

Fundamentals of IoT Software © 2022 by Luca Mottola
is licensed under CC BY-NC 4.0



To view a copy of this license, visit
creativecommons.org/licenses/by-nc/4.0/

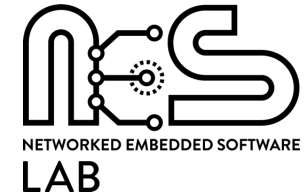




POLITECNICO
MILANO 1863



POLITECNICO
MILANO 1863



Node-RED Lab

Luca Mottola

`luca.mottola@polimi.it`

(version 0.1)

SMTP Server

- In some of the exercises, you are required to use the **node-red-node-email** node
- Unless you prefer to use your own SMTP server, you can use
 - Server: **smtps.aruba.it**
 - Userid: **nodered@neslab.it**
 - Password: **Node22\$\$**
 - Port: 465
 - Authentication: password with secure connection



Exercise 1

- Find here an example code to start from:
`bit.ly/3AS5mFG`
- Change the example flow we just created to use a custom subject and a CC address
 - To do that, check the documentation of the email node
`flows.nodered.org/node/node-red-node-email`



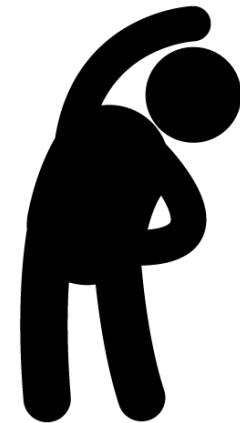
Exercise 2

- Starts from the solution of Exercise 1
- Replace the delay node with a function node that swaps “to” and “cc” fields
- Use the function node in the previous point to insert the content of the “to” field in the email content
 - Example email content:
Hello World luca.mottola@polimi.it!
- Further modify the function node to send email when the timestamp is even, or dump the message on the debug window otherwise



Exercise 3

- Modify Exercise 2 seen before so that the last three timestamps appear in the email content
 - Bear in mind: using any form of context requires proper initialization



Exercise 4



- Now install the `node-red-node-openweathermap` extension
- Configure the node with the following data:
 - API Key: `2caa90098525566a5c251ebb92abd882`
 - City: `Milan`
 - Country: `IT`
- First, inspect the output of the node when triggered
- Next, develop a flow that creates a file log of the Celsius temperature every minute



Exercise 5

- Extend the solution to Exercise 4 to read the entire log from the file every minute
- Note: this may be implemented as a separate flow, or as part of the flow of Exercise 4



Exercise 6

- A UDP **Echo server** is a UDP application that simply bounces back whatever data it gets to the original sender
- Find here a simple Node-RED implementation of an Echo server: bit.ly/3GRQ1ZK
- Create a flow that sends to the Echo server an object with two properties:
 - A string “The temperature in Milan is”
 - A number with the current temperature as reported by OpenWeatherMap
- Wait for the reply on port 5555
 - ...and verify the data is the same sent earlier!



MQTT Server

- The exercises coming next use an MQTT server bridging from sensor.community
 - Server name: `mqtt.neslab.it`
 - Port: 3200
 - No client ID
 - No authentication



Exercise 7

- Using MQTT, subscribe to `/smartcity/milan` to receive data from `sensor.community`
- Use a debug node to show the highest value received so far for
 - Temperature
 - Humidity
 - P2.5 (indicated as `P2`)
 - PM10 (indicated as `P1`)
- Note: assume these measures cannot be lower than 0



Building Bots (1/4)

- The exercises coming next use the `node-red-contrib-chatbot` extension, which provides a rich set of nodes to build **bots** for Telegram, Facebook Messenger, ...
- We use Telegram
 - For configuration, you need to ask `t.me/BotFather` to create an API key for your new bot
 - You use the username and API key to configure the nodes



Building Bots (2/4)



Luca Mottola

/start



BotFather

I can help you create and manage Telegram bots. If you're new to the Bot API, please [see the manual](#).

You can control me by sending these commands:

[/newbot](#) - create a new bot

[/mybots](#) - edit your bots **[beta]**

Edit Bots

[/setname](#) - change a bot's name

[/setdescription](#) - change bot description

[/setabouttext](#) - change bot about info

[/setuserpic](#) - change bot profile photo

[/setcommands](#) - change the list of commands

[/deletebot](#) - delete a bot

Bot Settings

[/token](#) - generate authorization token

[/revoke](#) - revoke bot access token

[/setinline](#) - toggle [inline mode](#)

[/setinlinegeo](#) - toggle inline [location requests](#)

[/setinlinefeedback](#) - change [inline feedback](#) settings

[/setjoiningroups](#) - can your bot be added to groups?

[/setprivacy](#) - toggle [privacy mode](#) in groups

Games

[/mygames](#) - edit your [games](#) **[beta]**

[/newgame](#) - create a new [game](#)

[/listgames](#) - get a list of your games

[/editgame](#) - edit a game

[/deletgame](#) - delete an existing game



Building Bots (3/4)



Luca Mottola
/newbot



BotFather

Alright, a new bot. How are we going to call it? Please choose a name for your bot.



Luca Mottola
Node-RED Lab



BotFather

Good. Now let's choose a username for your bot. It must end in `bot`. Like this, for example: TetrisBot or tetris_bot.



Luca Mottola
nodered_lab_bot



BotFather

Done! Congratulations on your new bot. You will find it at t.me/nodered_lab_bot. You can now add a description, about section and profile picture for your bot, see [/help](#) for a list of commands. By the way, when you've finished creating your cool bot, ping our Bot Support if you want a better username for it. Just make sure the bot is fully operational before you do this.

Use this token to access the HTTP API:

5686644067:AAHgqS-PiJY-0t0bhAaNwuqPv-8c-j0piiQ

Keep your token **secure** and **store it safely**, it can be used by anyone to control your bot.

For a description of the Bot API, see this page: <https://core.telegram.org/bots/api>



Building Bots (4/4)

- Download the example at bit.ly/3ViBg6p
- Check the configuration of the Telegram Receiver node

Edit Telegram In node

Delete Cancel Done

Properties [Settings] [File] [View]

Bot configuration (development)

Node-RED Lab [Edit]

Bot configuration (production)

Add new chatbot-telegram-node... [Edit]

Bot for **production** will be launched only if the global variable "environment" in settings.js is set to "production", otherwise will be used the configuration for **development**.

Enter here the name and API key of your bot, leave the rest unchanged



Exercise 8-1

- Modify the example bot with two more commands that people can use to ask what is the current temperature in Milan or Rome
- Example:
 - User: “What is the temperature in Milan?”
 - Bot: “The current temperature in Milan is 7C”
- Hint: check the documentation of the **Text** node in the extension



Exercise 8-2

- Now extend the previous solution to
 - Add the ability for the bot to greet back the user
 - Keep a persistent log of all the usernames seen so far
- Example:
 - User: “My name is Luca!”
 - Bot: “Hello Luca!”
 - (while “Luca” is made persistent)

