

6.1 Loop

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!

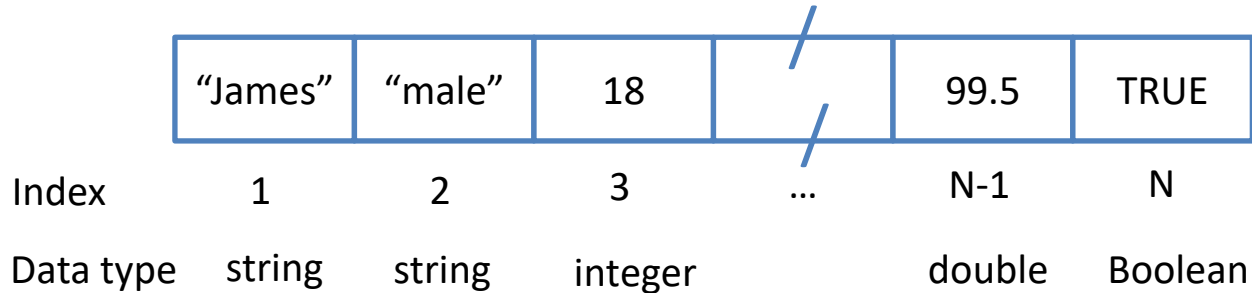
- ①ペーパーにアルコールを噴霧してください。
- ②アルコールが噴霧されたペーパーで、使用箇所（テーブル、椅子など）を拭き取ってください。
- ③使用済のペーパーは廊下のごみ箱に捨ててください。



1. Recap of Week 5

List

- A list is multiple pieces of data stuck together, one after another in fixed order.
- The position of an element in a list is called its `??`. In App Inventor, the first element in a list is index `??`.
- The data of each item can be different types.



List

- The list itself is a container and can be treated as a single piece of data
- List can be stored in variables (they are just a type of data) and passed as inputs and outputs to blocks

initialize global `my_list` to  create empty list

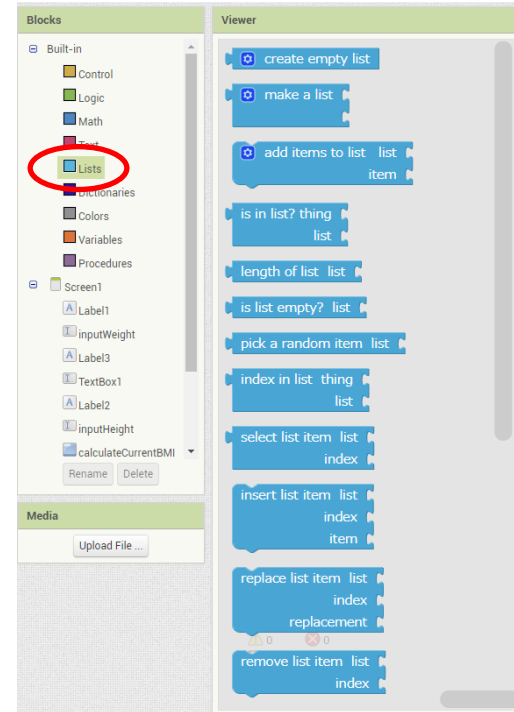
Initialize a global variable with data type of list

get global `my_list`

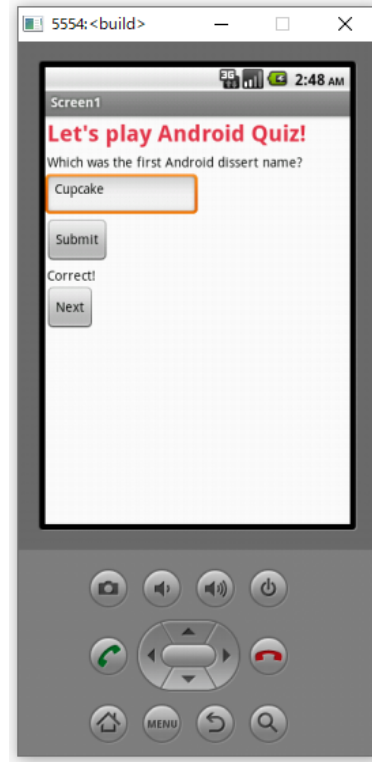
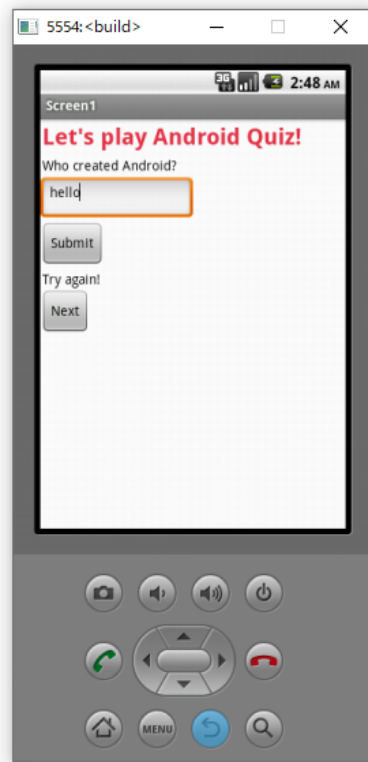
Using the getter block, we'll get all items in `my_list`

The List Block in App Inventor

- Create empty list
- Make a list with some items
- Add items to list
- Add items to list
- Get the length of the list
- Check if the list is empty
- ...



Hands-on: Android Quiz App



Code Anatomy: Android Quiz App

Initialize a global variable to store the index of the current question

```
initialize global currentQuestionIndex to 0
```

If 'Next' button is clicked, increase the index by 1

```
when nextButton .Click
```

```
do set global currentQuestionIndex to get global currentQuestionIndex + 1
```

```
if get global currentQuestionIndex > length of list list get global QuestionList
```

```
then set global currentQuestionIndex to 1
```

If it's larger than the length of the question list, reset it to 1.

```
set showQuestionLabel .Text to select list item list get global QuestionList index get global currentQuestionIndex
```

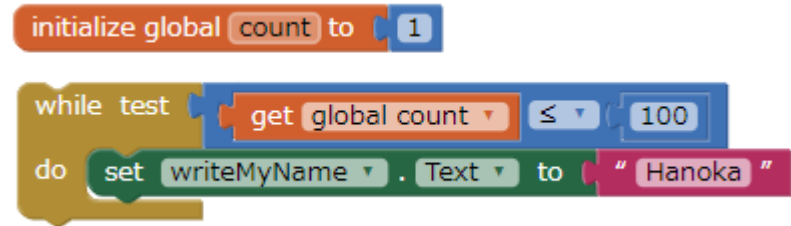
Show question

2. Loop

Loops

- Loops are a way to tell a computer to do something many times.
- Two types of loops: while-loops & for-loops

Imagine you were asked to write your name down 100 times in a row. This might take you a long time, and you might make a few mistakes along the way.



```
initialize global count to 1
while test
  get global count ≤ 100
do
  set writeMyName . Text to "Hanoka"
```

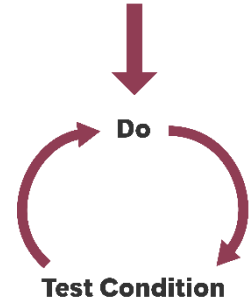
The image shows a sequence of Scratch code blocks. The first block is an orange 'initialize global' block with 'count' selected and the value '1'. The second block is a yellow 'while test' block with a blue 'get global count' block, a blue '≤' operator, and a blue '100' block. The third block is a green 'do' block with a green 'set writeMyName . Text' block and a pink 'Hanoka' text block.

While-Loops

- While-loops will continue to go until a condition is no longer true.
- The loop will check the condition each time before it repeats to make sure the condition is still true.
- You should choose the condition carefully
 - It needs to start out as true; otherwise, the loop will never run.
 - If you choose a condition that will never be false, then your loop will never stop.



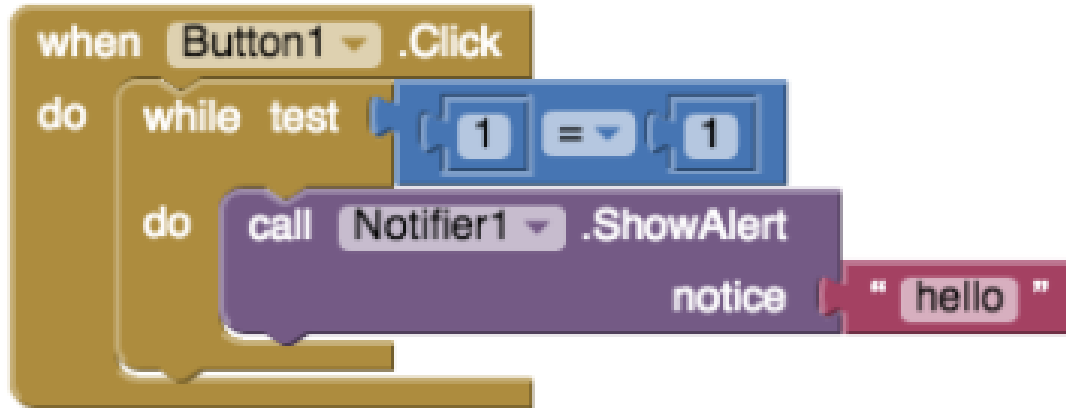
Start Loop
if condition = true



if condition = false
Exit Loop

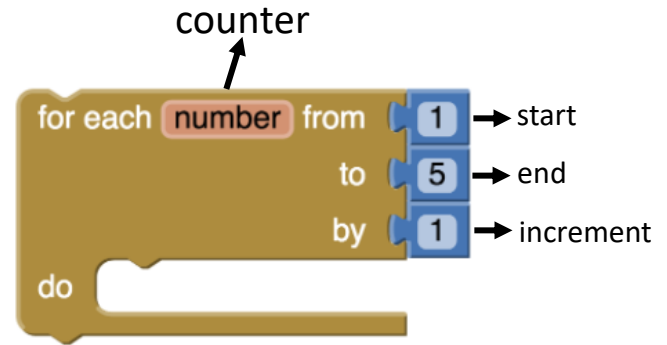
While-Loops

What's the problem with this while-loop?

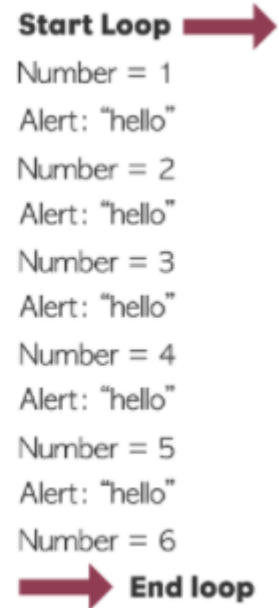
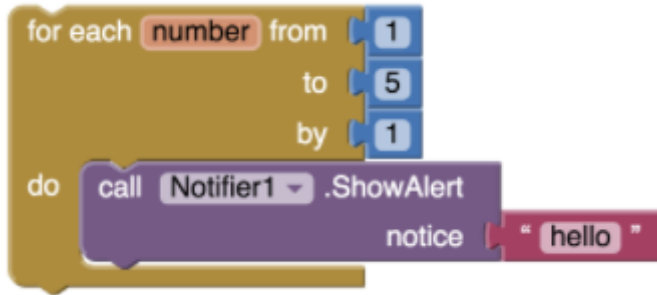


For-Loops

- For-loops will repeat a block of code for a set number of times.
- For-loops use a 'counter' variable to count how many times the code has been repeated. You need to set
 - the start and end for the counter
 - how much the counter goes up by each time the code repeats; in many cases, we set the counter to increase by 1 each time the loop repeats.



For-Loops Example



For-Each-Loops

- For-each-loops will repeat for the number of items in a list.
- Very useful whenever you need to do something with a list.



For-Each-Loops Example

What will be shown in Label1? Is there any information missing?

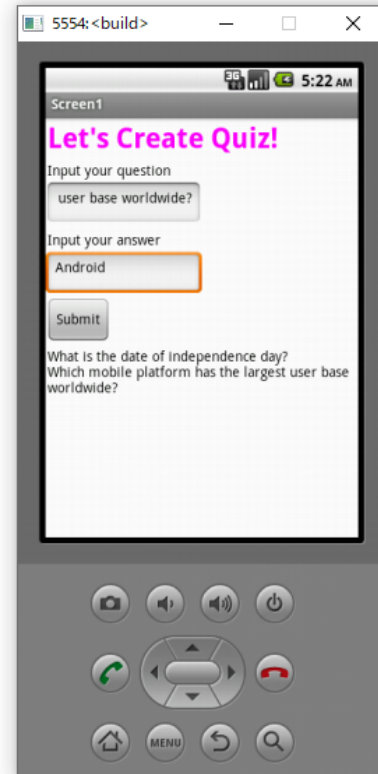
```
initialize global numberList to [make a list [44 [7 [16]]]
```

```
for each item in list [get global numberList] do [set global sum to [get global sum + get item]]  
set Label1 . Text to [get global sum]
```


Hands-on Preview

```
initialize global QuestionList to create empty list
initialize global AnswerList to create empty list

when Submit.Click
do
  add items to list list get global QuestionList
  item inputQuestionTextbox.Text
  add items to list list get global AnswerList
  item inputAnswerTextbox.Text
  set showQuestionList.Text to ""
  for each item in list get global QuestionList
  do
    set showQuestionList.Text to join showQuestionList.Text
    get item
    "%n"
```



Any questions?

KUAS

KYOTO UNIVERSITY of ADVANCED SCIENCE

京都先端科学大学