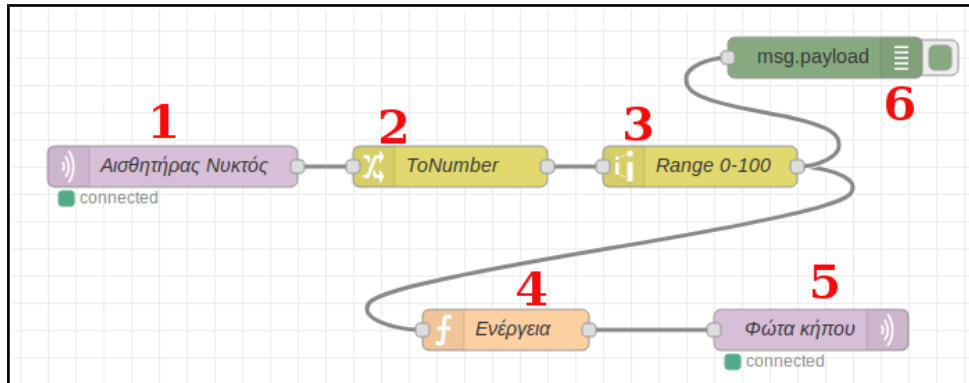


Φύλλο εργασίας 2

Να αυτοματοποιήσετε την διαδικασία ενεργοποίησης του φωτισμού κήπου όταν η στάθμη του ηλιακού φωτός πέσει κάτω από ένα συγκεκριμένο επίπεδο π.χ. 50%.

Αρχικό διάγραμμα ροής

Σχεδιάζουμε το παρακάτω διάγραμμα και εξηγούμε ένα-ένα τα blocks.



1. Είναι ένα **mqtt in** το οποίο το βρίσκουμε στην ενότητα **input**. Εδώ πρέπει υποχρεωτικά να δηλώσουμε τον Server που είναι το mosquitto με όνομα Smart_Local (localhost), το topic που είναι 'garden/light_sens' και προαιρετικά δίνουμε το όνομα 'Αισθητήρας Νυκτός'.
2. Επειδή ο controller δημοσιεύει τις τιμές έντασης φωτός σε μορφή συμβολοσειράς (string), πρέπει να μετατραπεί σε αριθμό. Επιλέγουμε το block **Change** από την ενότητα **function** και το διορθώνουμε όπως το παρακάτω στιγμιότυπο.

Dialog box: Edit change node

Buttons: Delete, Cancel, Done

node properties

Name: ToNumber

Rules

Set msg.payload to \$number(payload)

3. Επειδή τώρα έχουμε τιμές ακέραιες από 0-1023 (0:σκοτάδι – 1023:μέγιστο φως) και θα θέλαμε να δουλέψουμε με ποσοστό, πρέπει να γίνει μετατροπή στη κλίμακα 0-100. Αυτό το κάνει το block **range** από την ενότητα **function**. Δηλαδή μετατρέπει τις τιμές με εύρος 0-1023 στο εύρος 0-100.

Dialog box: Edit range node

Buttons: Delete, Cancel, Done

node properties

Property: msg.payload

Action: Scale the message property

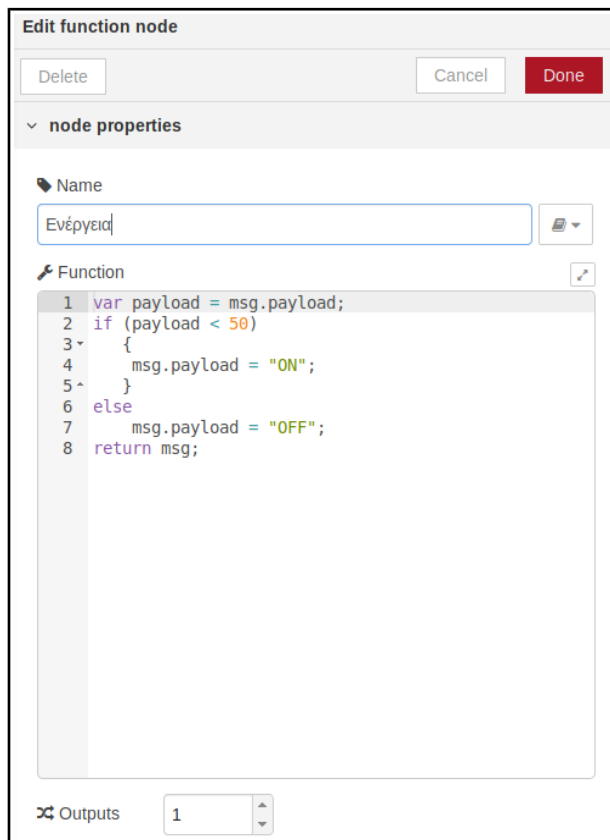
Map the input range: from: 0 to: 1023

to the result range: from: 0 to: 100

☒ Round result to the nearest integer?

Name: Range 0-100

4. Εδώ θα χρειαστεί και λίγο κώδικας, έτσι βάζουμε το block **function** από την ενότητα **function** και την συμπληρώνουμε σύμφωνα με το στιγμιότυπο. Η γλώσσα είναι Java Script.

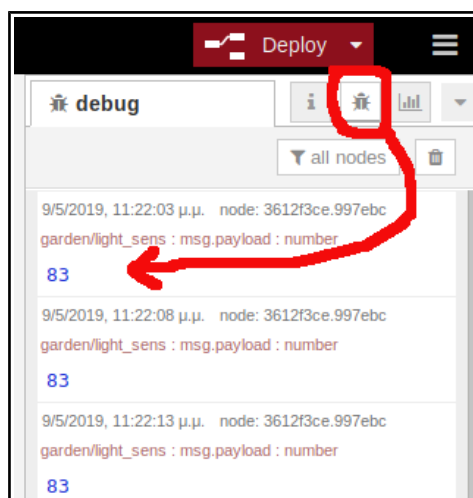


Κώδικας

```
var payload = msg.payload;
if (payload < 50)
{
    msg.payload = "ON";
}
else
    msg.payload = "OFF";
return msg;
```

Σύμφωνα με το κώδικα ελέγχουμε την τιμή του payload που λαμβάνουμε από την range και αν είναι μικρότερο από 50 μετατρέπουμε το payload σε string "ON", αλλιώς σε "OFF".

5. Από την ενότητα **output** βάζουμε ένα block **mqtt out**. Εδώ απαιτείται το πεδίο Server που είναι Smart_Local και το Topic που είναι 'garden/light_switch'. Δηλαδή λαμβάνει το payload τύπου string από το block function και το δημοσιεύει στον broker.
6. Πολλές φορές είναι απαραίτητη η εκσφαλμάτωση, γι' αυτό από την ενότητα **output** εισάγουμε το block **debug**. Για να δούμε τα μηνύματα debugging πατάμε το κουμπί με το σκαθάρι.



Πατάμε το Deploy και δοκιμάζουμε το διάγραμμα. Με το χέρι καλύπτουμε το LDR του κήπου μέχρι η ένταση του φωτός να πέσει κάτω από 50% και τότε τα φώτα πρέπει να ανάβουν.

Μια μικρή διόρθωση

Επειδή ο controller μεταδίδει κάθε 4 δευτερόλεπτα ένα μήνυμα με την ένταση του ηλιακού φωτός, το διάγραμμα που μόλις φτιάξαμε, δημοσιεύει ένα μήνυμα 'ON' ή 'OFF' στο topic 'garden/light_switch' με την ίδια συχνότητα, φορτώνοντας το σύστημα με περιττά publish χωρίς αυτά να είναι απαραίτητα. Αν κάνουμε την παρακάτω τροποποίηση στην συνάρτηση Ενέργεια που εμείς φτιάξαμε, τότε θα κάνει publish μόνο αν υπάρχει αλλαγή της κατάστασης από 'OFF' σε 'ON' και αντίστροφα.

```
1  var payload = msg.payload;
2  var state = context.get('state') || false;
3  if (payload < 50)
4      if (!state)
5          {
6              msg.payload = "ON";
7              state = true;
8          }
9      else
10         msg.payload = "";
11  else
12      if (state)
13          {
14              msg.payload = "OFF";
15              state = false;
16          }
17      else
18         msg.payload = "";
19  context.set('state', state);
20  if (msg.payload !== "")
21      return msg;
```

Στην γραμμή 1 αποθηκεύουμε την παράμετρο msg.payload που δεχόμαστε από το αντικείμενο msg από το προηγούμενο block, στην τοπική μεταβλητή payload, η οποία στην συγκεκριμένη περίπτωση είναι τύπου String. Στην 2η γραμμή διαβάζουμε την τιμή μιας μεταβλητής με όνομα 'state' και την αποθηκεύουμε στην τοπική state. Η μεταβλητή αυτή είναι όπως οι στατικές μεταβλητές της C, δηλαδή την επόμενη φορά που θα κληθεί η συνάρτηση, πρέπει να έχει κρατήσει την προηγούμενη τιμή. Την πρώτη φορά αν δεν έχει οριστεί την αρχικοποιούμε με την τιμή false. Πριν βγούμε από την συνάρτηση πρέπει να την αποθηκεύσουμε για να την έχουμε την επόμενη φορά. Αυτό γίνεται στην γραμμή 19. Η μεταβλητή αυτή έχει εμβέλεια μόνο στο συγκεκριμένο context δηλαδή ισχύει μόνο για αυτό το Node. Στην 3η γραμμή γίνεται η σύγκριση της έντασης και αν είναι κάτω από 50 τότε πρέπει να ανάψει το φως. Στην 4η γραμμή ελέγχει αν είναι ήδη αναμμένο και αν είναι (γραμμή 9) τότε το payload είναι κενό string. Στις 20 και 21 επιστρέφουμε το msg μόνο αν δεν είναι κενό, δηλαδή μόνο αν υπάρχει αλλαγή της κατάστασης.

Ένα βήμα πιο πέρα

Τώρα θα τοποθετήσουμε ένα control στο ui ώστε ο χρήστης να μπορεί να ορίσει το επίπεδο του φωτός που θα ανάβουν τα φώτα κήπου. Αρχικά από την ομάδα **dashboard** εισάγουμε ένα control τύπου **slider** και διορθώνουμε τις ιδιότητες όπως στην εικόνα που ακολουθεί.

Μετά από την ομάδα **function** εισάγουμε ένα block **function** και γράφουμε τον εξής κώδικα:

```
1 var gard_intens = flow.get('gard_intens') || 0;
2 gard_intens = msg.payload;
3 flow.set('gard_intens', gard_intens);
```

Δηλαδή μόλις σύρουμε το slider θα δημιουργηθεί ένα συμβάν το οποίο θα καλέσει την παραπάνω συνάρτηση. Στην γραμμή 1 ορίζουμε μια μεταβλητή η οποία έχει εμβέλεια σε όλο το Flow (Καθολική) και αν είναι η πρώτη φορά θα γίνει 0. Στην 2 αποθηκεύουμε την επιλεγμένη τιμή του slider στην τοπική μεταβλητή `gard_intens`. Τέλος στη 3 αποθηκεύουμε την τιμή στην καθολική μεταβλητή ώστε να την διαβάσει η συνάρτηση 'Ενέργεια'.

Κάνουμε μια μικρή τροποποίηση στη συνάρτηση 'Ενέργεια' και πατάμε Deploy.

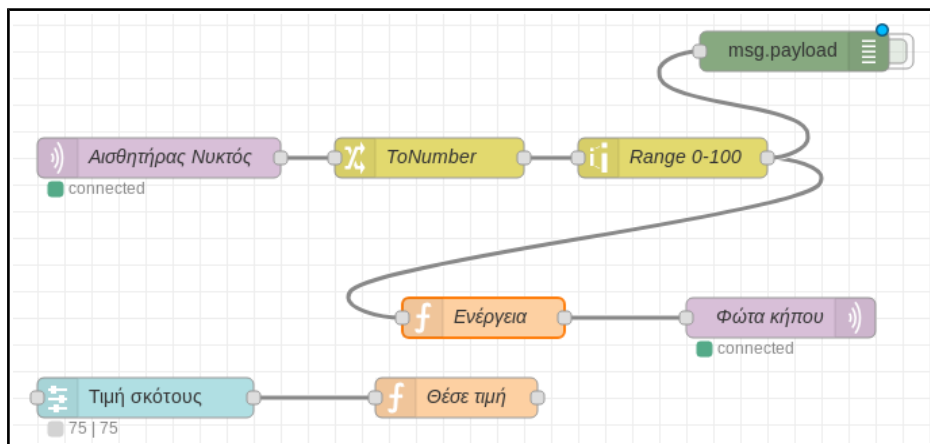
```
1 var payload = msg.payload;
2 var state = context.get('state') || false;
3 var gard_intens = flow.get('gard_intens') || 50; //<---
4 if (payload < gard_intens)                       //<---
5     if (!state)
6     {
7         msg.payload = "ON";
```

```

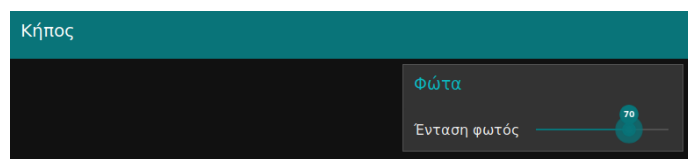
8      state = true;
9    }
10   else
11     msg.payload = "";
12 else
13   if (state)
14   {
15     msg.payload = "OFF";
16     state = false;
17   }
18   else
19     msg.payload = "";
20 context.set('state', state);
21 if (msg.payload !== "")
22   return msg;

```

Βασικά εδώ έγιναν δύο αλλαγές στο προηγούμενο. **α)** Στη γραμμή 3 διαβάζουμε τη τιμή της καθολικής μεταβλητής για όλο το flow και αν δεν έχει οριστεί βάζουμε 50. **β)** Στη γραμμή 4 συγκρίνουμε αυτή την τιμή και όχι την σταθερά 50.



Και το περιβάλλον UI.



Το Flow για εισαγωγή στο Node Red.

```

[
  {
    "id": "12590750.604991",
    "type": "tab",
    "label": "Flow 2",
    "disabled": false,
    "info": ""
  },
  {
    "id": "c65c8f3d.534c48",
    "type": "mqtt in",
    "z": "12590750.604991",
    "name": "Αισθητήρας Νυκτός",
    "topic": "garden/light_sens",
    "qos": "2",
    "broker": "4da92890.dfec98",
    "x": 150,
    "y": 140,
    "wires": [
      [
        "c69180b0.7c7118"
      ]
    ]
  },
  {
    "id": "c69180b0.7c7118",
    "type": "function",
    "z": "12590750.604991",
    "name": "Ενέργεια",
    "func": "msg.payload = 50; if (msg.payload !== \"\") return msg;",
    "x": 250,
    "y": 140,
    "wires": [
      [
        "c65c8f3d.534c48"
      ]
    ]
  },
  {
    "id": "c65c8f3d.534c48",
    "type": "mqtt out",
    "z": "12590750.604991",
    "name": "Φώτα κήπου",
    "topic": "garden/light_act",
    "qos": "2",
    "broker": "4da92890.dfec98",
    "x": 350,
    "y": 140,
    "wires": [
      [
        "c65c8f3d.534c48"
      ]
    ]
  },
  {
    "id": "c65c8f3d.534c48",
    "type": "function",
    "z": "12590750.604991",
    "name": "Τιμή σκότους",
    "func": "msg.payload = 75; if (msg.payload !== \"\") return msg;",
    "x": 350,
    "y": 180,
    "wires": [
      [
        "c65c8f3d.534c48"
      ]
    ]
  },
  {
    "id": "c65c8f3d.534c48",
    "type": "function",
    "z": "12590750.604991",
    "name": "Θέσε τιμή",
    "func": "msg.payload = 70; if (msg.payload !== \"\") return msg;",
    "x": 350,
    "y": 220,
    "wires": [
      [
        "c65c8f3d.534c48"
      ]
    ]
  }
]

```

```

    "id": "c69180b0.7c7118",
    "type": "change",
    "z": "12590750.604991",
    "name": "ToNumber",
    "rules": [
      {
        "t": "set",
        "p": "payload",
        "pt": "msg",
        "to": "$number(payload)\t",
        "tot": "jsonata"
      }
    ],
    "action": "",
    "property": "",
    "from": "",
    "to": "",
    "reg": false,
    "x": 350,
    "y": 140,
    "wires": [
      [
        "a97504d3.52fe"
      ]
    ]
  },
  {
    "id": "a97504d3.52fe",
    "type": "range",
    "z": "12590750.604991",
    "minin": "0",
    "maxin": "1023",
    "minout": "0",
    "maxout": "100",
    "action": "scale",
    "round": true,
    "property": "payload",
    "name": "Range 0-100",
    "x": 530,
    "y": 140,
    "wires": [
      [
        "9be7e85d.a3f6d8",
        "3612f3ce.997ebc"
      ]
    ]
  },
  {
    "id": "9be7e85d.a3f6d8",
    "type": "function",
    "z": "12590750.604991",
    "name": "Ενέπνευστα",
    "func": "var payload = msg.payload;\nvar state = context.get('state') || false;\nvar gard_intens = flow.get('gard_intens') ||\n50;\nif (payload < gard_intens)\n  if (!state)\n    {\n      msg.payload = \"ON\";\n      state = true;\n    }\nelse\n  msg.payload = \"\";\nelse\n  if (state)\n    {\n      msg.payload = \"OFF\";\n      state = false;\n    }\nelse\n  msg.payload = \"\";\ncontext.set('state', state);\nif (msg.payload !== \"\")\n  return msg;",
    "outputs": 1,
    "noerr": 0,
    "x": 390,
    "y": 260,
    "wires": [
      [
        "3055154.f53d8ea"
      ]
    ]
  },
  {
    "id": "3055154.f53d8ea",
    "type": "mqtt out",
    "z": "12590750.604991",
    "name": "Φώτα κήπου",
    "topic": "garden/light_switch",
    "qos": "",
    "retain": "",
    "broker": "4da92890.dfec98",
    "x": 610,
    "y": 260,
    "wires": []
  },
  {
    "id": "3612f3ce.997ebc",
    "type": "debug",
    "z": "12590750.604991",
    "name": "",
    "active": false,
    "tosidebar": true,
    "console": false,
    "tostatus": false,
    "complete": "payload",
    "x": 620,
    "y": 60,
    "wires": []
  },
  {
    "id": "1b0774b7.7ed803",
    "type": "ui_slider",
    "z": "12590750.604991",
    "name": "Τιμή σκότους",
    "label": "Ένταση φωτός",

```

```

"tooltip": "Ρυθμίστε το επίπεδο στο οποίο θα ανάβουν τα φώτα",
"group": "3cbb92d3.ef1a4e",
"order": 0,
"width": 0,
"height": 0,
"passthru": false,
"outs": "end",
"topic": "",
"min": 0,
"max": "100",
"step": "5",
"x": 140,
"y": 320,
"wires": [
  [
    "62dc458e.12e5fc"
  ]
]
},
{
  "id": "62dc458e.12e5fc",
  "type": "function",
  "z": "12590750.604991",
  "name": "Θέσε τιμή",
  "func": "var gard_intens = flow.get('gard_intens') || 0;\ngard_intens = msg.payload;\nflow.set('gard_intens', gard_intens);",
  "outputs": 1,
  "noerr": 0,
  "x": 370,
  "y": 320,
  "wires": [
    []
  ]
},
{
  "id": "4da92890.dfec98",
  "type": "mqtt-broker",
  "z": "",
  "name": "Smart_Local",
  "broker": "localhost",
  "port": "1883",
  "clientId": "",
  "usetls": false,
  "compatmode": true,
  "keepalive": "60",
  "cleansession": true,
  "birthTopic": "",
  "birthQos": "0",
  "birthPayload": "",
  "closeTopic": "",
  "closeQos": "0",
  "closePayload": "",
  "willTopic": "",
  "willQos": "0",
  "willPayload": ""
},
{
  "id": "3cbb92d3.ef1a4e",
  "type": "ui_group",
  "z": "",
  "name": "Φώτα",
  "tab": "9e5733bc.6a4a9",
  "order": 1,
  "disp": true,
  "width": "6",
  "collapse": false
},
{
  "id": "9e5733bc.6a4a9",
  "type": "ui_tab",
  "z": "",
  "name": "Κήπος",
  "icon": "dashboard",
  "order": 1,
  "disabled": false,
  "hidden": false
}
]

```

Άσκηση

Βάζοντας ένα block τύπου Text από την ομάδα Dashboard, φτιάξτε τις κατάλληλες ρυθμίσεις ώστε ο απομακρυσμένος χρήστης να βλέπει το μήνυμα ‘Αναμμένα’ ή ‘Σβηστά’ από το περιβάλλον UI.