Lesson4 PartI: Basic Concept of App Inventor

【OS & Software】

Windows XP/7, App Inventor

【Learning Outcomes】

Be familiar with MIT APP Inventor, programming with APP. Running your own APP in android phone.

【Instructions】

1) Introduce to MIT App Inventor

App Inventor is a B/S model software, you can use is once you got the right URL. In our lab we have our own URL. You can get the URL from your teacher.

1. To know the function of each element.
2. To understand the basic component of App Inventor.
3. Programming with APP Inventor, and downloading APK running in your android phone.

2) Make a calculator in MIT App Inventor

1. Complete the design of interface for calculator.
2. Be able to support arithmetic operation.

【Instructions】

1) Introduce to App Inventor

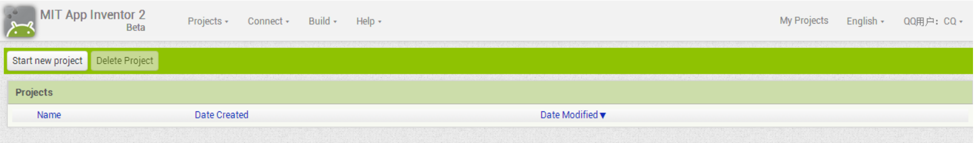
App Inventor is a B/S software, you can use any browser as follows to use it:

•Google Chrome 29+

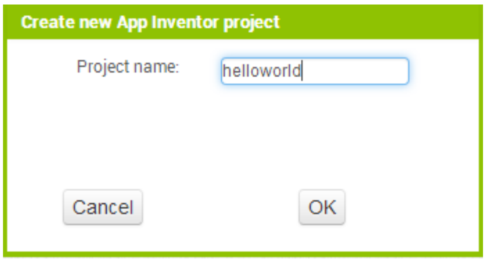
•Safari 5+

•Firefox 23+

1. Open the specify URL which is published by your teacher (http://ai2.appinventor.mit.edu/ or http://app.gzjkw.net/?locale=en) .

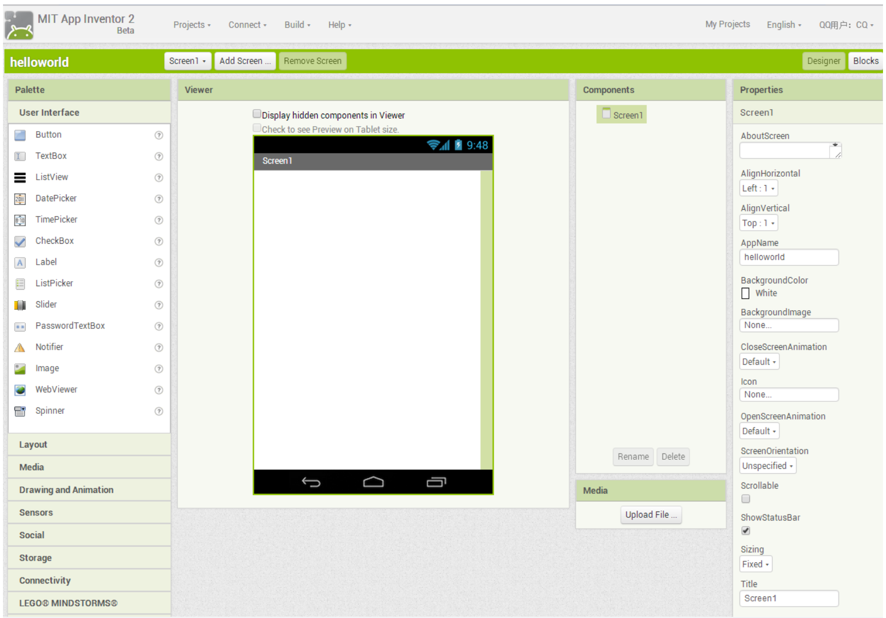


2. Start new Project, input the project name.



Project name: helloworld

Enter in the main board.



Screen1

AboutScreen

AlignHorizontal: Left : 1

AlignVertical: Top : 1

AppName: helloworld

BackgroundColor: ☐ White

BackgroundImage: None...

CloseScreenAnimation: Default

Icon: None...

OpenScreenAnimation: Default

ScreenOrientation: Unspecified

Scrollable: ☐

ShowStatusBar: ☑

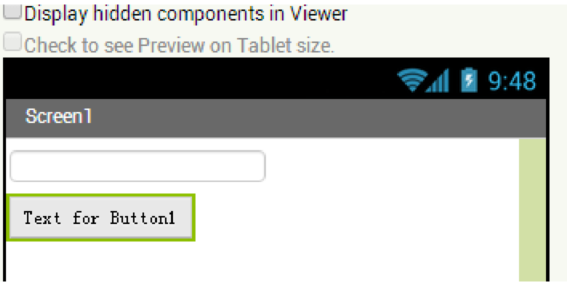
Sizing: Fixed

Title: Screen1

The left side contains Palette, there are Button, Textbox and et al. "User interface" and other component like "Layout" are also exists in Palette. With the help of simple User interface control, we can build our first simple App—HelloWorld!

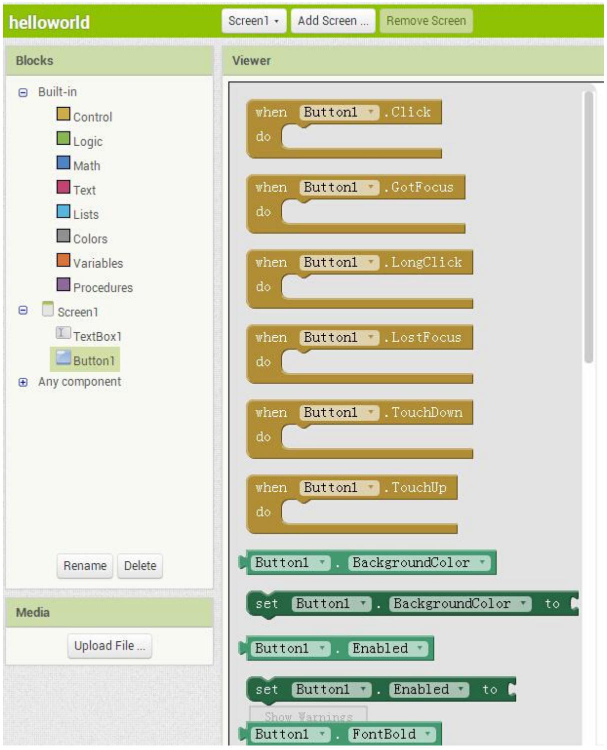
3. Drag control

   In this program, we place one textbox and one button, when button is pressed, "hello world!" is displayed in the textbox.


☐ Display hidden components in Viewer
☐ Check to see Preview on Tablet size.
9:48
Screen1
Text for Button1

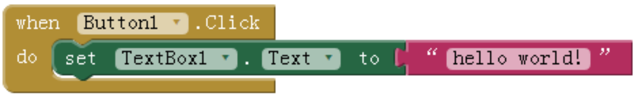
4. Coding

   Press the button "Blocks" which lay out on the right-top of the interface, and then click "Button1" in the left panel "Blocks".



```
when Button1 .Click
do
when Button1 .GotFocus
do
when Button1 .LongClick
do
when Button1 .LostFocus
do
when Button1 .TouchDown
do
when Button1 .TouchUp
do
Button1 . BackgroundColor
set Button1 . BackgroundColor to
Button1 . Enabled
set Button1 . Enabled to
Button1 . FontBold
```

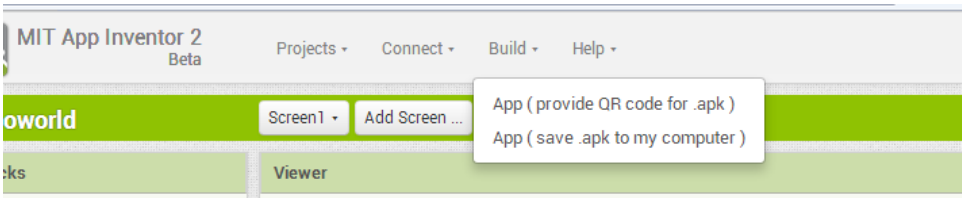
After completed the program, the code should be like this.



```
when Button1 .Click
do  set TextBox1 . Text to " hello world! "
```

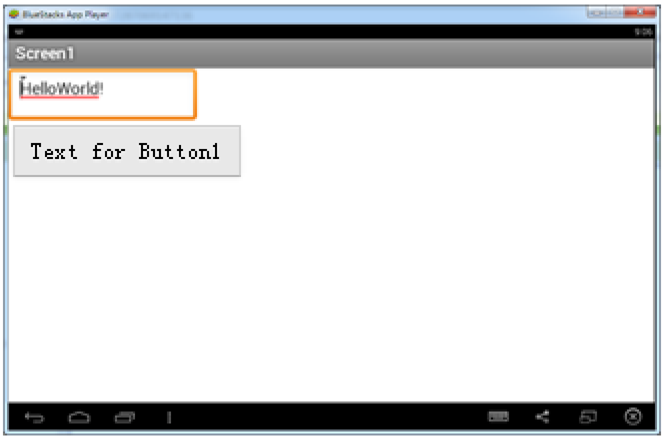
5. Build App

   Press the menu "Build"-"App(save.apk to my computer" to download the APK files.



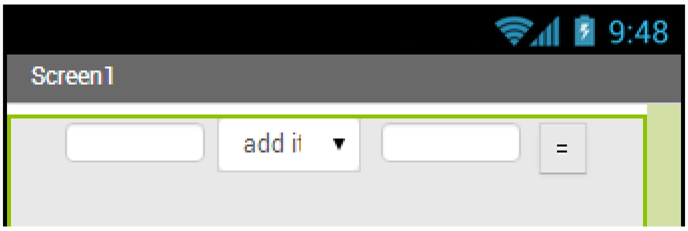
6. Run in android simulator or android phone

   The following figure shows the operating results in BlueStacks simulator.



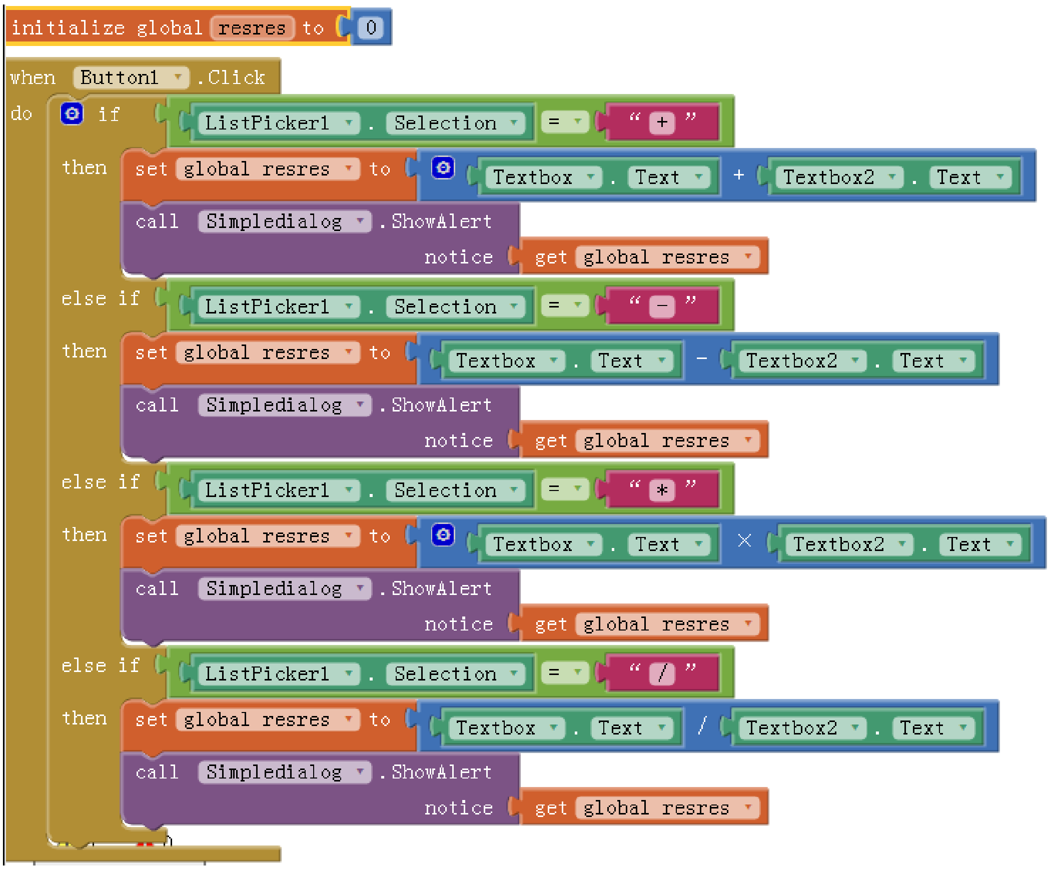
2) A simple calculator under App Inventor

1. Appearance of the calculator



2. Codes

You can follow the following flowchart to complete the task（be careful, the flowchart does not contains all the requirement of the task, you should tackle with some conditions like the error occur when dividing a number by zero）



```
initialize global resres to 0

when Button1 .Click
do  if      ListPicker1 . Selection = "+"
    then    set global resres to  Textbox . Text + Textbox2 . Text
            call Simpledialog .ShowAlert
                         notice  get global resres
    else if ListPicker1 . Selection = "-"
    then    set global resres to  Textbox . Text - Textbox2 . Text
            call Simpledialog .ShowAlert
                         notice  get global resres
    else if ListPicker1 . Selection = "*"
    then    set global resres to  Textbox . Text × Textbox2 . Text
            call Simpledialog .ShowAlert
                         notice  get global resres
    else if ListPicker1 . Selection = "/"
    then    set global resres to  Textbox . Text / Textbox2 . Text
            call Simpledialog .ShowAlert
                         notice  get global resres
```

【Assessments】

1．No lab report is needed in this task.

2．But the project file（*.aia）should be hand out to the specified place.

Lesson5 PartII:Building a Simple App（MoleMash ）

【OS & Software】

Windows XP/7, App Inventor

【Learning Outcomes】

Be familiar with MIT APP Inventor, understand the random number and clock sensors and human interface of touch screen.

【Instructions】

1) To understand the clock sensor.

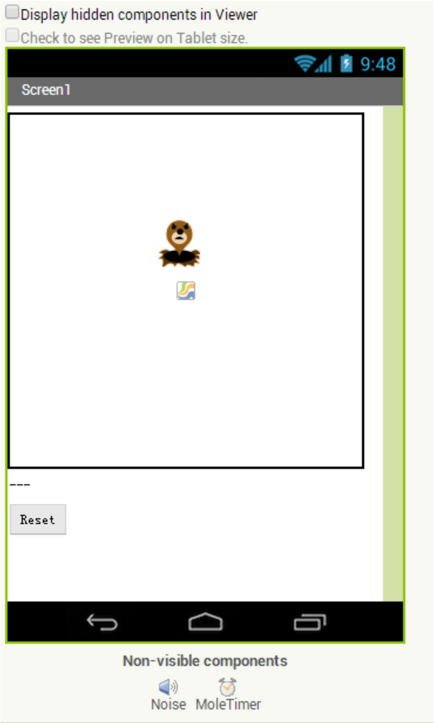
1. Be able to use clock sensor.
2. Be able to define a function to use the clock sensor.

2) Write a game named with MoleMash.

1. User can operate on the touch screen. When MoleMash have been pressed, the total score should be can automatically added.
2. The program can clear the score.

【Instructions】

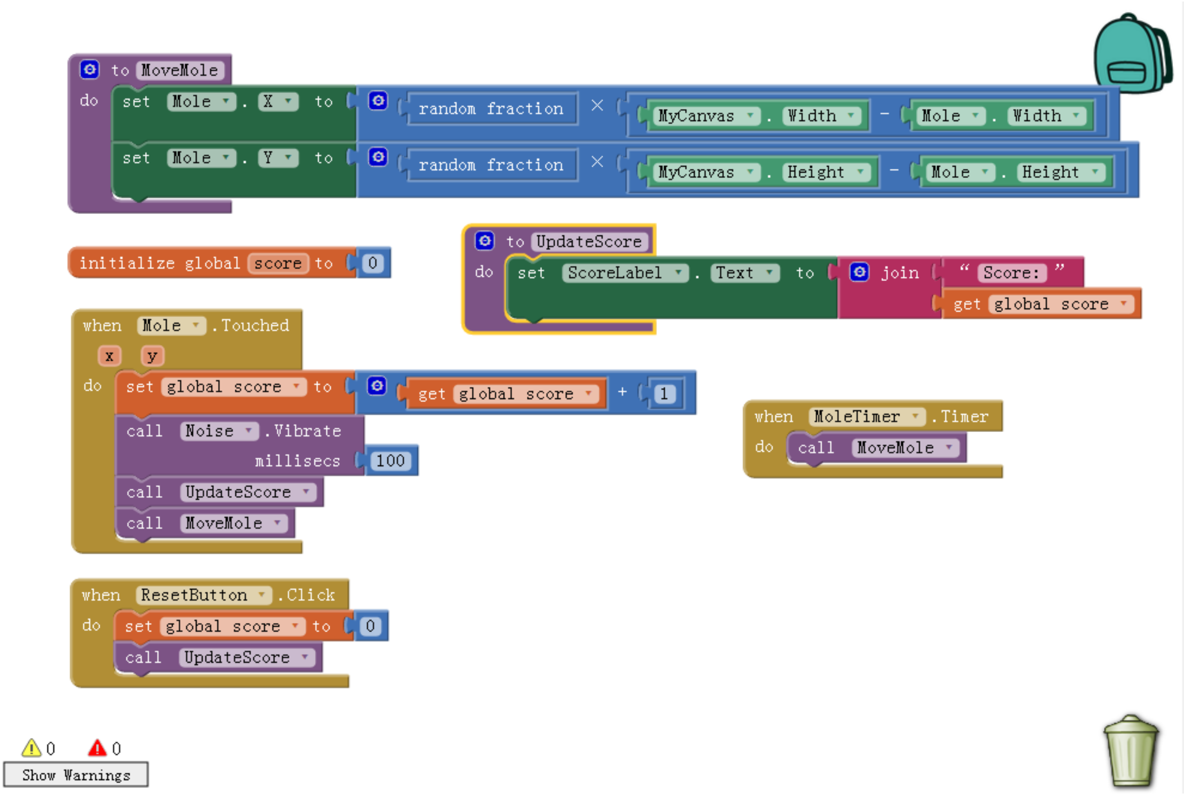
Appearance of the MoleMash game program (Be careful with the uploading way for the image of MoleMash).



□ Display hidden components in Viewer

□ Check to see Preview on Tablet size.

Codes are list in the figure.



```
to MoveMole
do  set Mole . X to  random fraction × ( MyCanvas . Width − Mole . Width )
    set Mole . Y to  random fraction × ( MyCanvas . Height − Mole . Height )

initialize global score to 0

to UpdateScore
do  set ScoreLabel . Text to  join " Score: "
                                   get global score

when Mole .Touched
  x  y
do  set global score to  get global score + 1
    call Noise .Vibrate
                millisecs 100
    call UpdateScore
    call MoveMole

when MoleTimer .Timer
do  call MoveMole

when ResetButton .Click
do  set global score to 0
    call UpdateScore
```

【Assessments】

Hand out the lab report to the specify URL.

Hand out the（*.aia）files at the same time.