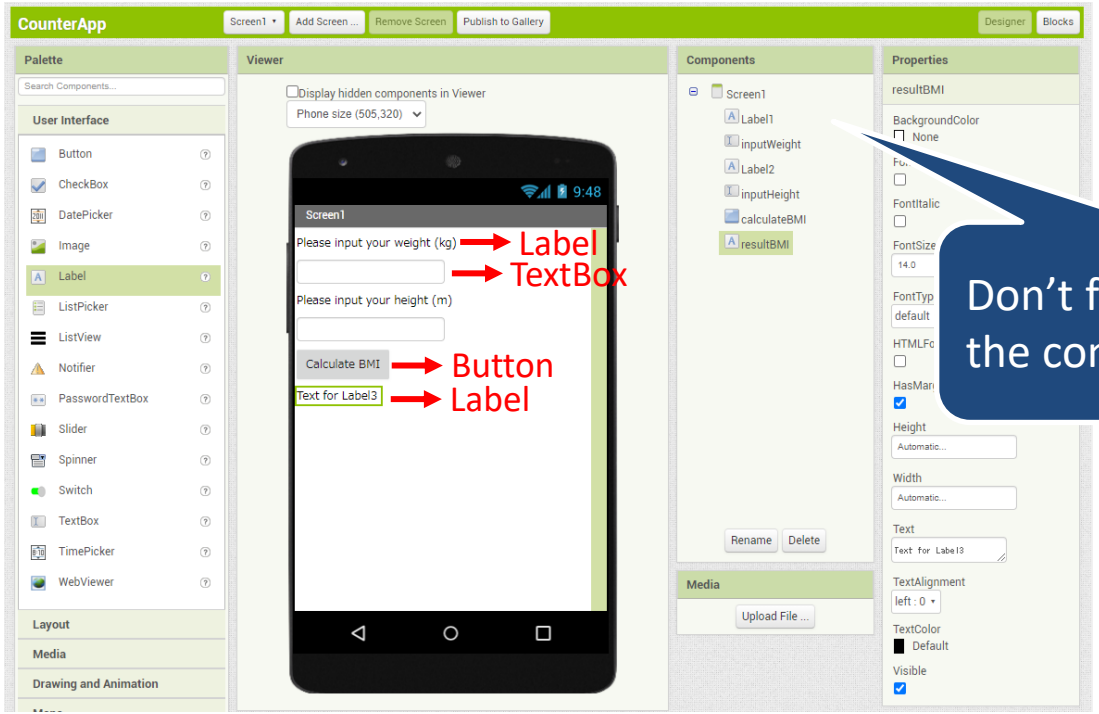


When ‘add2’ button is clicked, call ‘procedure’ and pass the argument ‘n=4’

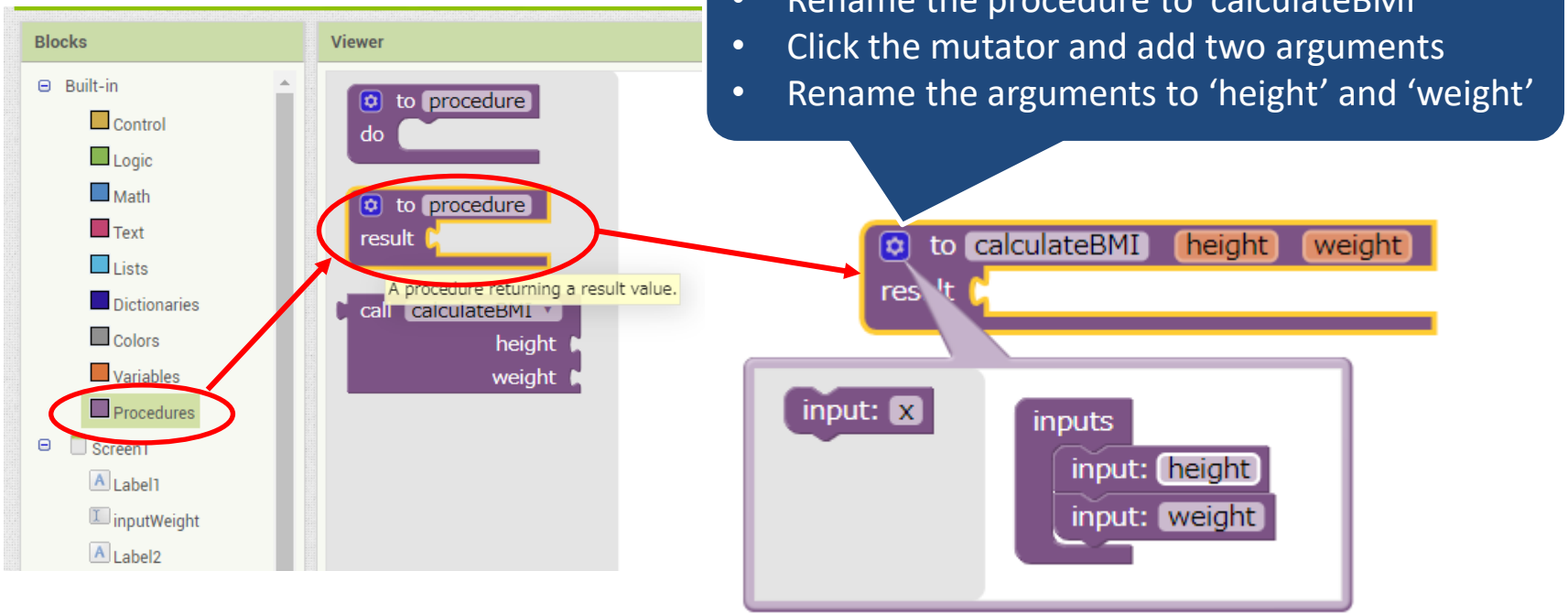
II. BMI calculator app with ‘to-result’ procedure



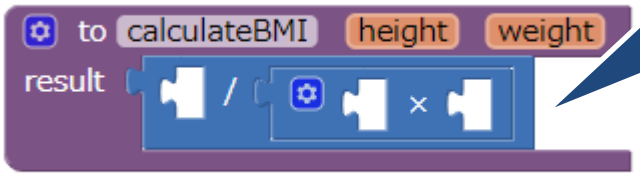
Build UI in designer view



Create a ‘to-result’ procedure

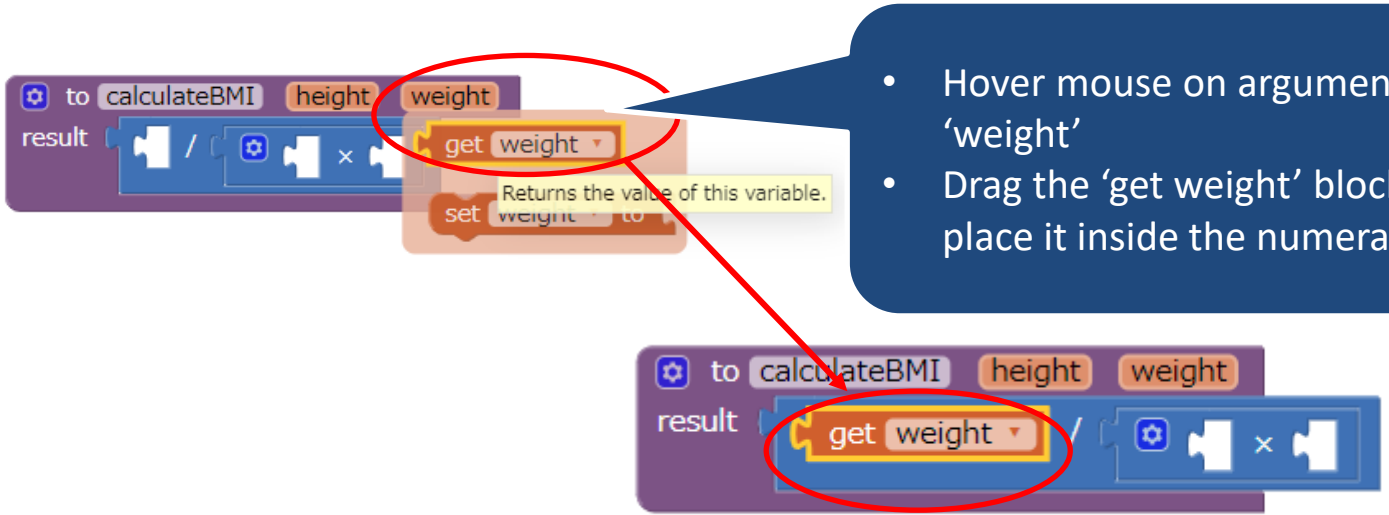


Create a ‘to-result’ procedure



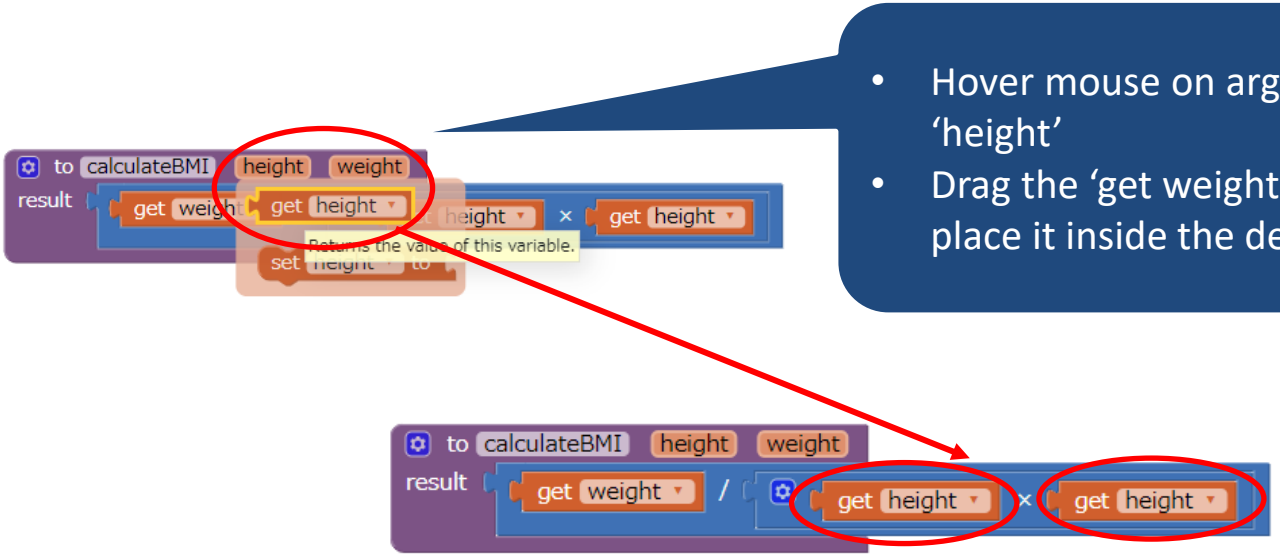
- Drag a division block from ‘Math’ and drop it inside the procedure
- Drag a multiplication block from ‘Math’ and drop it inside the denominator

Create a ‘to-result’ procedure



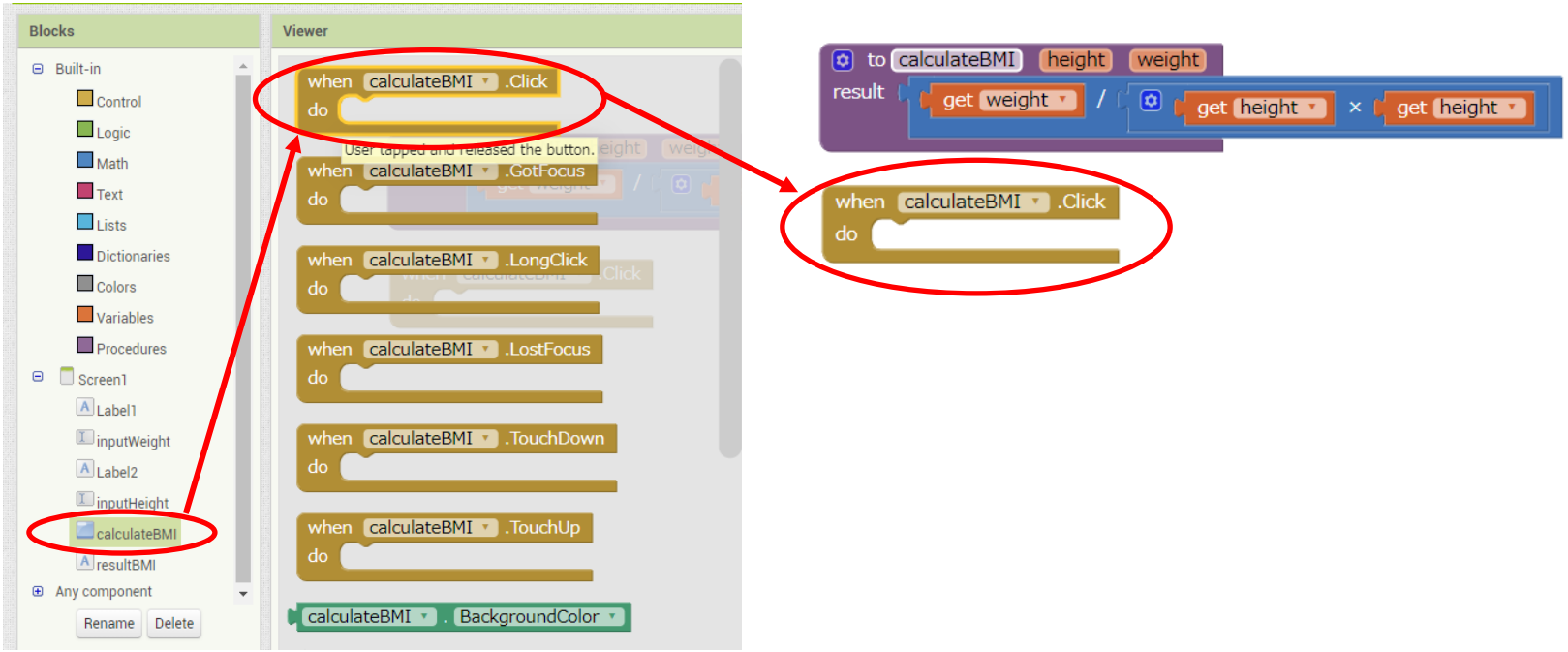
- Hover mouse on argument ‘weight’
- Drag the ‘get weight’ block and place it inside the numerator

Create a ‘to-result’ procedure

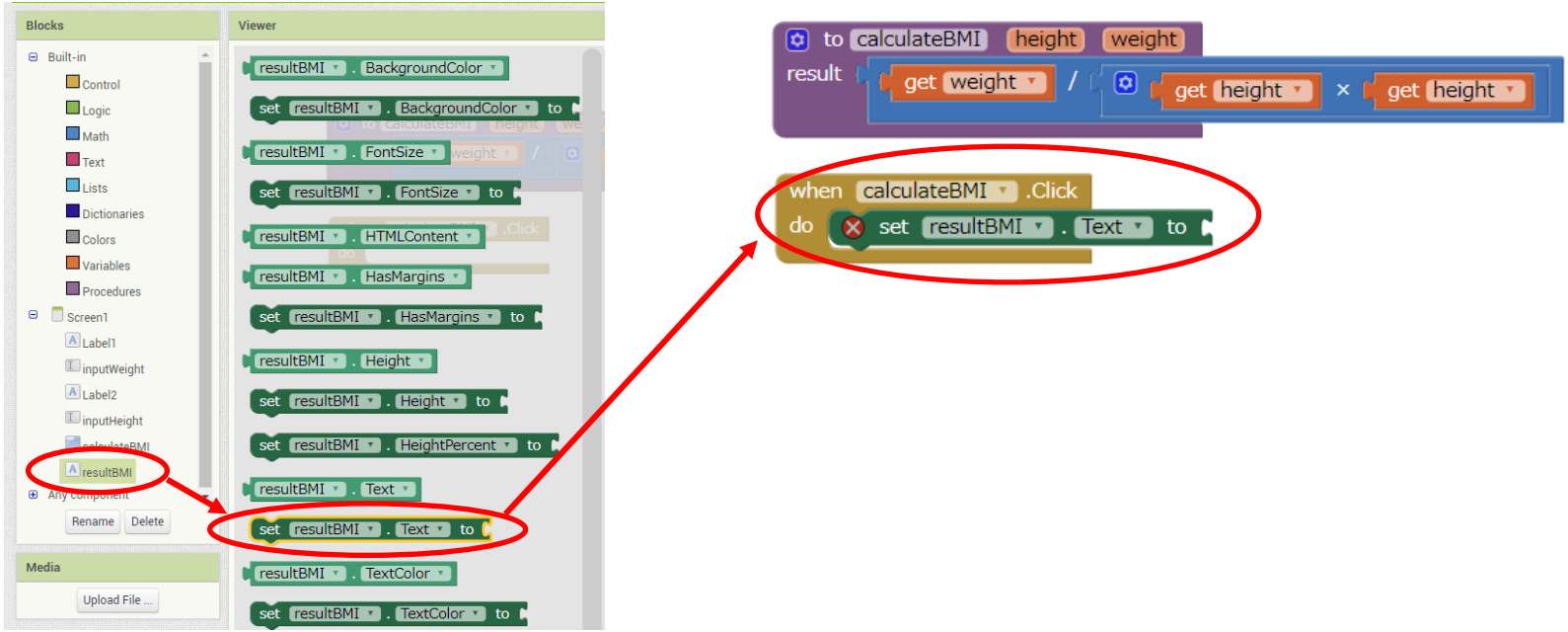


- Hover mouse on argument ‘height’
- Drag the ‘get weight’ block and place it inside the denominator

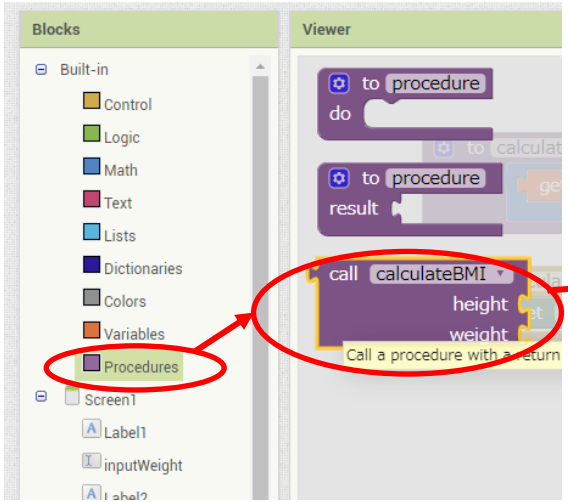
Add click handler to button ‘calculateBMI’

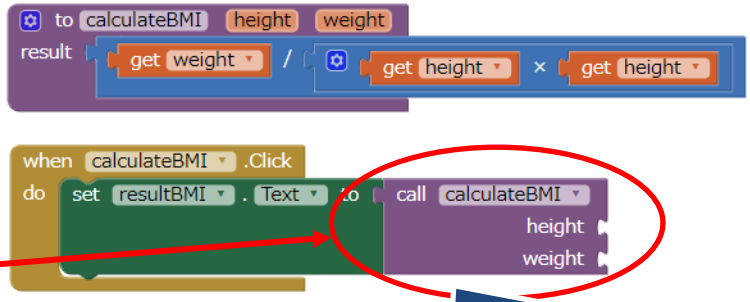


Add click handler to button ‘calculateBMI’



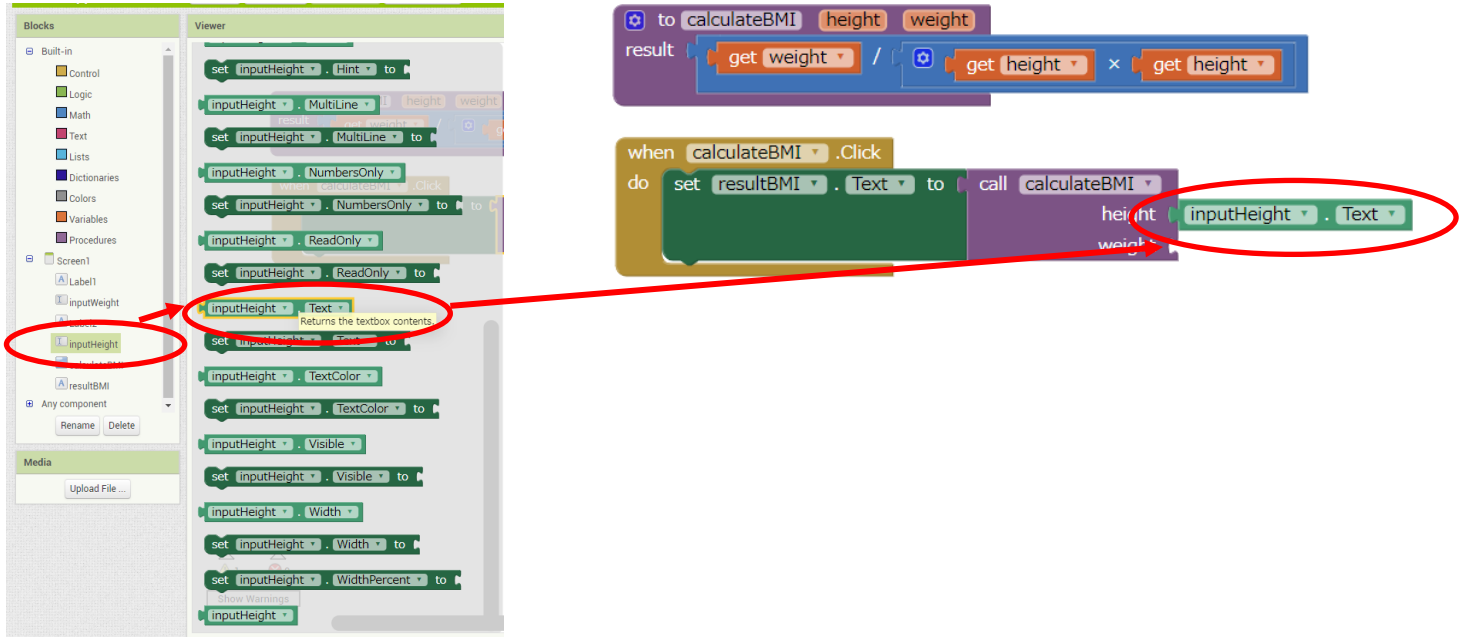
Add click handler to button ‘calculateBMI’



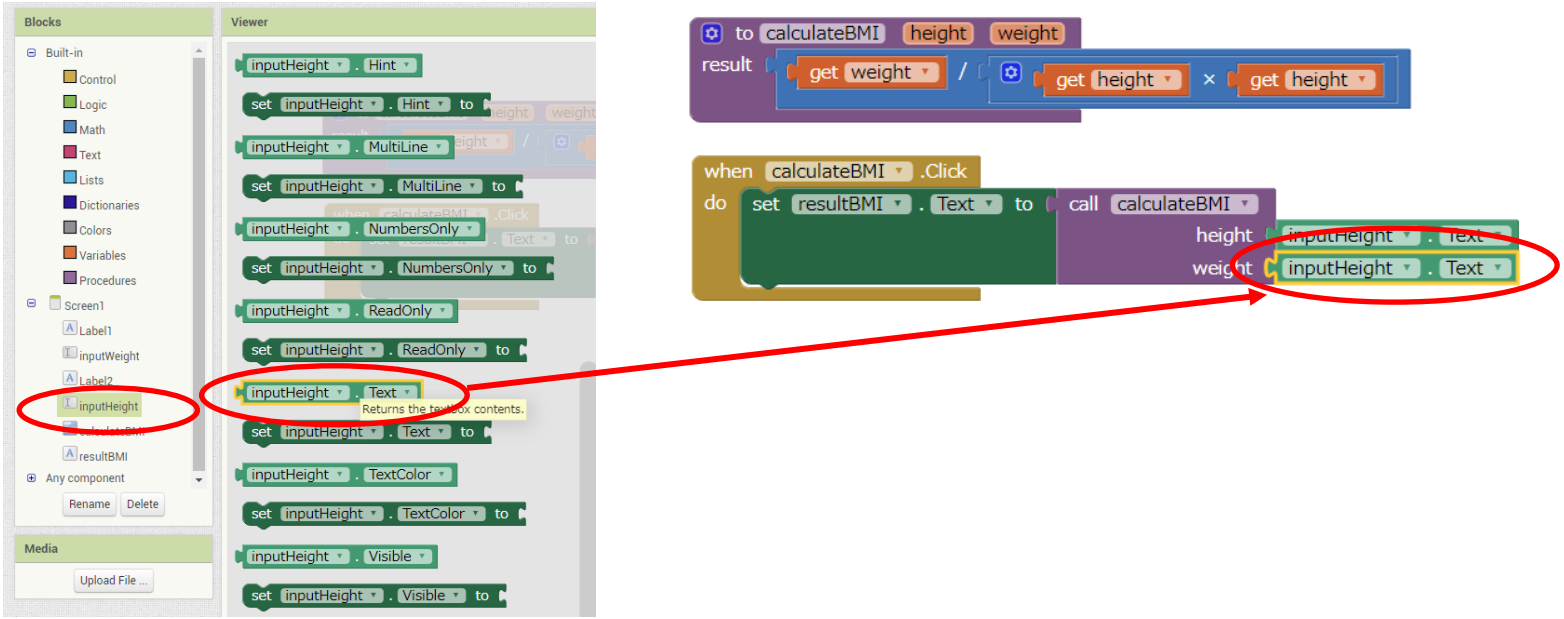


Call the ‘calculateBMI’ procedure and set the ‘resultBMI.Text’ to the return value.

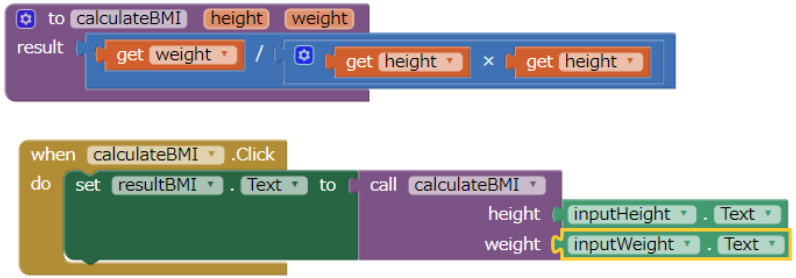
Get argument values from TextBox

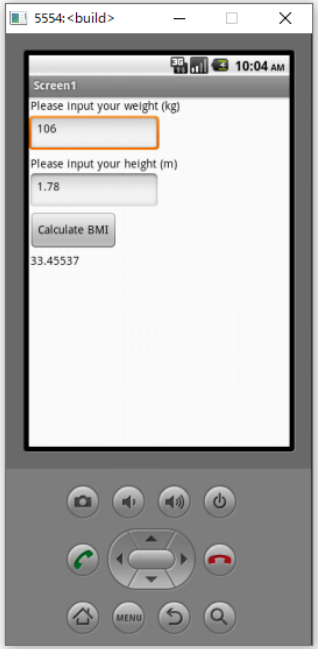


Get argument values from TextBox



Test on Emulator



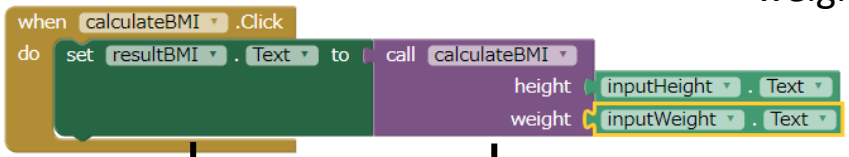


Code Anatomy: BMI Calculator App

A user needs to input weight and height

Click this button will calculate BMI

Define a 'calculateBMI' procedure that takes in the argument 'height' and 'weight' to calculate BMI



1. Set the 'weight' and 'height' arguments to the values that a user input in the textboxes

2. Call the 'calculateBMI' procedure and pass the two arguments 'weight' and 'height', and return the result value to the block on the left

3. Set the 'resultBMI' text to the result returned from the 'calculateBMI' procedure

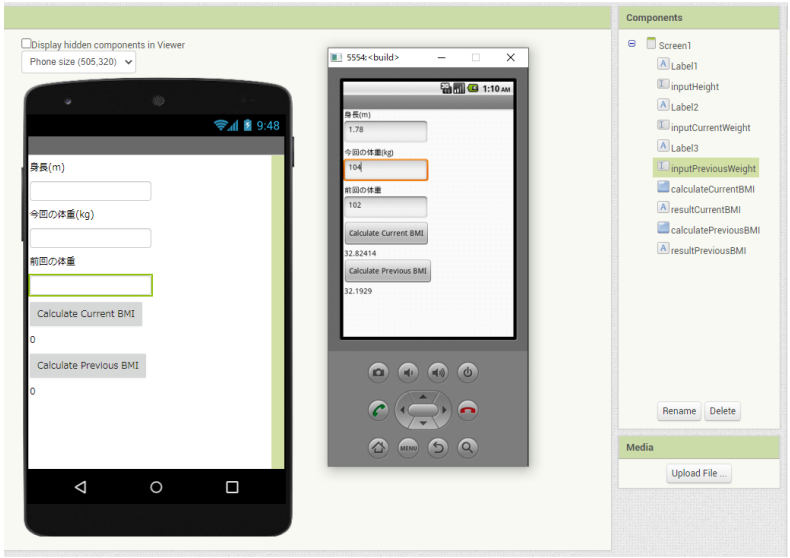
Assignment

- Complete the hands-on tasks in the tutorial
  - ✓ If you finish all the steps in class, show your Counter App and BMI Calculator App to one of the instructors before you leave
  - ✓ If you cannot finish all the steps, you can work on them after class and show your Apps to one of the instructors in the class next week

Assignment (Optional)

If you have time, why not trying to add more features to the BMI Calculator app!

1. Additional feature 1: calculate previous BMI
   - Add a TextBox that allows user to input their previous body weight
   - Add a button ‘Calculate Previous BMI’, so that when a user clicks this button, the app calculates the previous BMI and show it below the button (you must use the ‘calculateBMI’ procedure in the click handler)



Assignment (Optional)

2. Feature 2: show a simple advice based on the comparison of current weight and previous weight

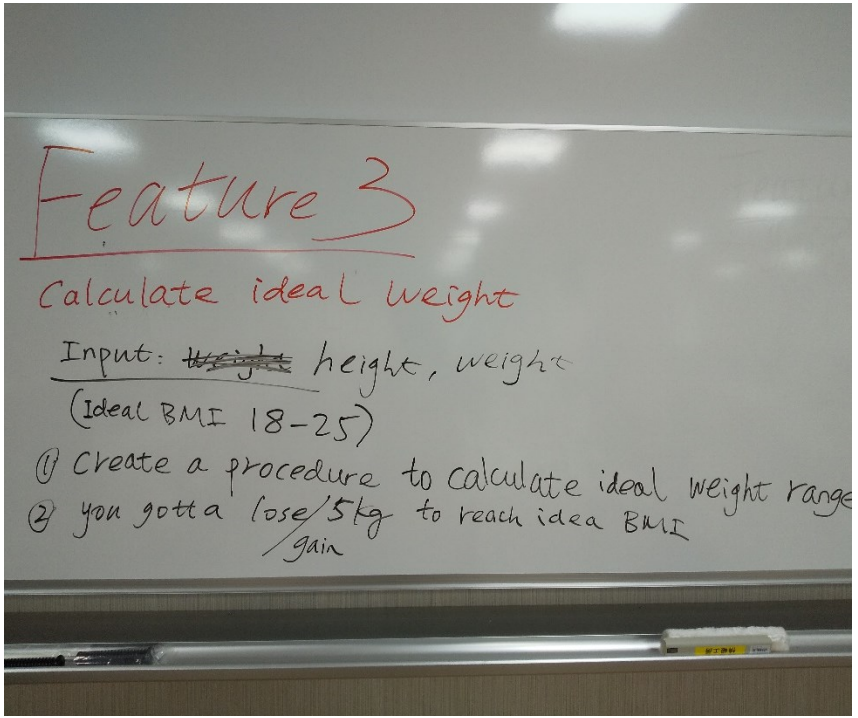
(Hint: you need to use the ‘if-then-else if-then-else’ block in the drawer of ‘Control’)

Assignment (Optional)

3. Feature 3: Calculate the ideal weight range

(1) Create a procedure to calculate ideal weight range based on a user's height

(2) Provide advice like 'another 5 kg to lose/gain before reaching the idea BMI'




KUAS
KYOTO UNIVERSITY of ADVANCED SCIENCE
京都先端科学大学