



College of Science & Engineering

COMP3033/COMP9033 Cloud and Distributed Computing

Laboratory Week 9

Learning outcomes for this session

This week, we will continue working with Node-RED and the Watsonx Assistant service. The learning outcomes of this week are technology building, technology resources, and service management.

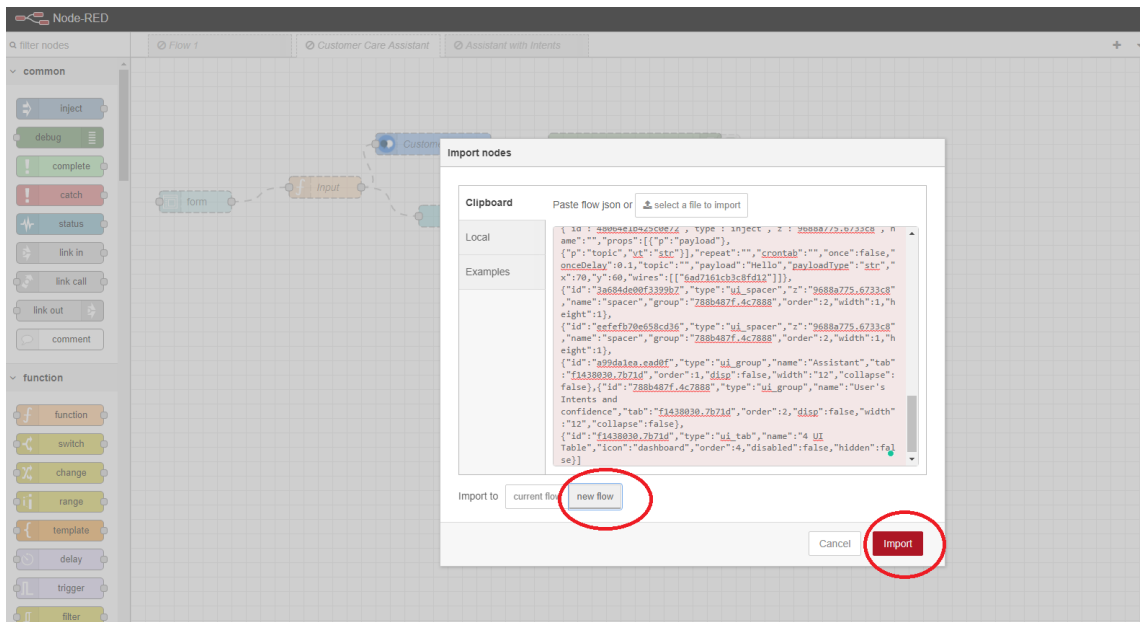
Notes.

- (1) When you “Deploy” your Node-RED application, make sure that other tabs are “disabled” and only the current (active) tab is enabled. Otherwise, all enabled tabs will be deployed together into a single run-time.
- (2) Make sure you backup your existing Node-RED flows by exporting them to the clipboard, pasting the text into a simple text editor and saving it as a JSON file.

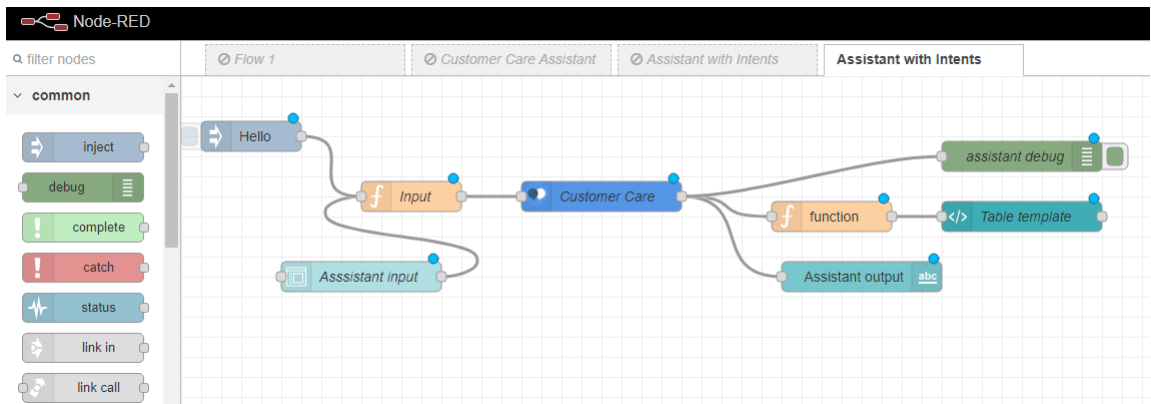
TASK 1. Node-RED Assistant with Intents flow

In the Assistant with Intents flow, you will learn how to include matched “Intent” returned by the “Watsonx Assistant” service in the user interface for testing.

- In the following steps, you will import, configure and deploy a Node-RED flow that uses a “Watsonx Assistant” service with Intents.
- Open “CustomerCareAssistant2.json” from Week 9 on the FLO page. Select and copy all JSON code.
- Open the top Menu, and click on “Import”.
- “Import Nodes” window will be opened. Paste copied JSON code from “CustomerCareAssistant2.json”. Select “new flow” and click on the “Import” button.



- Now navigate to the newly created “Assistant with Intents” tab and ensure it is enabled and all other tabs are disabled. Your newly imported flow should contain all nodes as shown below.



- Double click on the “Customer Care” node, which is “Watson Assistant” node type.
- When the “Edit Assistant” window is displayed (see the screenshot below).

Edit assistant V2 node

Delete Cancel Done

Properties

Name Customer Care

Username Username

Password Password

API Key API Key

Service Endpoint <https://api.eu-gb.assistant.watson.cloud.ibm.com>

Assistant ID Assistant ID

Timeout Period Leave empty to disable

☒ Switch on Debug

☐ Restart Dialog

☒ Return Context

☐ Return Alternate Intents

☒ Multiple Sessions

☐ Opt Out Request Logging

☐ Enabled

- (1) For Username, provide your IBM Cloud user id (your Flinders University email).
- (2) For Password, provide your IBM Cloud password.
- (3) Now paste the “API key” credentials, similar to what you did in our workshop last week.
- (4) Paste the service endpoint, similar to what you did in our workshop last week.
- (5) Paste the “Assistant ID”, similar to what you did in our workshop last week.
- (6) Leave other default settings unchanged.
- (7) Click on “Done”.

- Deploy your application to make sure it runs without errors.

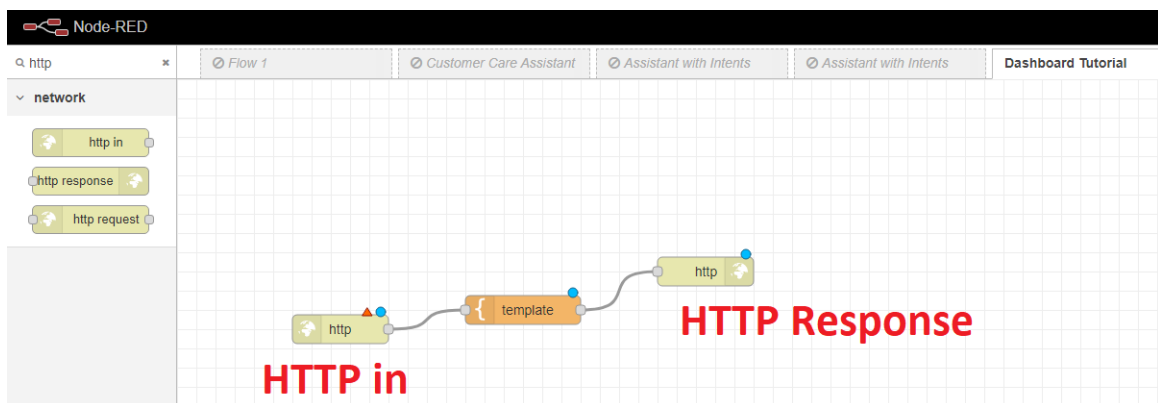
- **Test your application “Assistant with Intents” Node-RED flow.**

- (1) Copy the URL (web address) of your currently displayed Node-RED editor.
- (2) Open a new Browser Tab (or new Browser window) and paste the copied URL.
- (3) From the copied URL remove all characters after the IP address of your Node-RED and add **/ui** to the end of the URL (without quotes). The **ui** addition to the URL stands for “user interface”.
- (4) For example, my Node-RED editor flow URL is: **http://34.87.244.112/#flow/d21f2783.2bd5b8** and my (Assistant with Intents) application URL is: **http://34.87.244.112/ui/**
- (5) In your new browser tab a web form mimicking a chat-bot interface will be shown. Start testing by entering “Help” in the input, as shown below.

- (6) After you submit your “Help” input it will be sent to the “Watson Assistant” service (a REST API call via the “Service Endpoint” which is routed via “Watson Assistant URL”). The service will authenticate you by using the API key. It will also recognise your “Assistant ID”. The engine will flag “Help” input as “#Help” intent which has rule in the dialog to send “Help”. The bot will generate a response “How can I help?”.
 - (7) In addition “Help” intent will be shown with the engine’s confidence of 1 (100%) on mapping “Help” to the “Help” intent.
 - (8) Continue creating new intents and testing with various inputs such as “What are the business hours?”, “I need an appointment”, “can you assist me”, etc.
- After you complete testing go back to Node-RED editor browser tab. Export the “Assistant with Intents” flow and save it as a JSON file.

TASK 2. Node-Red Dashboard Tutorial.

- Open a new flow page by clicking on the “+” symbol on the right-hand-side of the screen, and name it “Dashboard Tutorial”.
- Before we get into the node-red dashboard module, let us can create custom web pages.
- We create an endpoint by dragging and dropping an “HTTP in” node. We also drag and drop a “template” and an “HTTP response” node. And we wire them together.



- Configure the node settings as follows:

Edit http in node

Delete Cancel Done

Properties

Method GET

URL /test1

Name Name

Edit template node

Delete Cancel Done

Properties

Name Name

Property msg. payload

Template Syntax Highlight: mustache

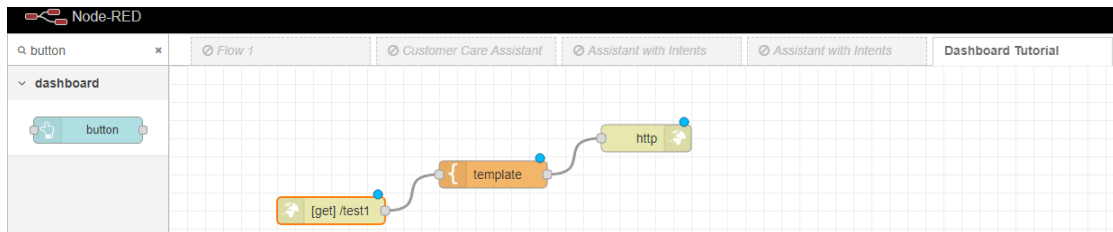
```
1 <html>
2   <head>
3     <title>Node-RED costume Site</title>
4   </head>
5   <body>
6     <h1>Hello World!</h1>
7     <p>Your text here!</p>
8   </body>
9 </html>
```

- Now click “Deploy”.
- Copy the URL (web address) of your currently displayed Node-RED editor.
- Open a new Browser Tab (or new Browser window) and paste the copied URL.
- From the copied URL remove all characters after the IP address of your Node-Red and add **/test1** to the end of the URL (without quotes). Go to the URL.
- Congratulations! You created a web page.

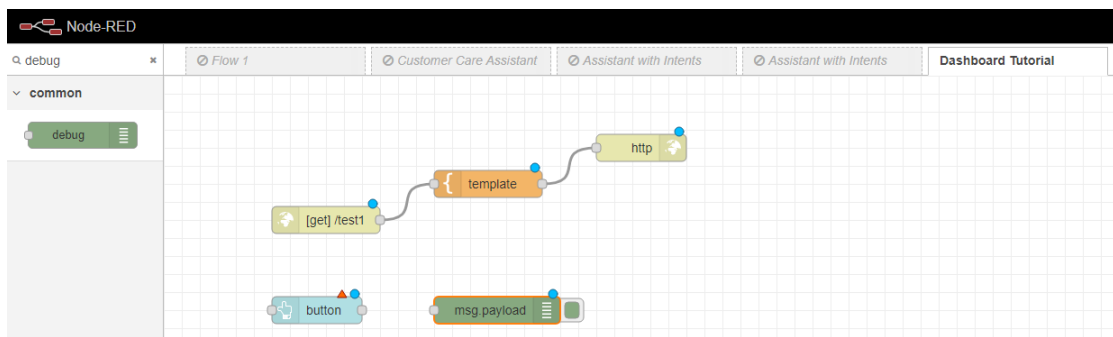
Hello World!

Your text here!

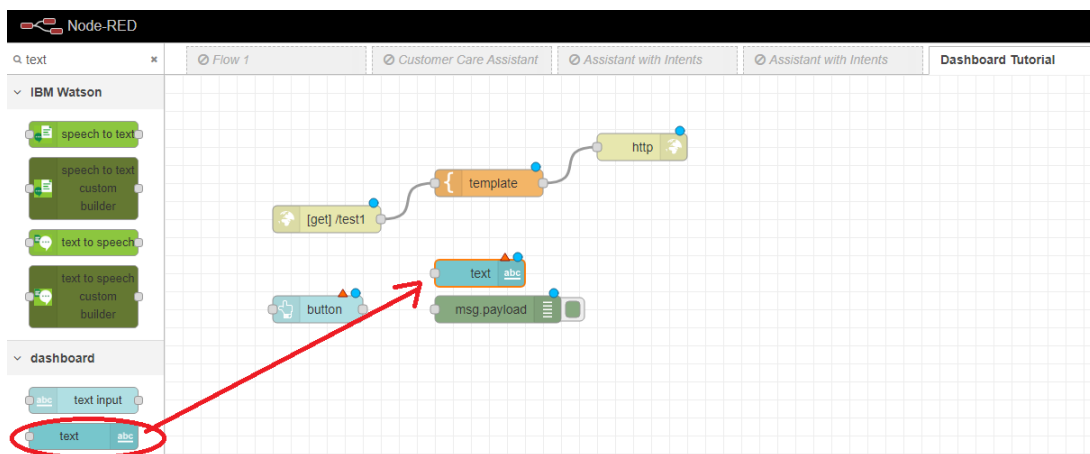
- Now drag and drop a “button” node.



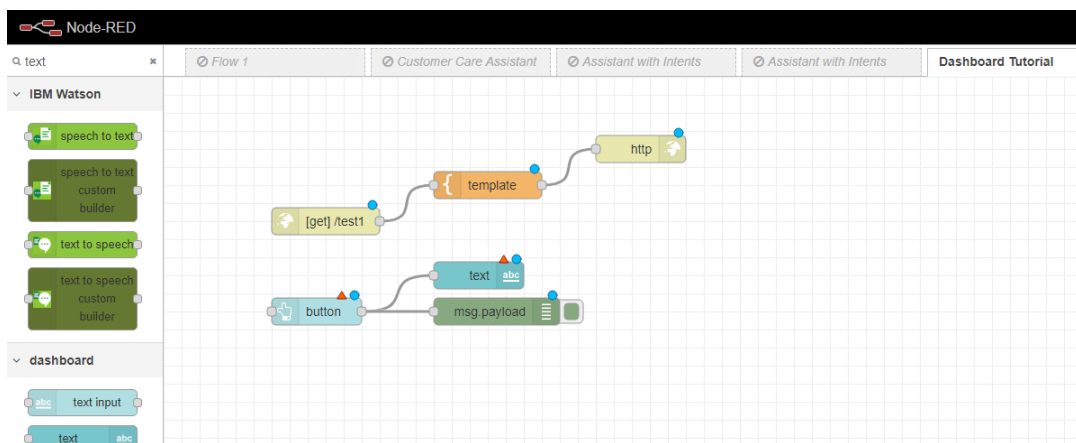
- Now drag and drop a “debug” node.



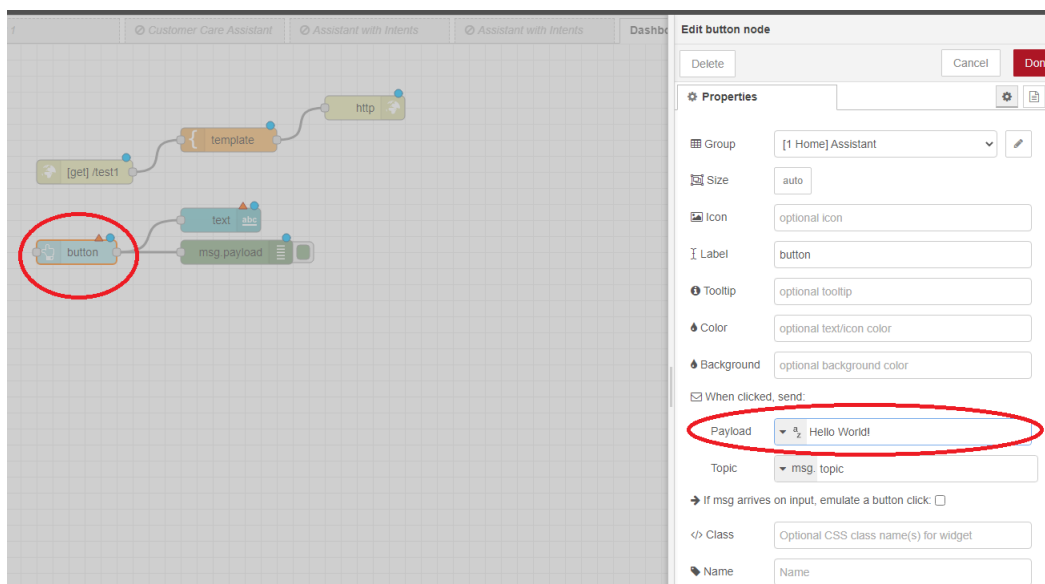
- Drag and drop a “text” node.



- Wire the nodes as follows:



- Double click on the “button” node and set the payload to **Hello World!**, as follows:



- Edit the groups.

Edit button node

Delete Cancel Done

Properties

Group [1 Home] Assistant

Size auto

Icon optional icon

Label button

Tooltip optional tooltip

Color optional text/icon color

Background optional background color

When clicked, send:

Payload α_2 Hello World

Topic msg. topic

→ If msg arrives on input, emulate a button click: ☐

Class Optional CSS class name(s) for widget

Name Name

- Create a group named “button”. Edit “Tab”.

Edit button node > Edit dashboard group node

Delete Cancel Update

Properties

Name Button

Tab Add new dashboard tab...

Class Optional CSS class name(s) for widget

Width 6

☒ Display group name

☒ Allow group to be collapsed

- Name of the “Tab” property as **Demo**. Click “update” and then “done”.

Edit button node > Edit dashboard group node > Add new dashboard tab config node

Cancel Add

Properties

Name Demo

Icon dashboard

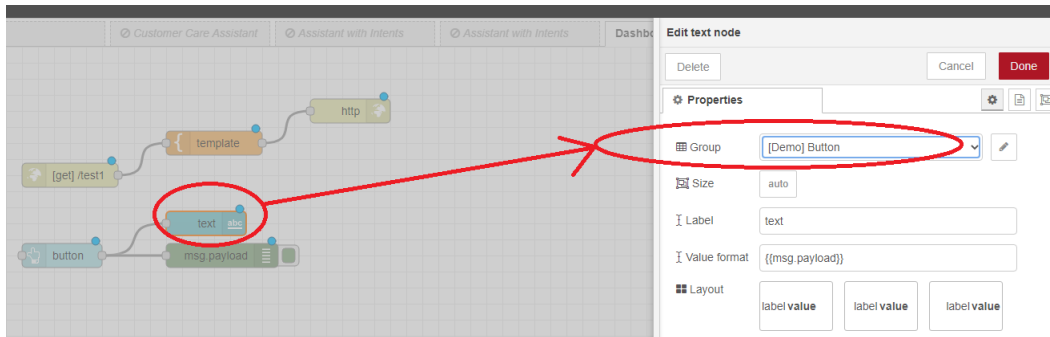
State ☒ Enabled

Nav. Menu ☒ Visible

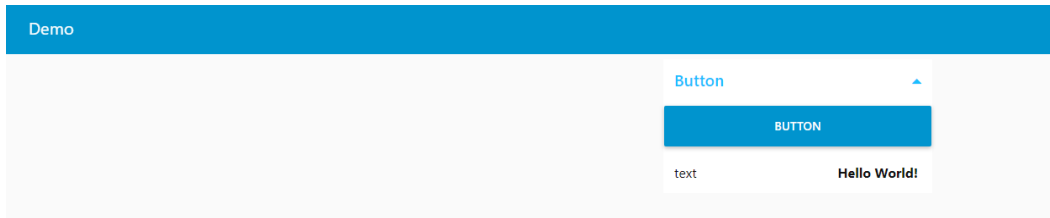
The **Icon** field can be either a **Material Design icon** (e.g. 'check', 'close') or a **Font Awesome icon** (e.g. 'fa-fire'), or a **Weather icon** (e.g. 'wi-wu-sunny').

You can use the full set of google material icons if you add 'mi-' to the icon name. e.g. 'mi-videogame_asset'.

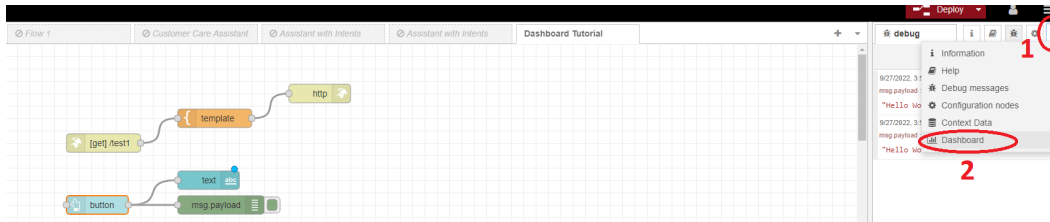
- Double click on the “text” node and make sure the correct group “[Demo] button” is selected.



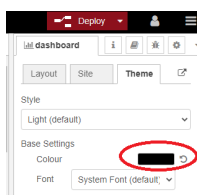
- Now go ahead and click on “Deploy”.
- Copy the URL (web address) of your currently displayed Node-RED editor.
- Open a new Browser Tab (or new Browser window) and paste the copied URL.
- From the copied URL remove all characters after your Node-Red IP address and add `/ui` to the end of the URL (without quotes). Go to the URL. There you can see a button and once you click on it you can see the text message “Hello World!”.



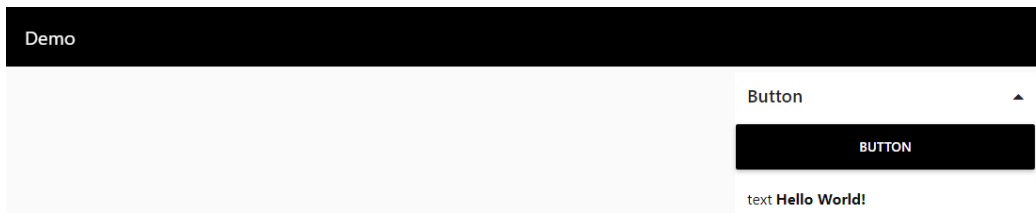
- Now go to the “Dashboard” tab.



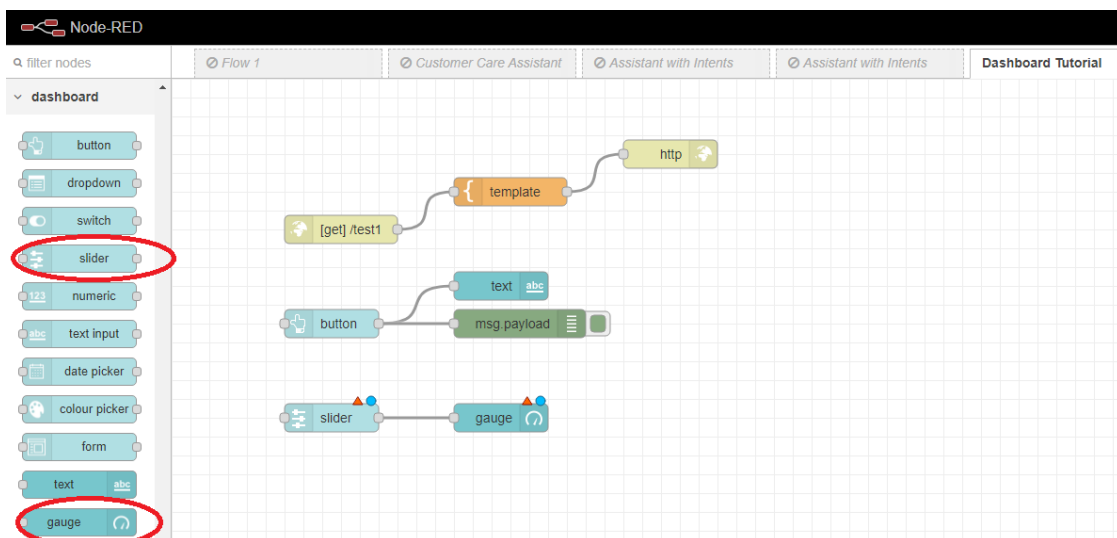
- Change the theme colour to “black”.



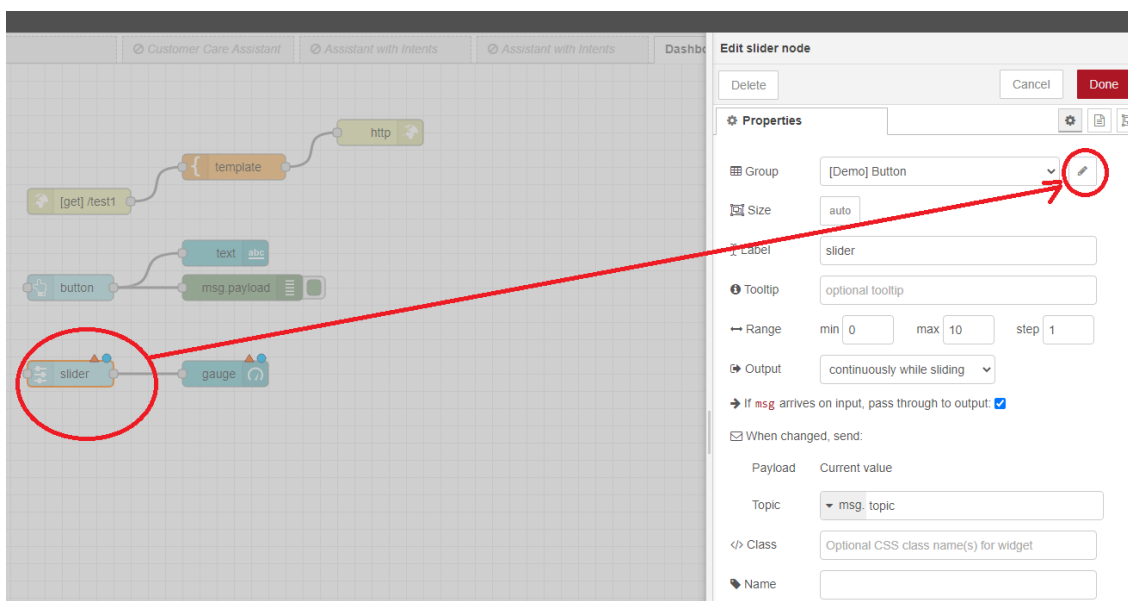
- Deploy the app again. You can now see a black theme on the `/ui` page.



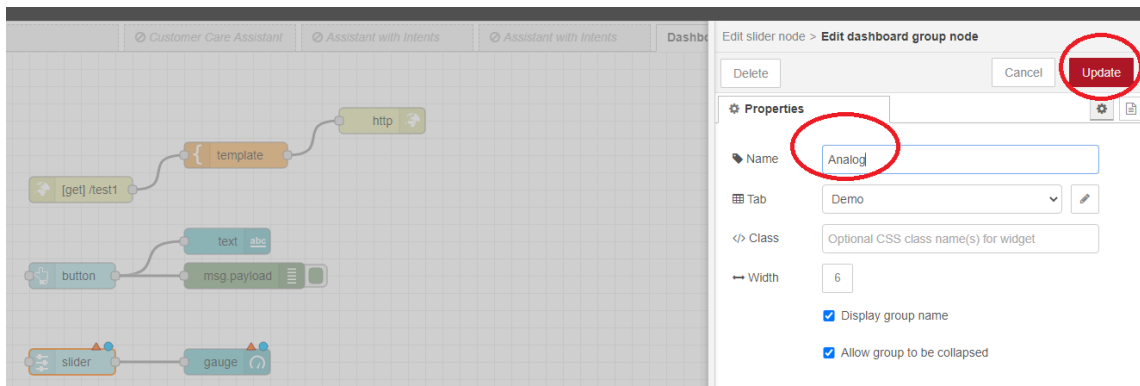
- Let us do a few more examples. Drag and drop a “slider” and a “gauge” node from the dashboard tab, and wire the nodes as follows.



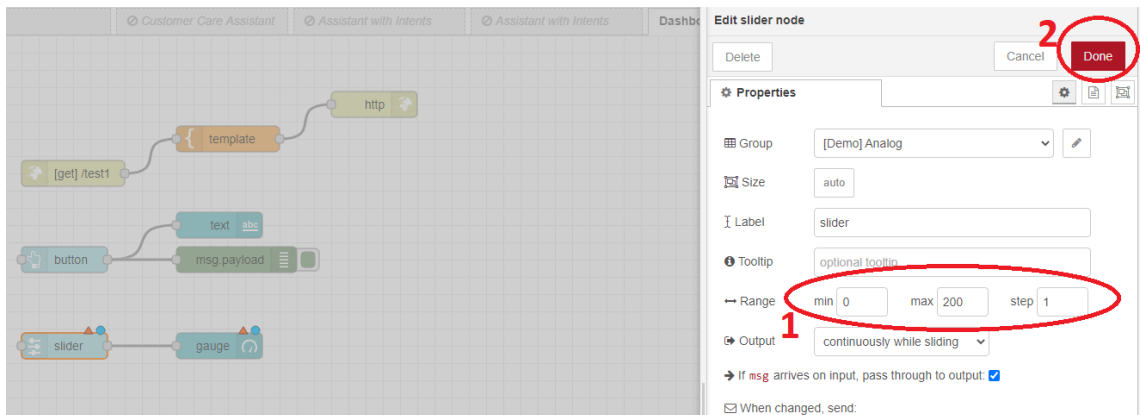
- Double-click on the “slider” node and define an “analog” group by clicking the edit button.



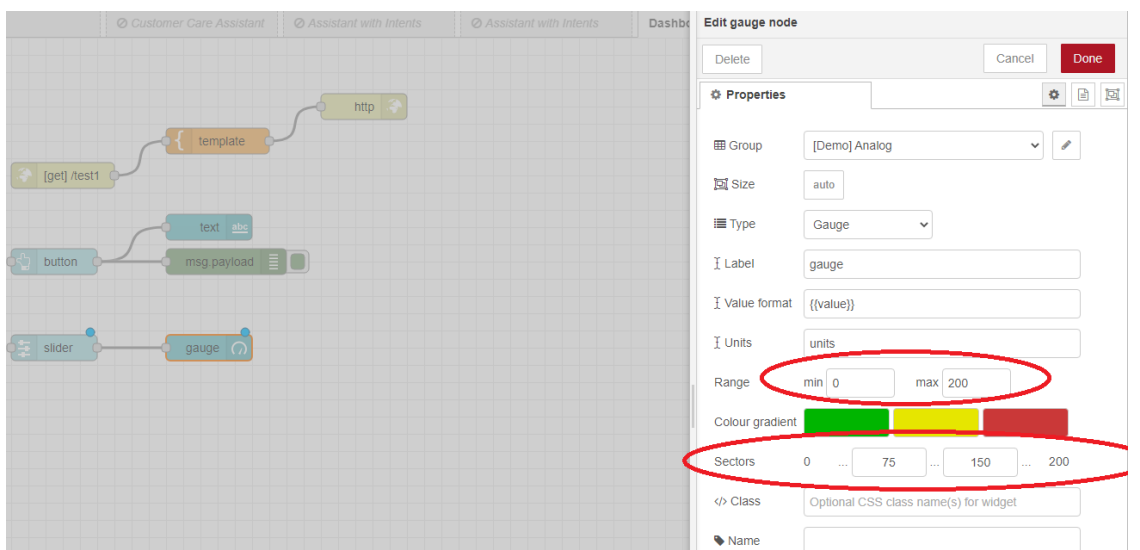
- Leave the tab as “Demo” and press update.



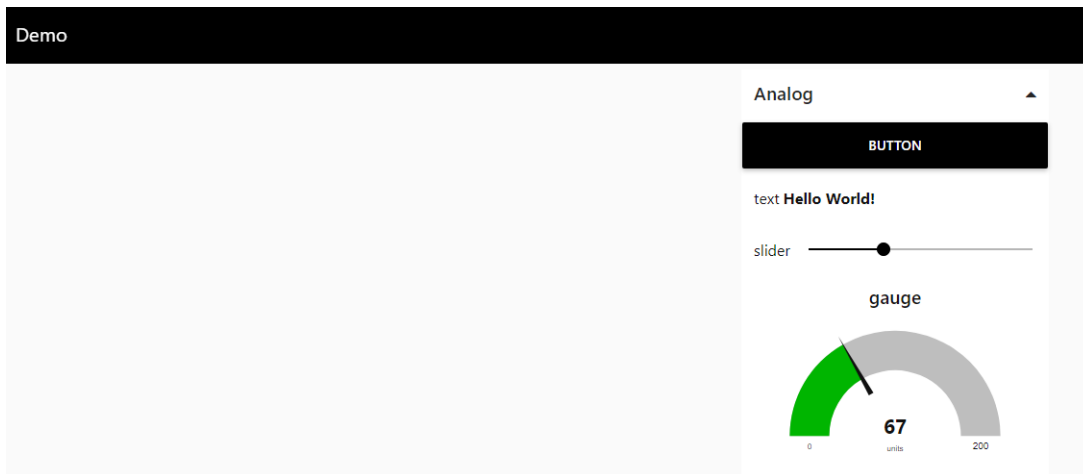
- Now choose the “Range” a value from 0 to 200, and then click “done”.



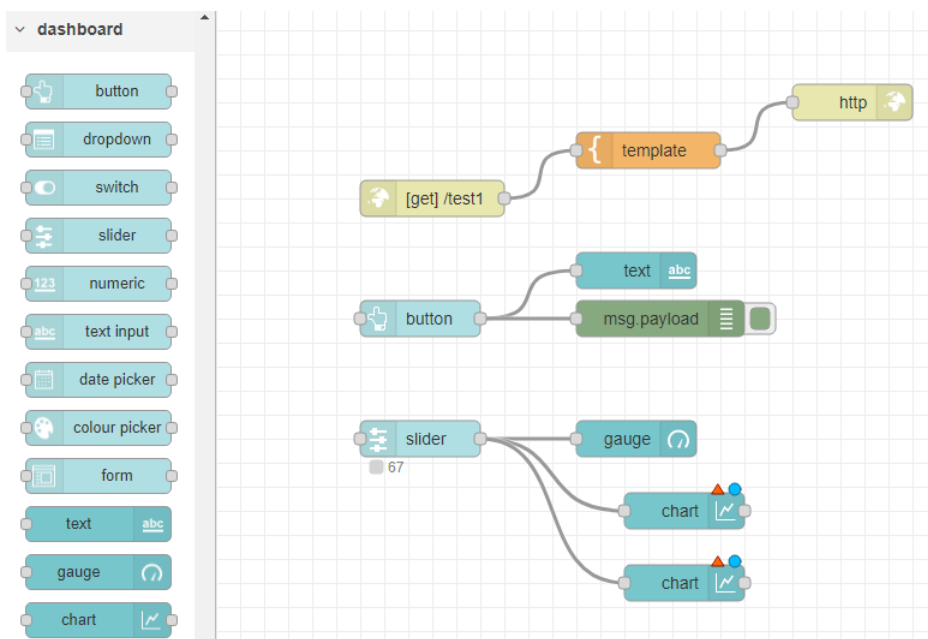
- Double click on the “gauge” node. Set up the gauge to be from zero to 200. We can also add colour sectors. For example, we can set between 0 and 75 to be green, between 75 and 150 to be yellow and then 150 to 200 to be red.



- Now deploy your flow and go to the `/ui` page. As you can see, when you move the slider, the gauge colour changes.



- Go back to your Node-RED flow page and add two “chart” nodes and wire them to your “slider” node.



- Let’s make one of the new nodes a “line chart” and make the other one a “bar chart”. Make sure the charts are on the “analog” tab.

DeleteCancelDone

Properties

Group

[Demo] Analog

Size

auto

Label

chart

Type

Line chart

☐ enlarge points

X-axis

last 1 hours OR 1000 points

X-axis Label

HH:mm:ss

☐ as UTC

Y-axis

min max

Legend

None

Interpolate linear

Series Colours

Blank label

display this text before valid data arrives

</> Class

Optional CSS class name(s) for widget

Name

Name

DeleteCancelDone

Properties

Group

[Demo] Analog

Size

auto

Label

chart

Type

Bar chart

☐ Use first colour for all bars

Y-axis

min max

Legend

None

Series Colours

Blank label

display this text before valid data arrives

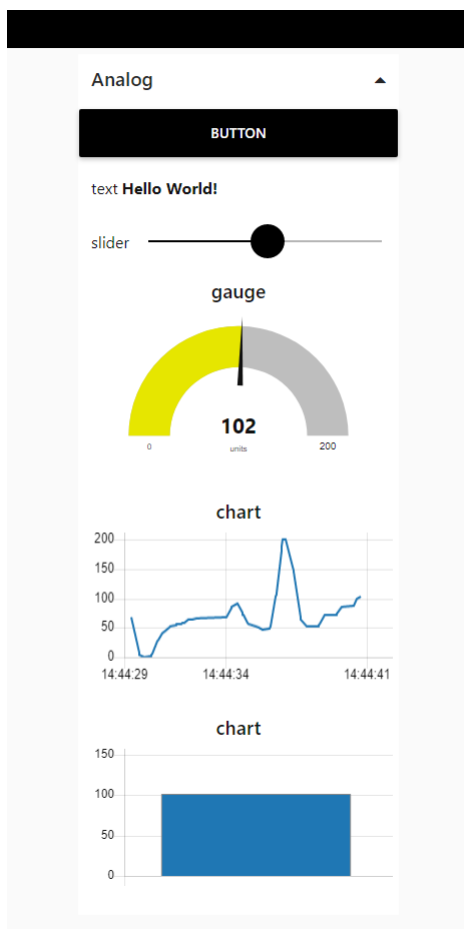
</> Class

Optional CSS class name(s) for widget

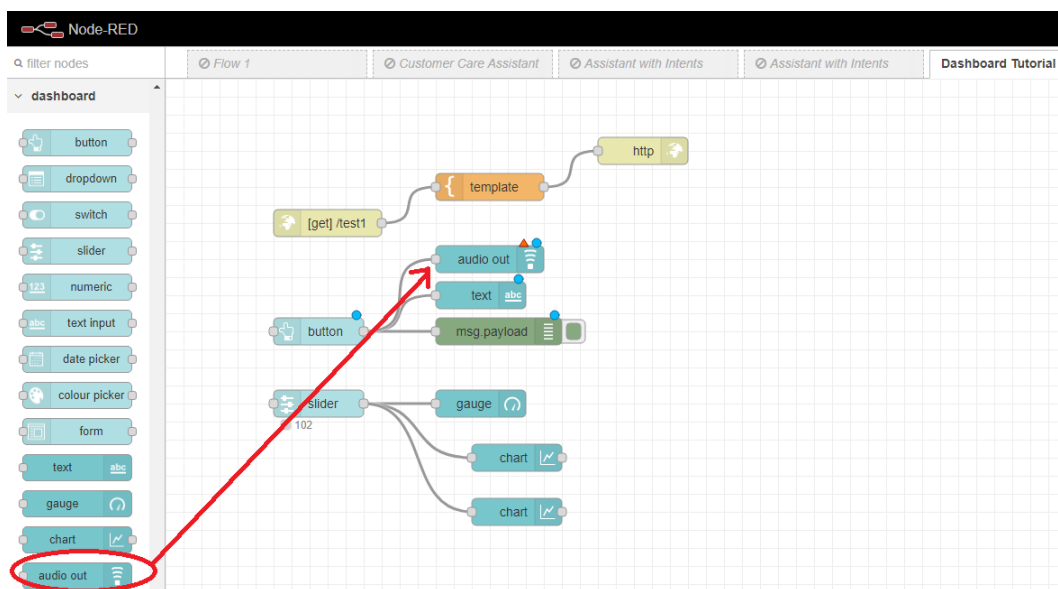
Name

Name

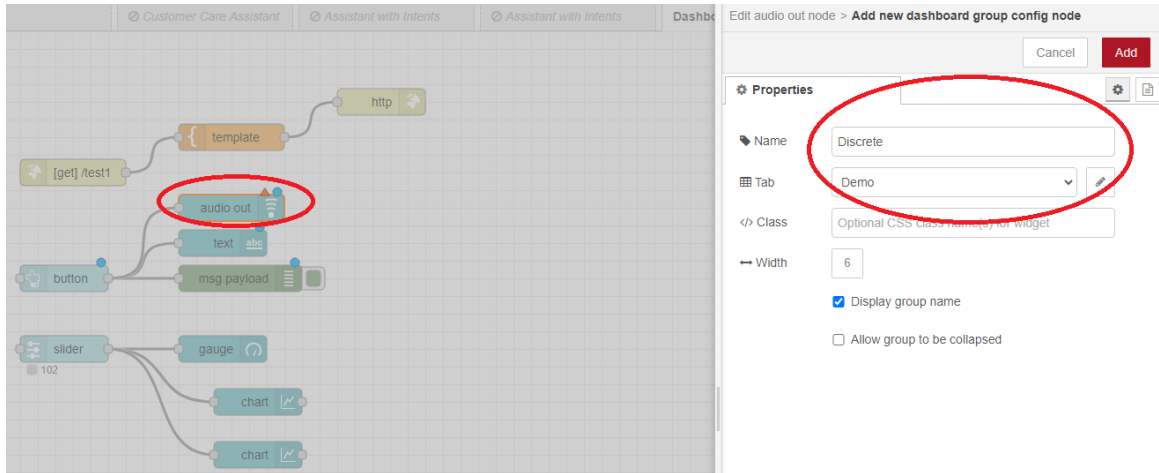
- Now deploy your flow and go to the `/ui` page. As you can see, when you move the slider, the graph in the charts would appear and it would change as you move the slider.



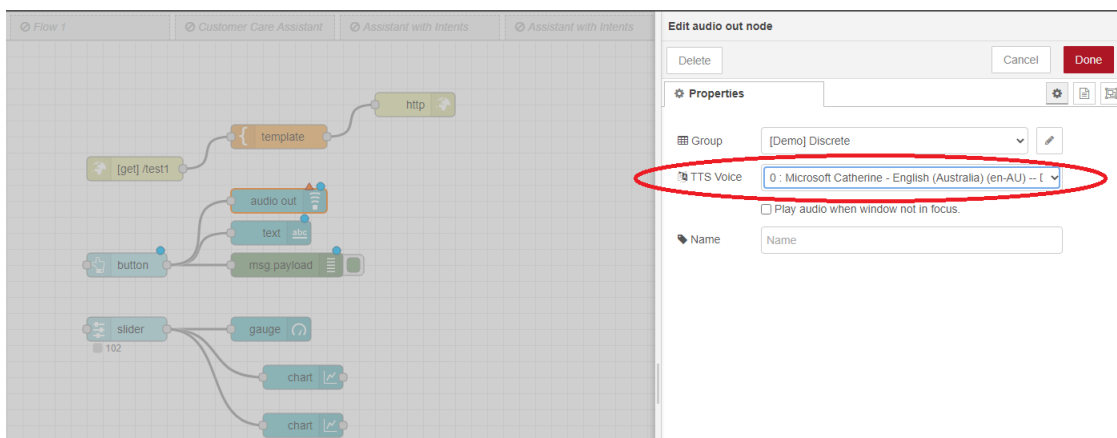
- Go back to your Node-RED flow page and add an “Audio out” node and wire it to your “button” node. So, when you input a text like our “Hello World!”, it should be able to speak out.



- Double click on the “Audio out” node and define a new “Discrete” group under the “Demo” tab.



- Select the default option for the “TTS Voice”.



- Deploy your flow again and go to the `/ui` page. Test your button and listen to the sweet “Hello World!” 😊.
- Congratulations and well-done for creating your own customised dashboard.

———— End of Laboratory Week 9 ————