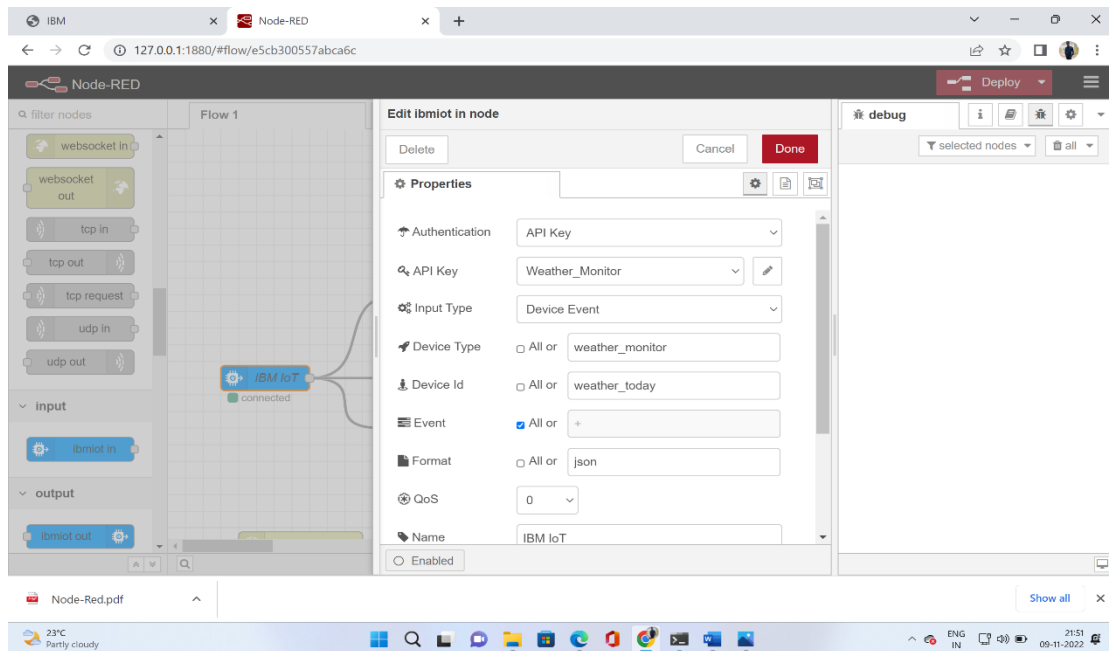


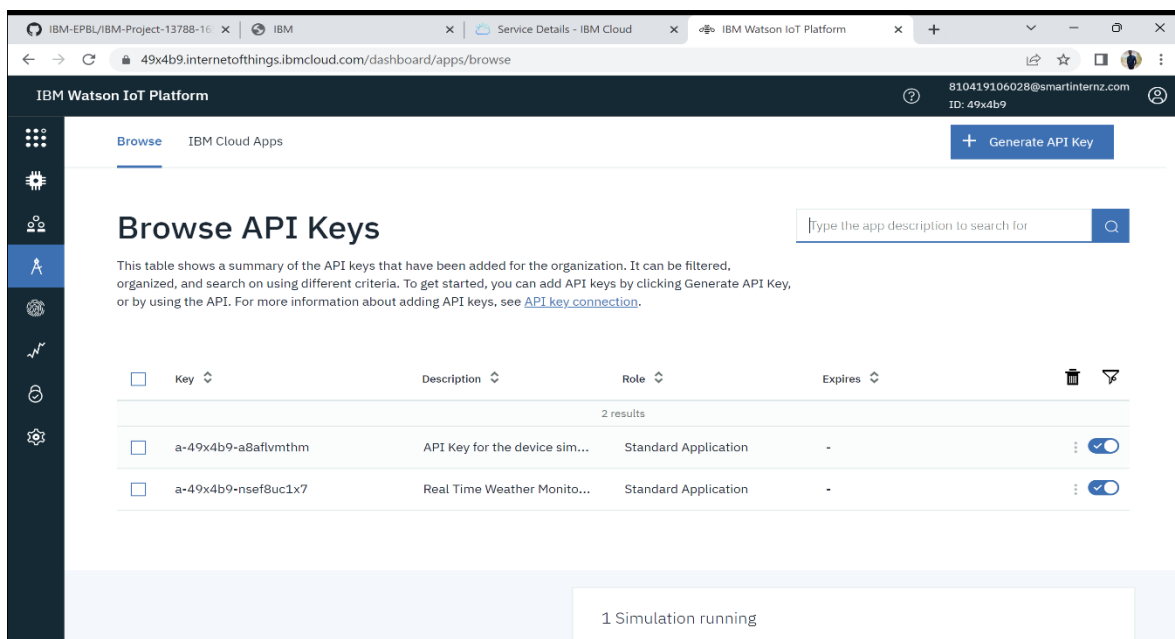
Build A Web Application Using Node-Red

Team ID	PNT2022TMID08469
Project Name	Smart Farmer-IOT Enabled Smart Farming Application

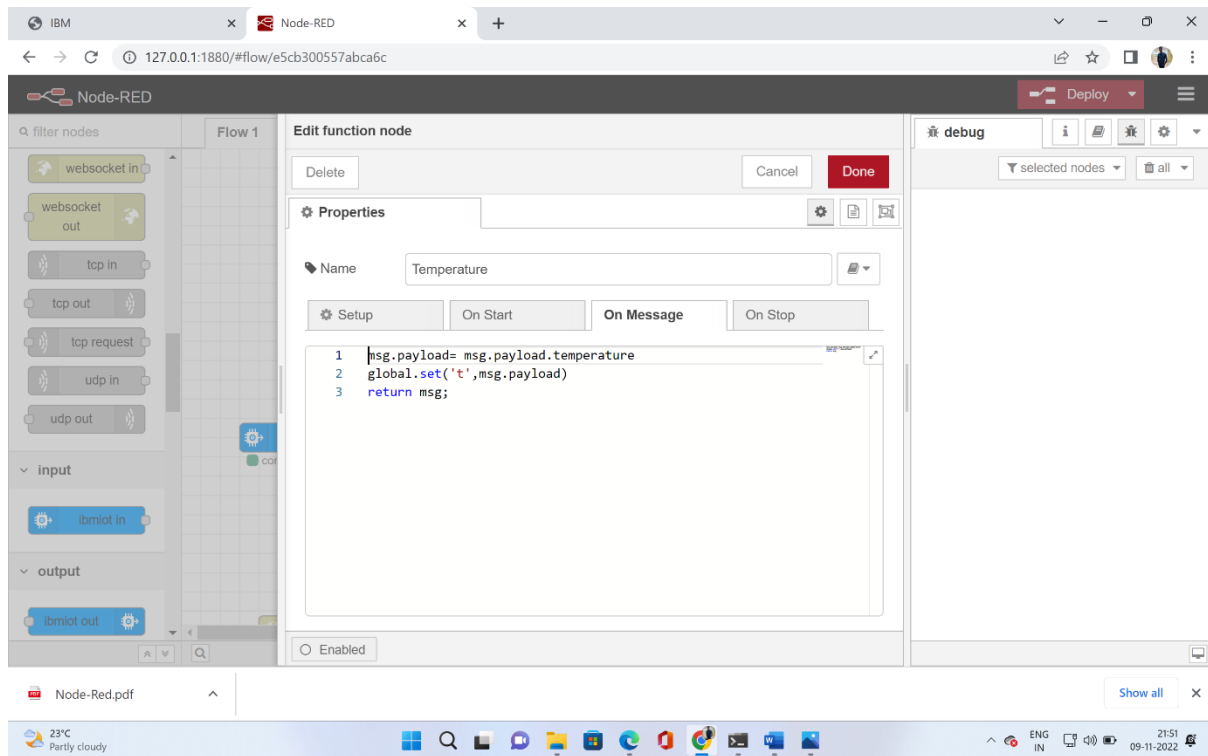
First open Node RED workspace and drag IBM iot input into the workspace. It will ask API key, device id ,device type etc.



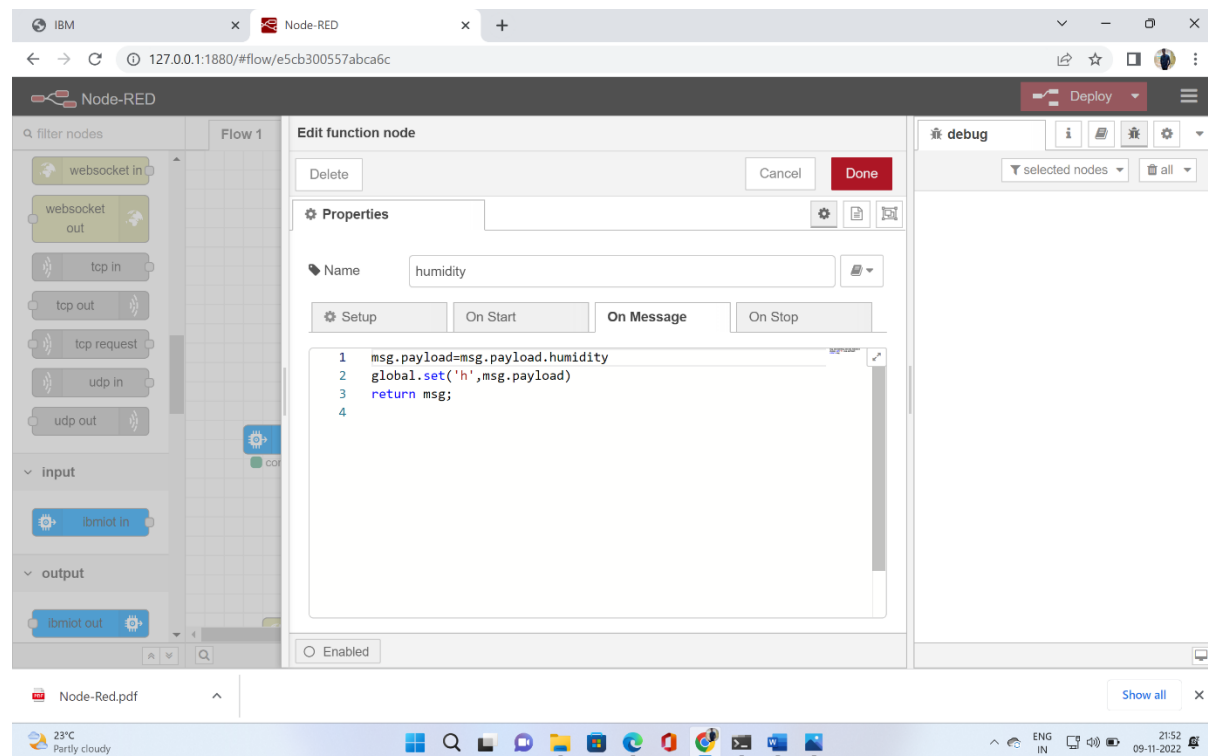
For Generating API keys open IBM Watson and click Apps and Generate API key it shown below like these:



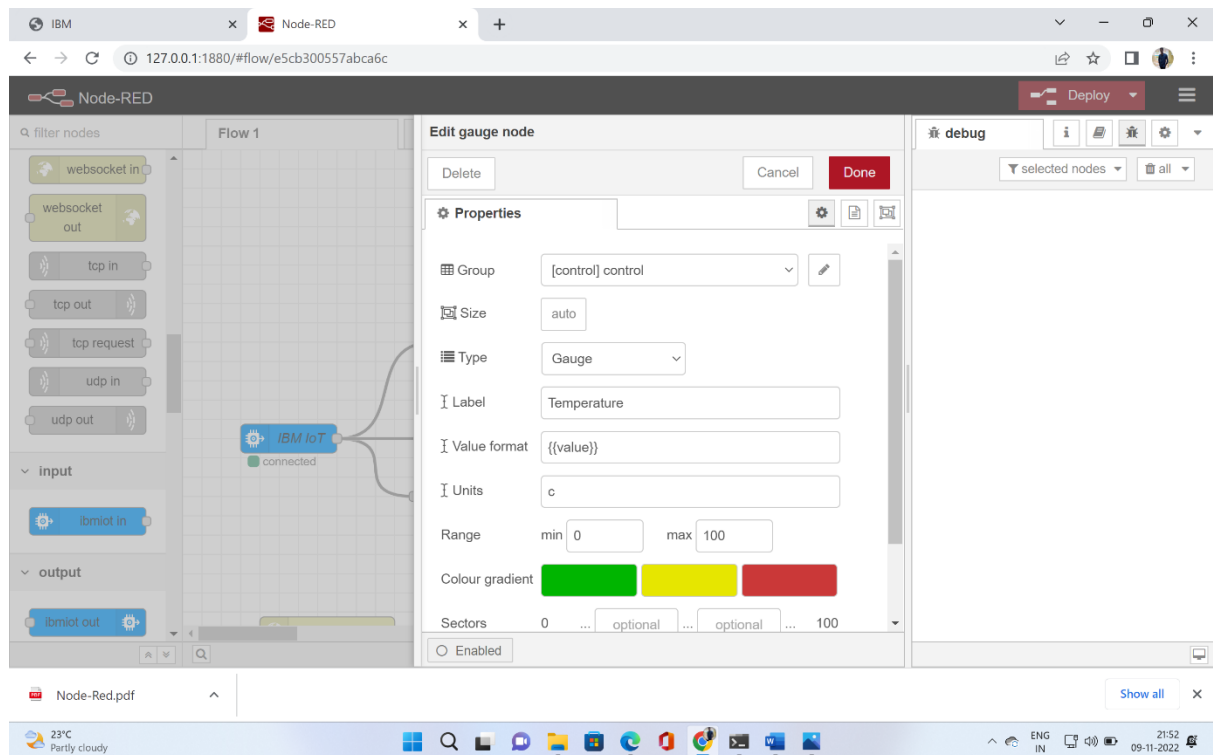
And take a function node and rename it has a temperature and message in the editor.



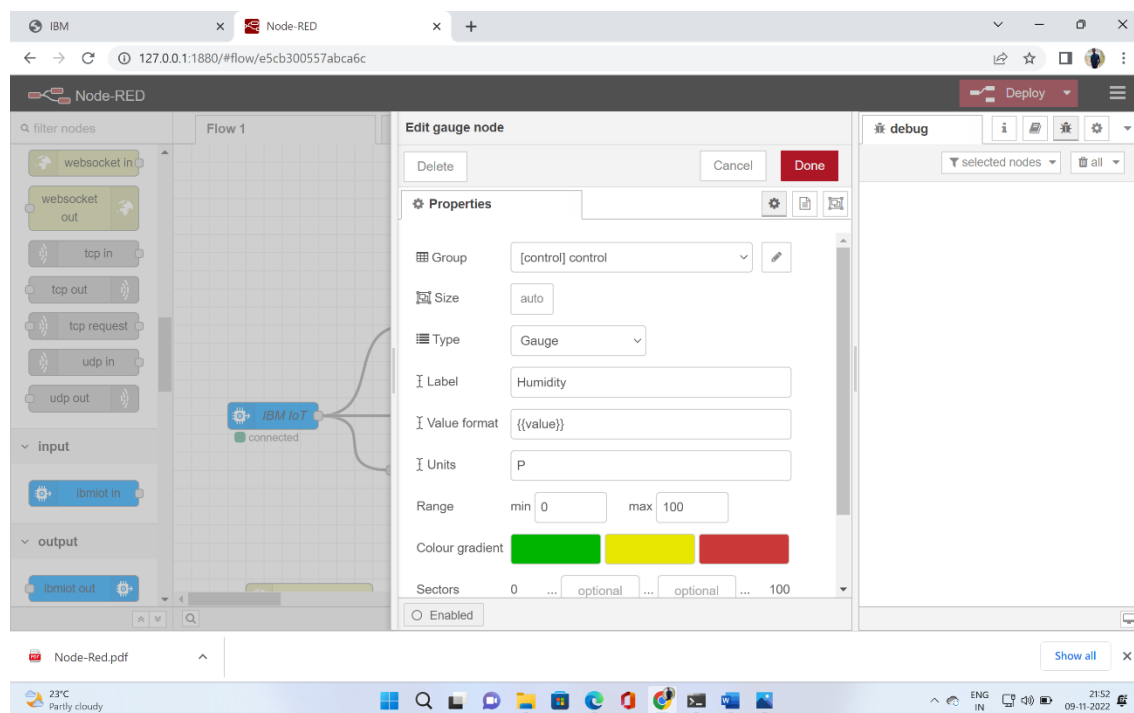
Smiliarly take a function node for humidity and type message in the editor.



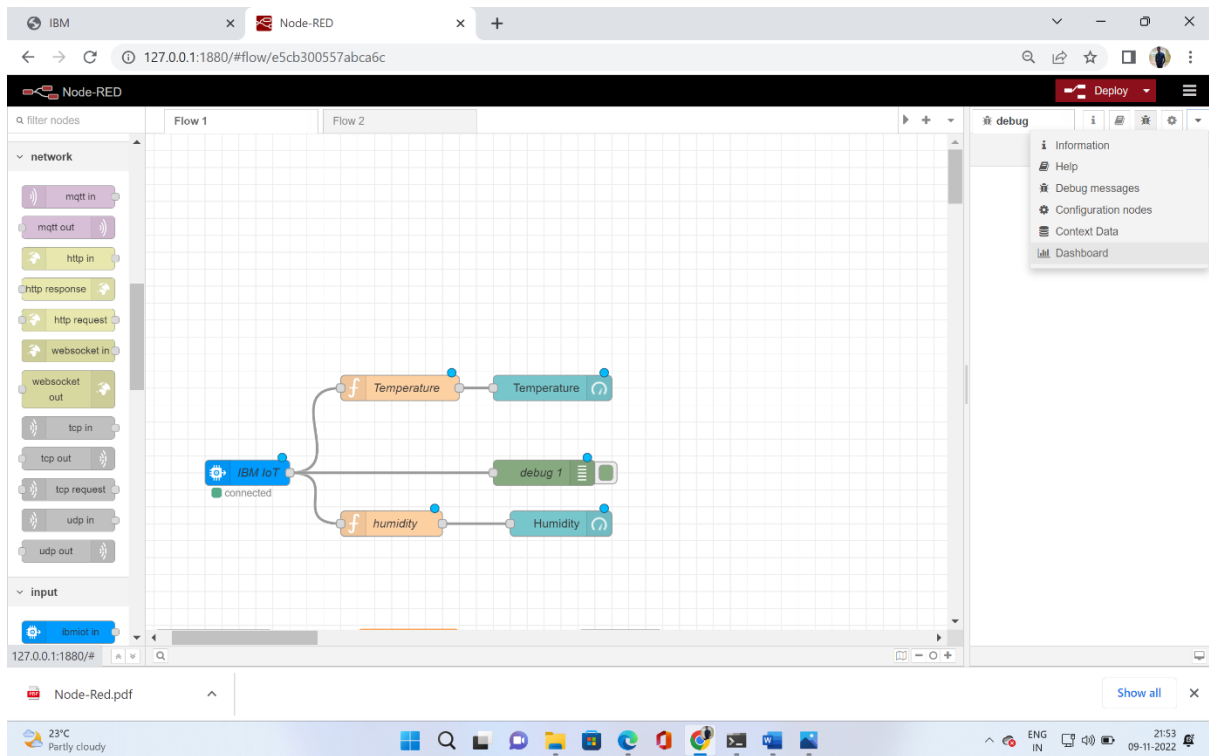
Now take temperature gauge meter in the dashboard and give name as temperature and range 0 to 100.



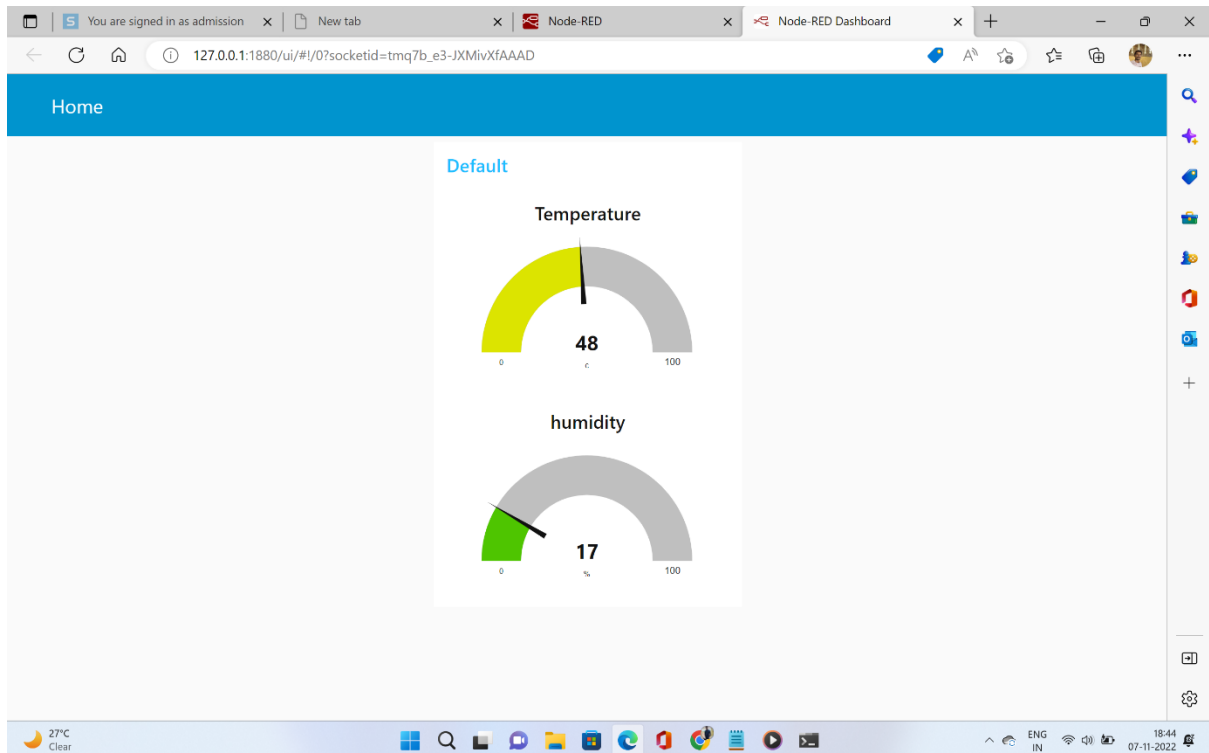
Smiliarly for humidity u take another gauge meter and range 0 to 100



Finally we can connected as shown below:



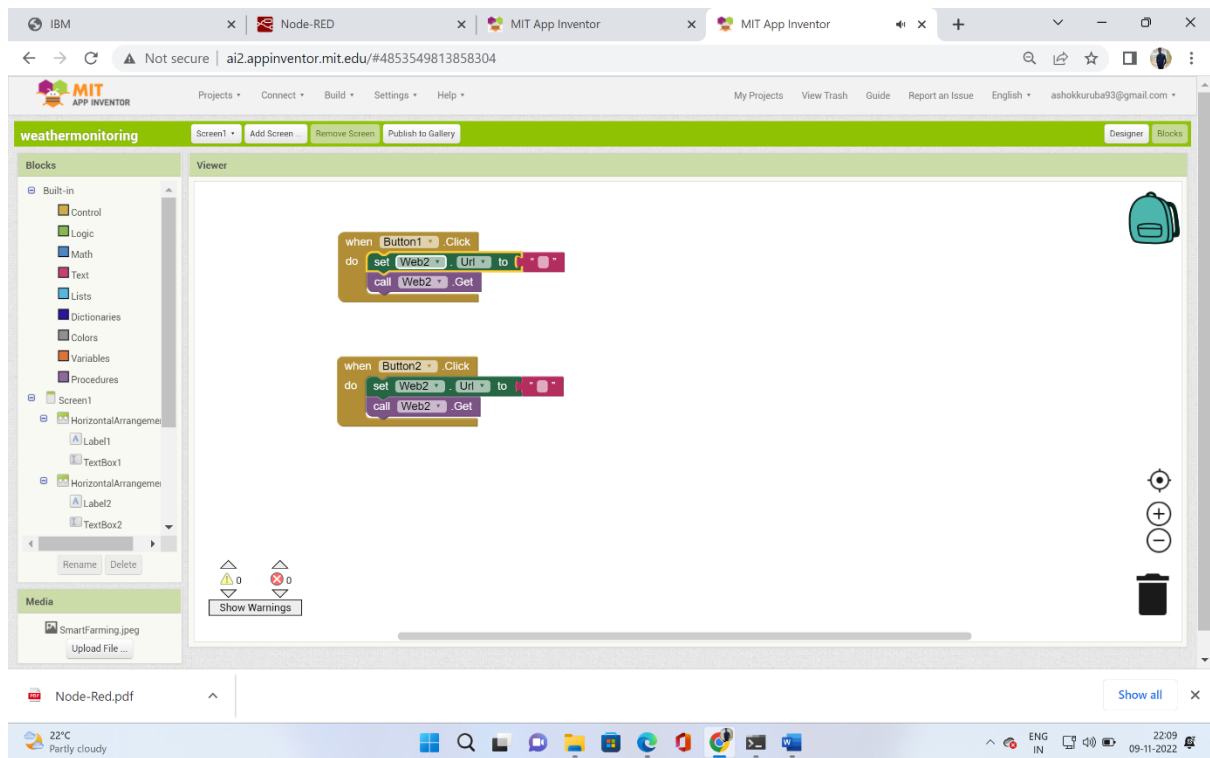
The Gauge metre is showing reading below like these:



The screenshot displays the MIT App Inventor web application. The browser's address bar contains the URL `ai2.appinventor.mitedu/#6285081757745152`. The page header includes navigation links like 'Projects', 'Connect', 'Build', 'Settings', and 'Help'. The main interface is divided into a left sidebar and a central workspace. The sidebar has a 'Blocks' section with categories like 'Built-in' (Control, Logic, Math, Text, Lists, Dictionaries, Colors, Variables, Procedures) and 'Screen1' (Web1, Clock1, Web2). Below this is a 'Media' section with an 'Upload File...' button. The central workspace, titled 'Viewer', shows a visual programming block: 'when Clock1.Timer do set Web1.Uri to "" call Web1.Get'. The bottom of the browser window shows a taskbar with various application icons and a system tray indicating the time as 22:02 on 09-11-2022.

The screenshot displays the MIT App Inventor web application. The top navigation bar includes links for 'Projects', 'Connect', 'Build', 'Settings', and 'Help'. The main workspace is titled 'weathermonitoring' and shows a visual programming canvas with two event-driven code blocks. The first block, 'when Clock1 ticks', sets a variable 'Web1' to 'url' and calls 'Web1 Get'. The second block, 'when Web1 GET text', contains logic for parsing JSON data from a weather API, including error handling for 'not found' responses. The left sidebar shows 'Blocks' and 'Media' sections, and the bottom of the screen shows a Windows taskbar with various application icons and system status.

Now take the button-1,button-2, and copy their links from nodered and paste in the web application:



Now we successfully connected the nodered with the Mobile application.