Practical and Innovative Applications of IoT & IoT Networks

Smart Home

TITLE:

To Design and implement IoT system for the applications like: Home Automatiom

THEORY:

Now-a-days the world is in the search of comfort, making IoT devices is the best solution to ease the lifestyle. Like we can control home appliances, door-locks, machines using smartphones or webserver. Here, we are also doing the same using an Android app designed by MIT App Inventor.

We upload the code into NodeMCU to create a simple HTTP webserver for controlling home applicances. We will be using the HTTP GET method for communication between NodeMCU and Android APP.

Now to check whether your web server is working or not, navigate to your browser and use the following URLs to ON or OFF your light.

http://192.168.1.40/gpio/1

http://192.168.1.40:/gpio/0

Where 192.168.1.40 is NodeMCU's IP address. You can find your NodeMCU's IP address in the serial monitor. When you run the code in Arduino IDE, it will print your device's IP address on the serial monitor. Therefore, it will be confirmed that the webserver is working or not.

MIT App Inventor is an open-source web application for Android. Originally it was created by Google, but now it is maintained by the Massachusetts Institute of Technology (MIT). By using MIT app inventor a beginner can also create software applications for Android easily. MIT app inventor uses graphical interface, in which users can drag-and-drop visual objects to create an application that can run on Android devices.

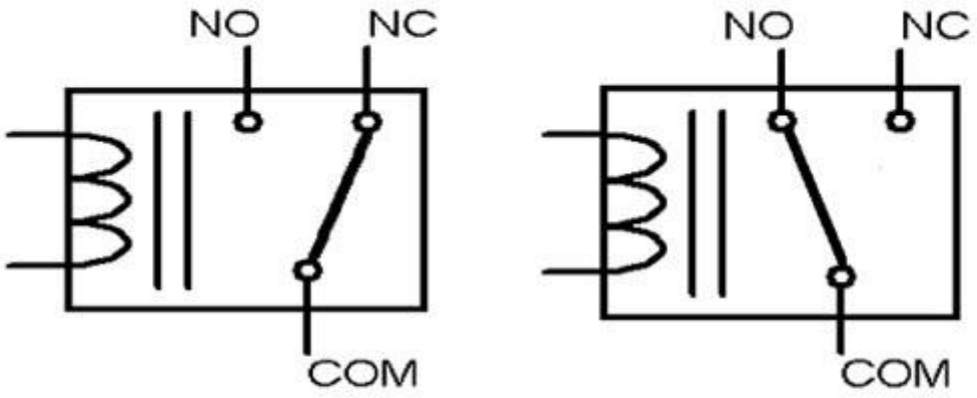
After **designing app on MIT app inventor**, you can download it on your Android phone using the QR code, or you can download its APK on your PC and later install it on your smartphone. After that, we will connect the app to **ESP8266** and control the home appliances.

MIT

How Relay works?

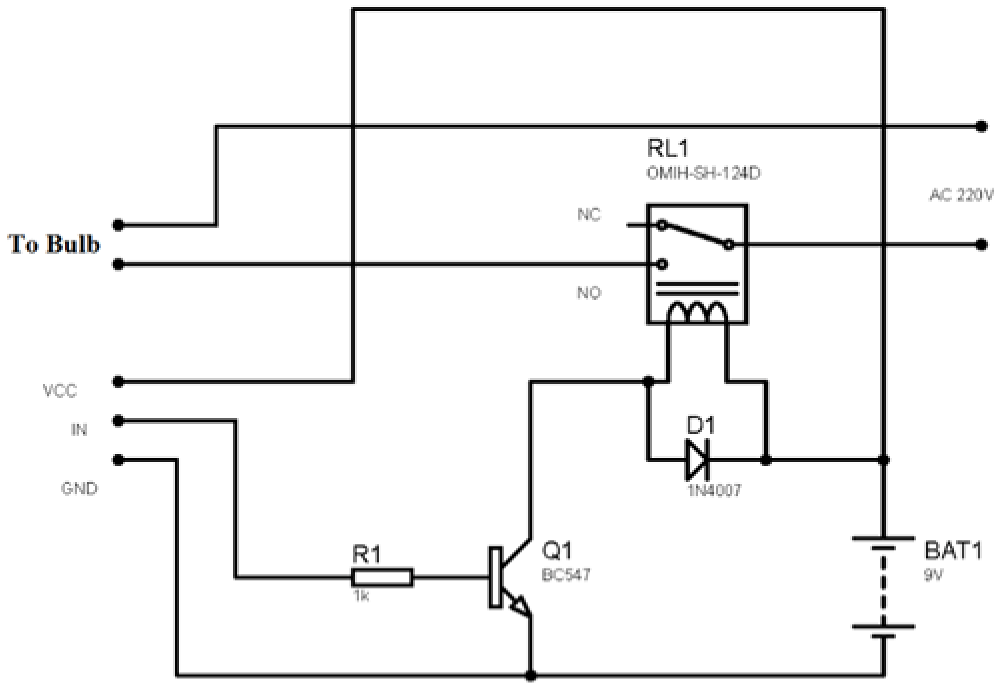
A **relay is an electrically operated switch**. Relays are used when we have to control many circuits by one signal. So by using relay, we can turn on/off a circuit electrically. Relay is controlled by small current and can switch ON and OFF larger current. Generally, relay has five terminals as shown below:

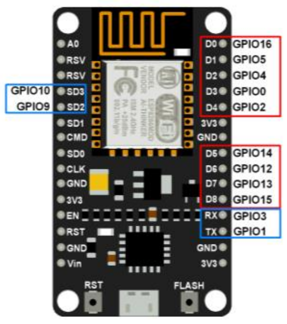
When no voltage applied to the coil, COM terminal will be connected to NC (normally closed) terminal. And when the voltage is applied to the coil, electromagnetic field produced that attract the Armature, and COM and NO (normally open) terminal gets connected.



Relay Driver:

A small Driver circuit that consists of a Transistor, Diode and a resistor are used to configure the relay. Transistor is used to amplify the current, Resistor is used to provide bias to the transistor, and in the case when the transistor is Off, Diode is used to prevent reverse current flow, here we have used 5V Relay module.





COMPONENTS REQUIRED:

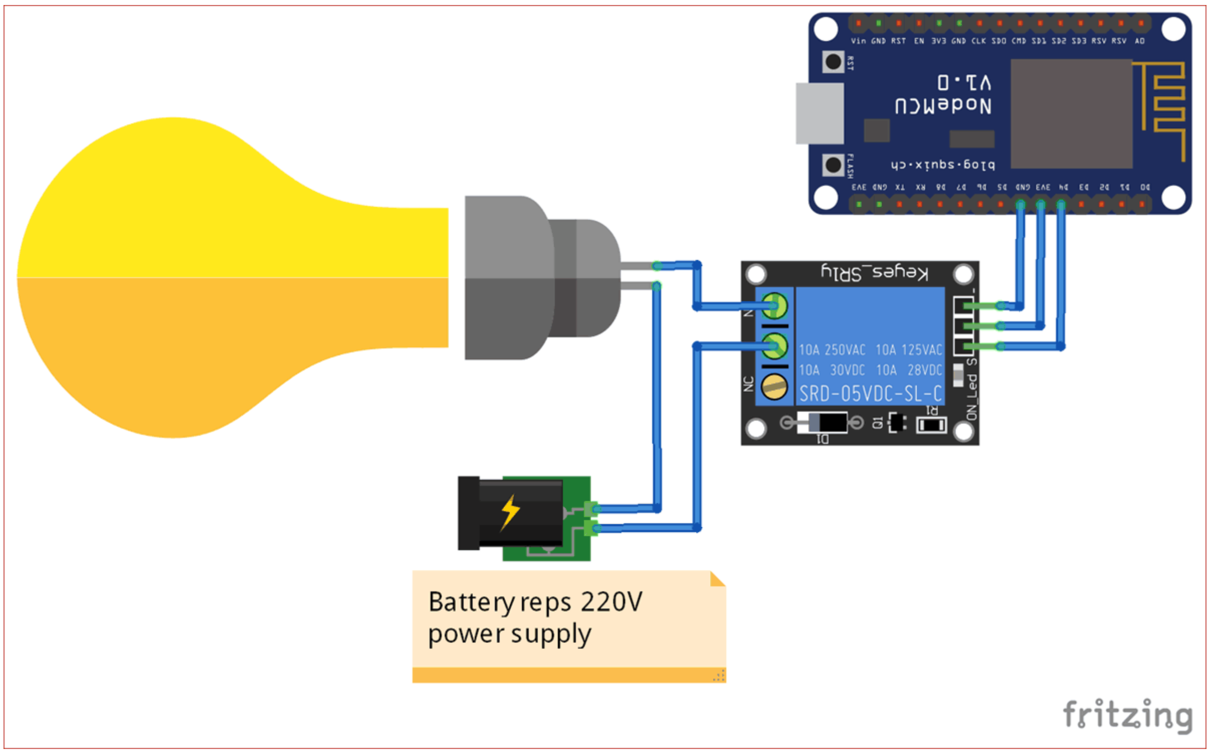
- ESP8266 (NodeMCU)
- Lamp
- 5V Relay
- Breadboard
- Connecting Wires

Circuit Diagram

In this experiment, we are going to log and monitor data over internet using MIT app Inventer IoT server. And we can view the logged data over time on Mobile app. System is made using ESP8266 WiFi module and Relay module. ESP8266 WiFi chip reads the current date and sends it to MIT app inventer server for live monitoring.

In this experiment, relay module can be used with esp8266 or Nodemcu and program from Arduino IDE, the mobile data has used to send data to MIT app and it was really productive, results are quite good.

Connection:

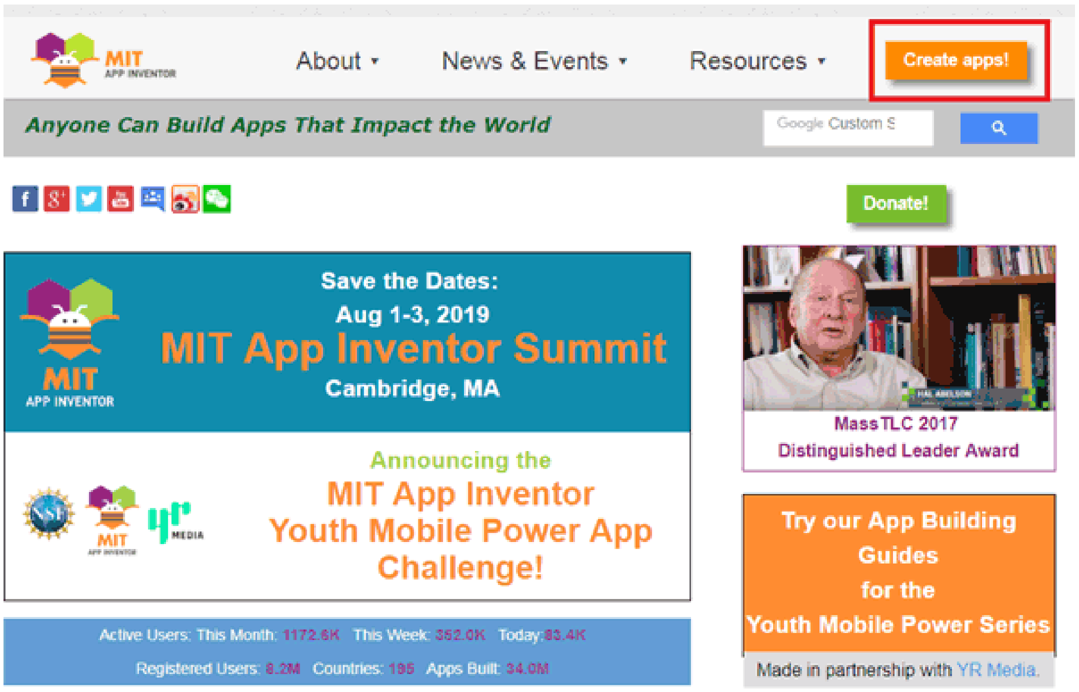


Create an Android APP using MIT App Inventor

Now we will create an android app using MIT app Inventor to control light using the following steps:

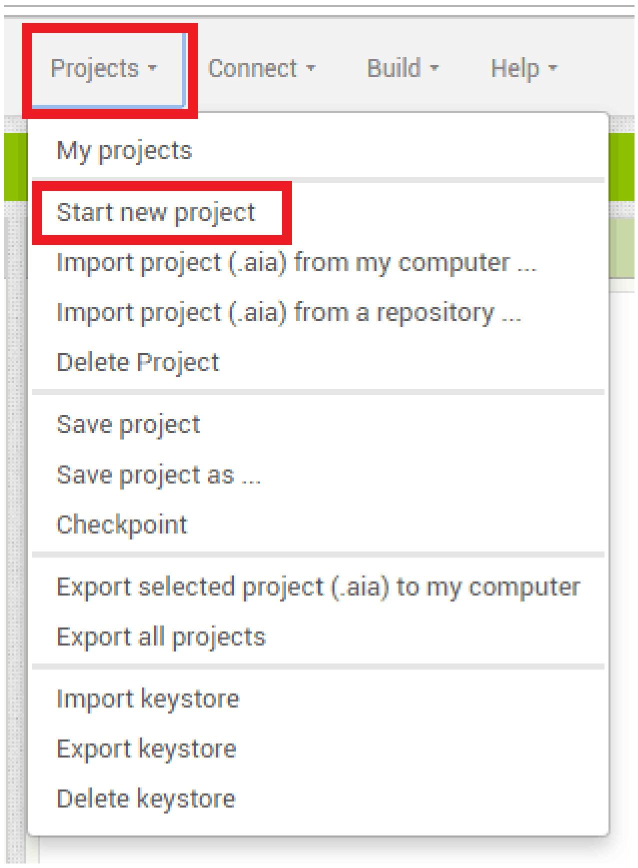
First of all go to the MIT app inventor's website: http://ai2.appinventor.mit.edu/

Then click on *'Create apps'* on the top right corner.


MIT APP INVENTOR
About
News & Events
Resources
Create apps!
Anyone Can Build Apps That Impact the World
Google Custom S
Donate!
Save the Dates:
Aug 1-3, 2019
MIT App Inventor Summit
Cambridge, MA
Announcing the
MIT App Inventor
Youth Mobile Power App Challenge!
MassTLC 2017
Distinguished Leader Award
Try our App Building Guides for the Youth Mobile Power Series
Made in partnership with YR Media.
Active Users: This Month: 1172.6K This Week: 352.0K Today: 83.4K
Registered Users: 8.2M Countries: 195 Apps Built: 34.0M

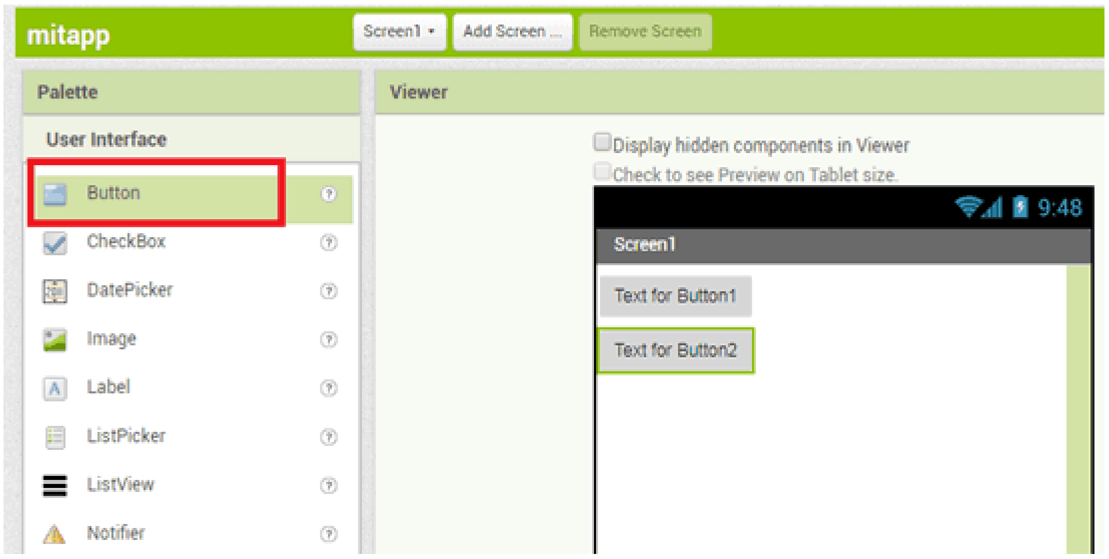
Create an Android APP using MIT App Inventor

Now in the next screen click on *'Projects'* and then *'Start new project'.*



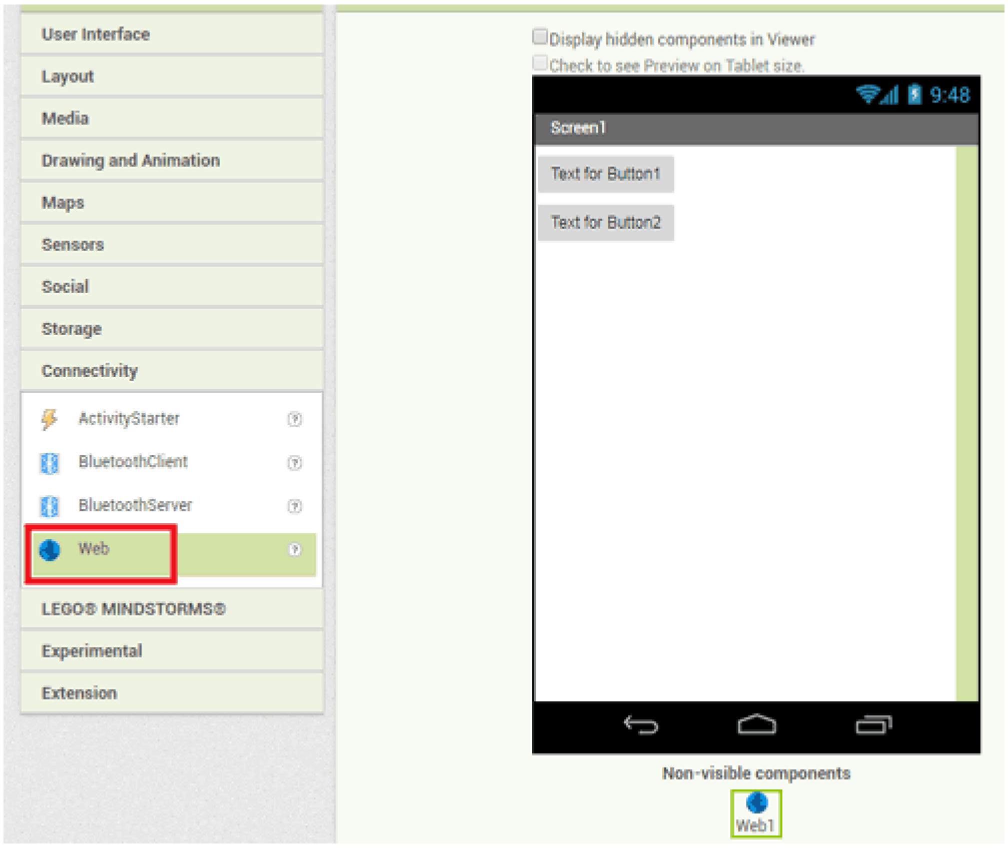
Create an Android APP using MIT App Inventor

Now click on *'Button'* and drag and drop two buttons on the main screen. You can enter the name of your choice on the buttons from the options on the right side.



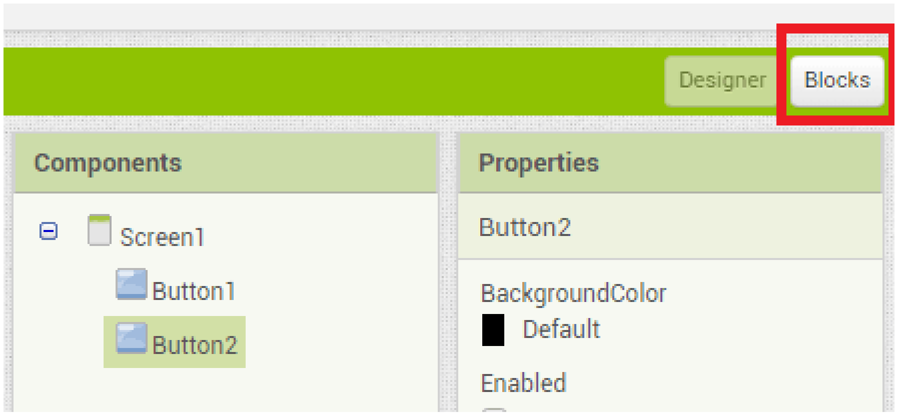
Create an Android APP using MIT App Inventor

After this click on the *'connectivity'* and drag and drop the web component on the main screen.



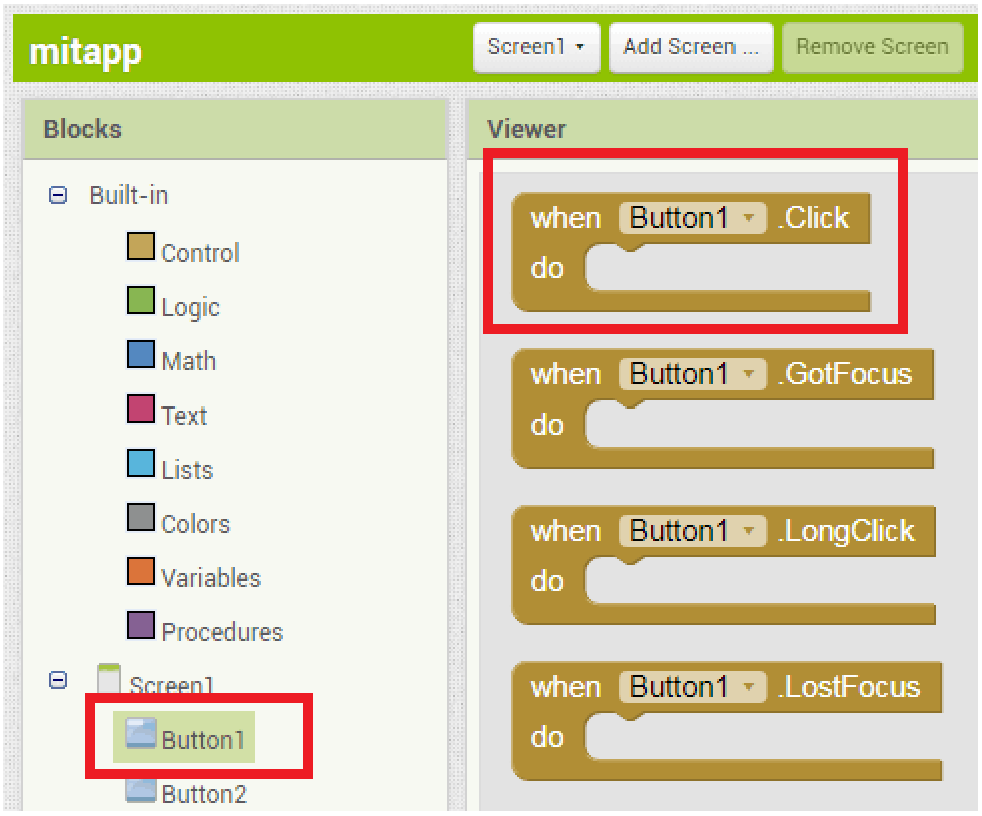
Create an Android APP using MIT App Inventor

Now click on the *'Blocks'* to add blocks in your app.



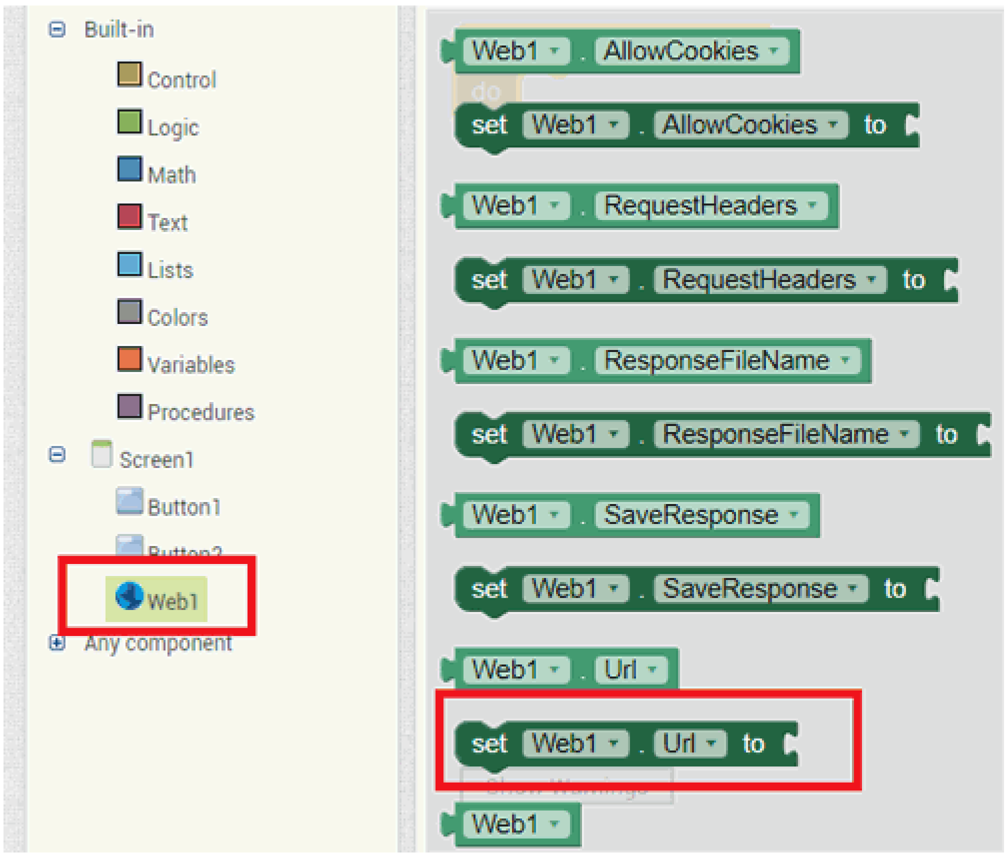
Create an Android APP using MIT App Inventor

Now in blocks menu click on the button1 and then click on the marked red option.



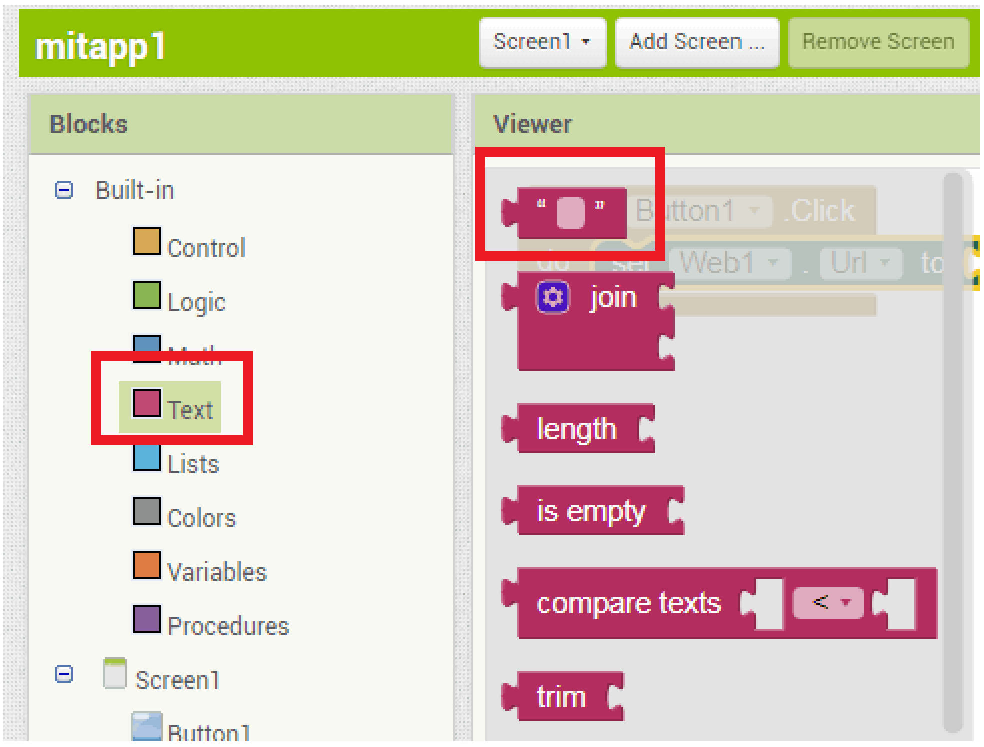
Create an Android APP using MIT App Inventor

After this click on web1. Scroll down and choose the red marked block.



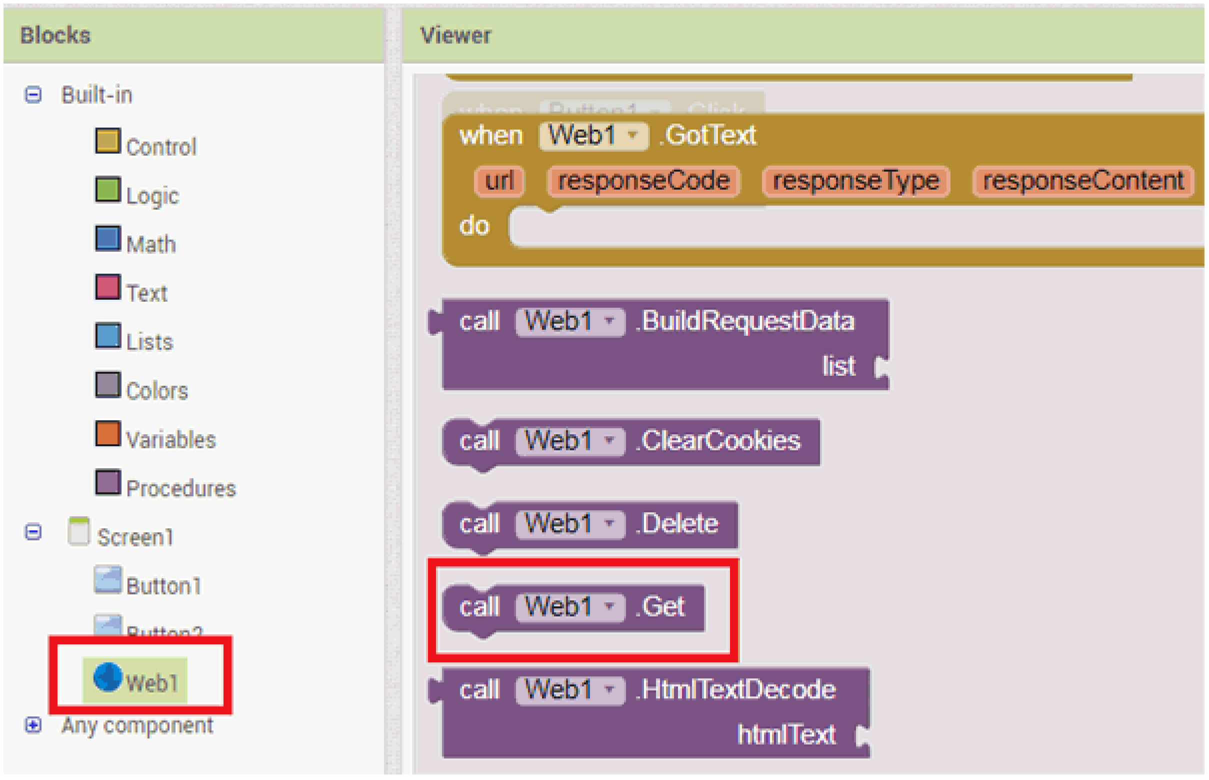
Create an Android APP using MIT App Inventor

Now click on the text menu and choose the first option. Enter your URL in the text menu.



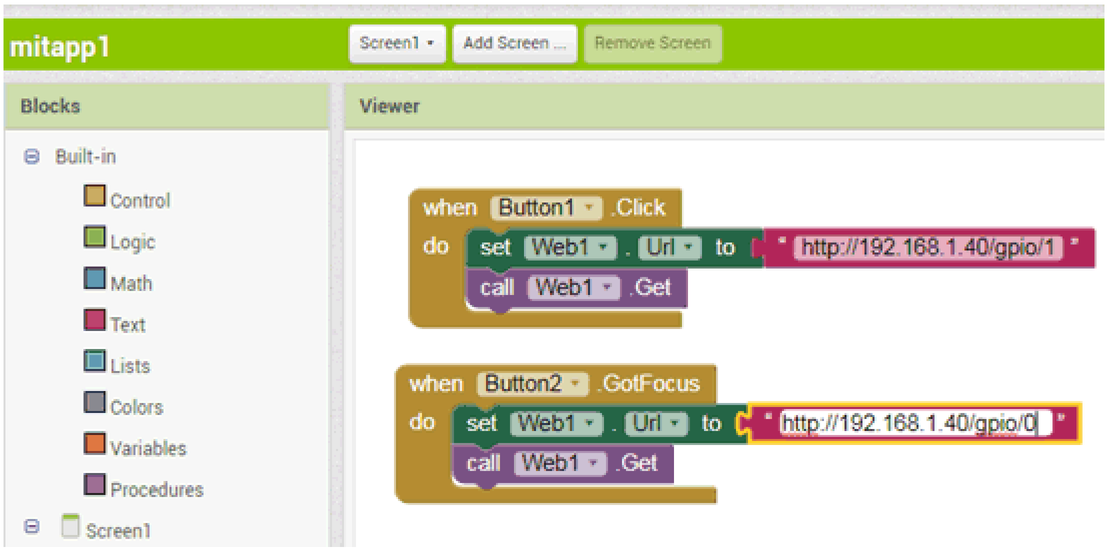
Create an Android APP using MIT App Inventor

After this click on the web1 again and then choose the marked red option.



Create an Android APP using MIT App Inventor

Follow the same procedure for the *'Button2'*.

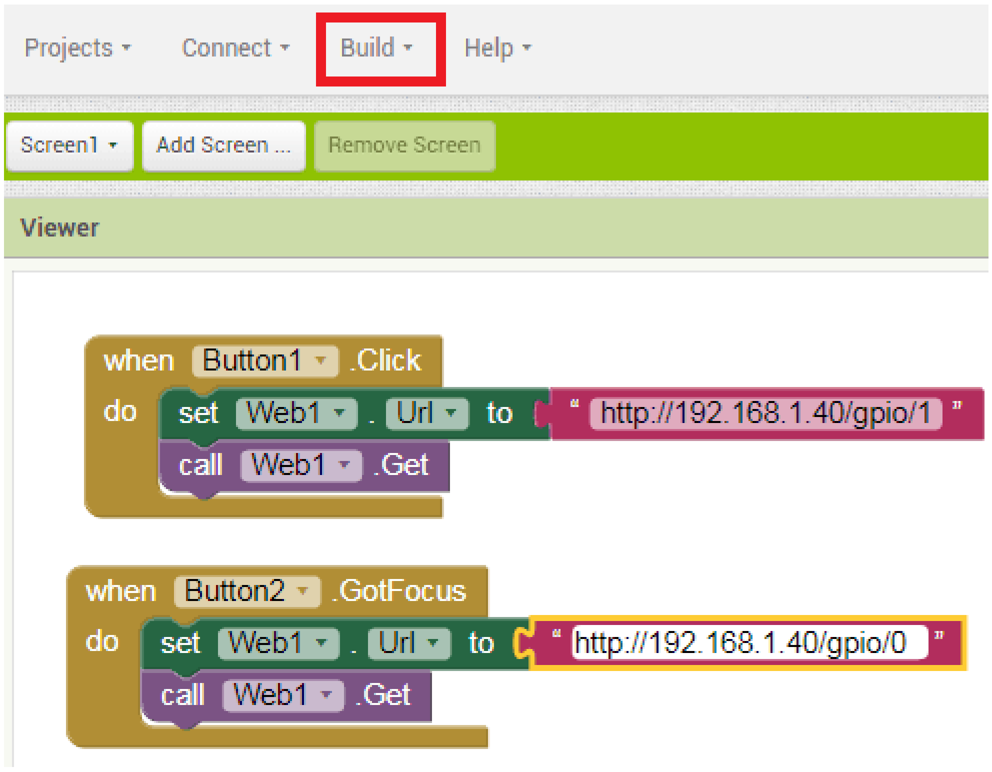


```
when Button1 .Click
do  set Web1 . Url to " http://192.168.1.40/gpio/1 "
    call Web1 .Get

when Button2 .GotFocus
do  set Web1 . Url to " http://192.168.1.40/gpio/0 "
    call Web1 .Get
```

Create an Android APP using MIT App Inventor

Now the app is ready to download, simply click on 'Build' to get he apk file . Also, there are two options to download the app apk, by the QR code and directly on PC, then later install it on the Android.

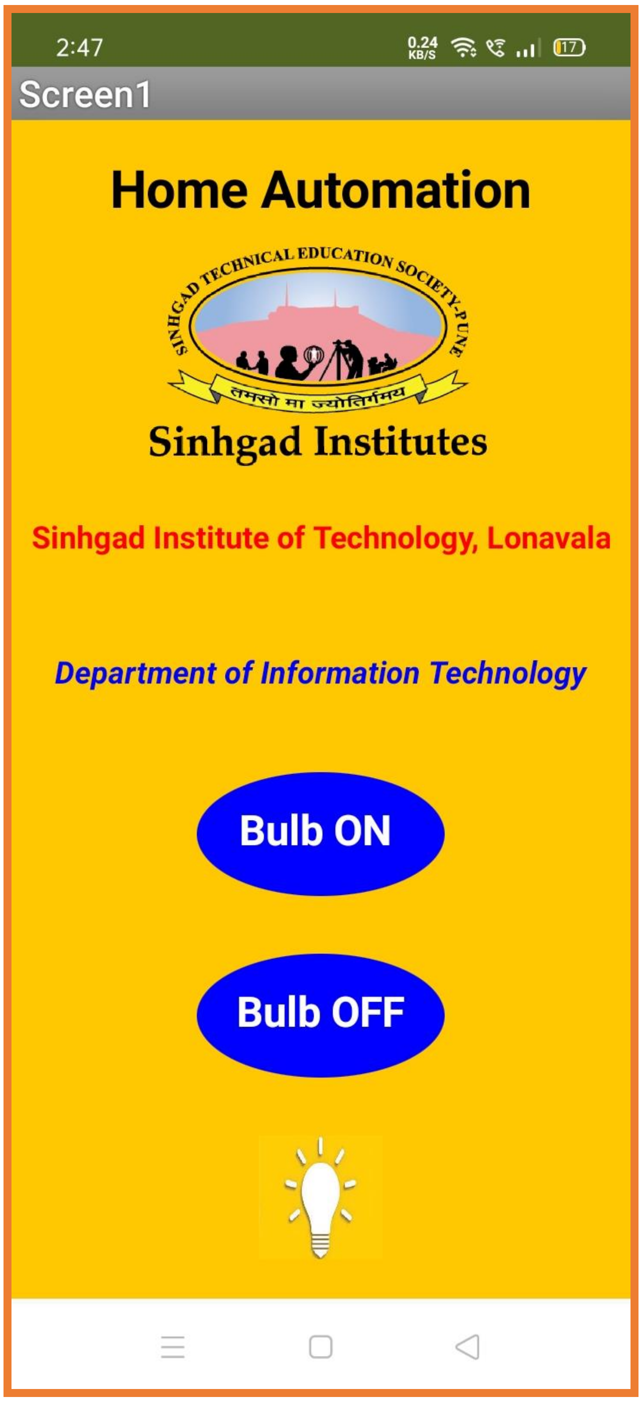


```
when Button1 .Click
do  set Web1 . Url to " http://192.168.1.40/gpio/1 "
    call Web1 .Get

when Button2 .GotFocus
do  set Web1 . Url to " http://192.168.1.40/gpio/0 "
    call Web1 .Get
```

Create an Android APP using MIT App Inventor

Now your app is ready, and you can control the light using the on-off buttons presented on the app.


2:47
Screen1
Home Automation
SINHGAD TECHNICAL EDUCATION SOCIETY-PUNE
तमसो मा ज्योतिर्गमय
Sinhgad Institutes
Sinhgad Institute of Technology, Lonavala
Department of Information Technology
Bulb ON
Bulb OFF

Program:

```
#include <ESP8266WiFi.h>
 const char* ssid = "OPPO_pcl";
const char* password = "pclatane123";
WiFiServer server(80);
 void setup() {
  Serial.begin(115200); //Default Baud Rate for NodeMCU
  delay(10);
  pinMode(2, OUTPUT);  // Connect Relay to NodeMCU's D4 Pin
  digitalWrite(2, 0);
  // Connect to WiFi network
  Serial.println();
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
  delay(500);
  Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");
  // Start the server
  server.begin();
  Serial.println("Server started");
  // Print the IP address
  Serial.println(WiFi.localIP());
}
 void loop() {
  // Check if a client has connected
  WiFiClient client = server.available();
  if (!client) {
    return;
  }
  // Wait until the client sends some data
  Serial.println("new client");
  while(!client.available()){
  delay(1);
  }
  String req = client.readStringUntil('\r');
  Serial.println(req);
  client.flush();
int val;
  if (req.indexOf("/gpio/0") != -1)
    val = 0;
  else if (req.indexOf("/gpio/1") != -1)
    val = 1;
  else {
    Serial.println("invalid request");
    client.stop();
    return;
  }
  String s = "HTTP/1.1 200 OK\r\nContent-Type:
 text/html\r\n\r\n<!DOCTYPE HTML>\r\n<html>\r\nGPIO is now ";
  s += (val)?"high":"low";
  s += "</html>\n";
// Send the response to the client
client.print(s);
delay(1);
Serial.println("Client disonnected");
}
```

CONCLUSION:

After the study of this assignment, we are familiar with the home automation and use of MIT app Inventer to upload data on cloud.