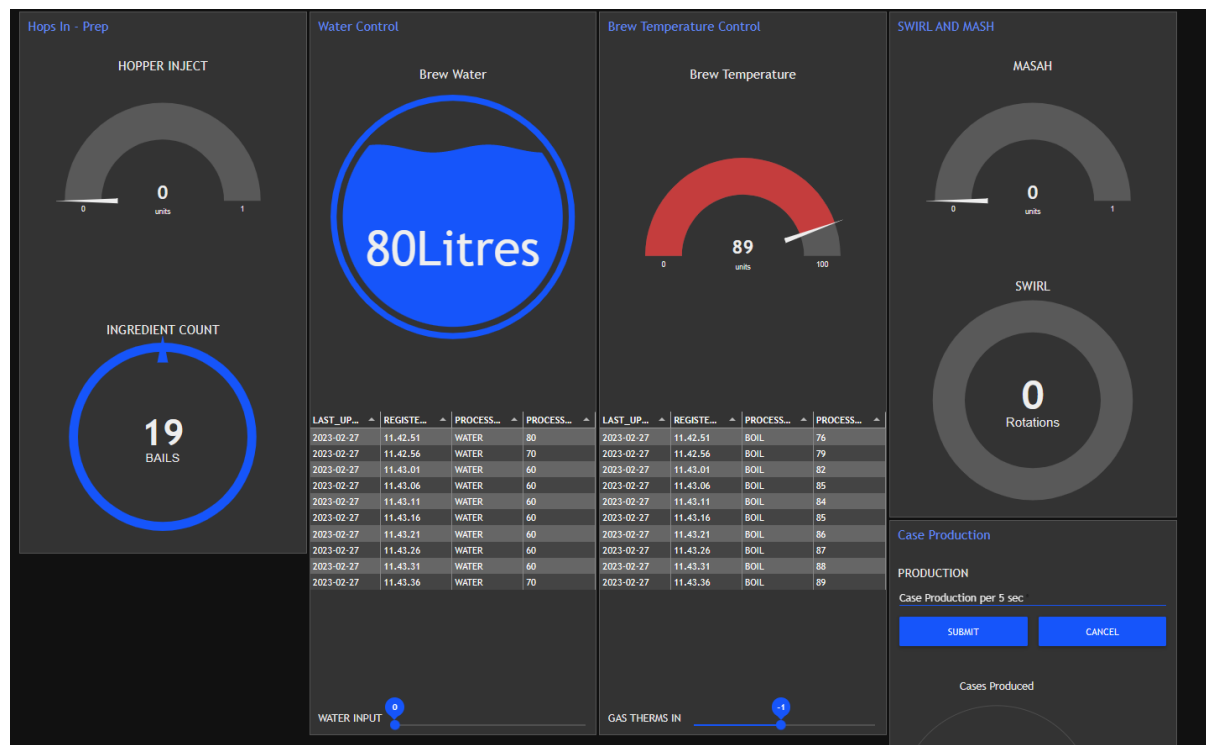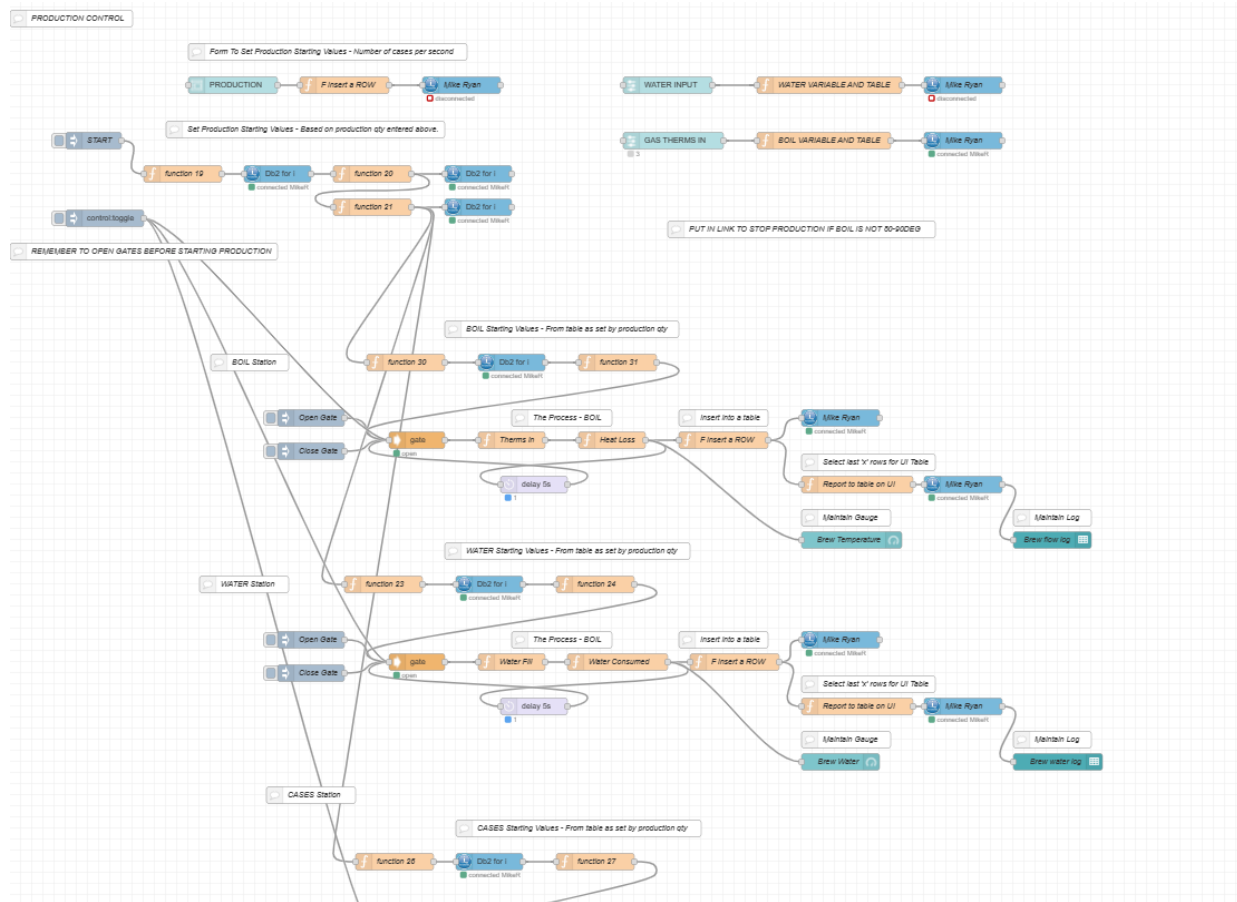i-UG Open Source Education
for IBM i

# Node-Red

The Brewery Challenge

i-UG

# 1. The Brewery Challenge

## 2. What is it all about?

The brewery is just a simple example of a process that might be used in manufacturing.  We thought that  Brewery had a bit more of a ring to it than a Papier Mache production line for producing hospital pulp urine bottles and sputum cups…

The whole process is dependent on what is coming out at the end.  We hope.  Not all of the processes are critical but 3 are: -

1.   The number of Cases out per 5 seconds

     We have kept the maths super simple as this is not a maths test, it is a logic test.  The Brewery is geared to produce Cases of 10 x 1 litre bottles.  In the beginning, we are looking to produce 1 case every 5 seconds.  These Cases are loaded onto a Van in multiples of 10 (So, 100 bottles) and then another van arrives for the next set of cases (You will need some imagination for this).

2.   The water for the brew

     Clearly, water is needed and, again keeping it simple, we have ignored losses and just made sure that 10 Litres are released every 5 seconds in order to meet the starting production target of 1 case per 5 seconds.

     Similarly, as we start, the water tank is filled at the same rate – 10 litres every 5 seconds.

3.   The Boil and Brew

     Cooking up the beer requires heat, and it is important that we keep this at a steady temperature of between 80 and 90 degrees… otherwise we get pretty poor beer!

     Here we measure in Therms (BTU).  We start with an input of 5 Therms per 5 seconds and have to accept that it will keep wanting to cool down as the new water enters the Boil and Brew process, so we lose 3 Therms for every 5 seconds of running.

     Needless to say, as we start, we need to get this up to temperature and then we need to maintain that temperature.

     Sound easy?

     Some base standards of the process.

     First of all, there is an interlock that will not allow the whole process to produce beer if the Boil station is not between 80 and 90 degrees C !

     There is a slider bar provided so that you can increase the "Therms in" and get up to temperature quickly… but beware, if you overcook it, it will go over 90 degrees C, and the process will stop.  Balance is crucial.
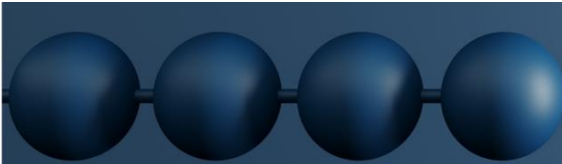
So, all is good. You will have found the balance for the temperature to keep it steady between 80 and 90 degrees C.

Look at each of the nodes in the Water Station, the Boil Station and the Cases Station and you will see how it initially picks up details of the starting water flow, temperature settings and Case production from the BREWCTL (Brewing Control) table, using SQL statements.
Look carefully at the various variables that are set up and used as the process runs. For Node-RED, these are not always necessary, but we have taken the stance that you would need these in any regular programming, so it is good to get used to these here.

You will note that each of the active stations (WATER, BOIL and CASES) update the database. They write records to the BREWTRANS (Brewing Transactions) Table. This is the record of how smooth the Brewery ran and where/when things went awry.

Understand all these well!

The next step is to increase production. Note well that as we move from 1 case per 5 seconds to 2 cases per 5 seconds, the heat losses increase (Multiplied by the case production number, in this case 2, therefore losses are now 6 Therms per cycle), whilst the input remains at 5 Therms per cycle… Clearly there will need to be some adjustment in order to maintain production.

Also, the amount of water needed will clearly increase. Again, the output (Losses) for the Water will increase by the case production number (In this case, 20 litres, will be required per cycle, but the input will still be set at 10 litres per cycle). Clearly some adjustment is needed here to maintain the input, and so, initially, the WATER slider bar can be used to increase the input.

This becomes increasingly difficult, and you will soon realise that twiddling with the Slider bars is just not the answer… especially when we increase production to 3, 4 or more crates per 5 seconds.

So, down to you.

You need to resolve this automatically. You need to update the database or update the variables in order to automatically compensate each station in order to smoothly produce Beer.

Good luck!

Here it is…

https://github.com/JoshRyanEOG/i-UG_Education_Node-Red_Brewery

## 3. Managing the Process

The information for each of these three stages is maintained in a Brewery Control table: -

| PROCESS_NAME | PROCESS_INPUT | PROCESS_LOSSES | PROCESS_ADJUSTMENT |
|---|---|---|---|
| WATER | 10 | 10 | 0 |
| BOIL | 5 | 3 | 0 |
| CASES | 1 | 1 | 0 |

A record of what is happening is kept in Brewery Transaction table.

| LAST_UPDATE | REGISTERED_AT | PROCESS_NAME | PROCESS_CURRENT_VALUE |
|---|---|---|---|
| 2023-01-25 | 10.05.50 | BOIL | 25 |
| 2023-01-25 | 10.05.55 | BOIL | 27 |
| 2023-01-25 | 10.06.00 | BOIL | 29 |
| 2023-01-25 | 10.06.05 | BOIL | 31 |
| 2023-01-25 | 10.06.10 | BOIL | 33 |
| 2023-01-25 | 10.06.15 | BOIL | 35 |
| 2023-01-25 | 10.06.20 | BOIL | 37 |
| 2023-01-25 | 10.06.25 | BOIL | 39 |
| 2023-01-25 | 10.06.26 | WATER | 80 |
| 2023-01-25 | 10.06.30 | BOIL | 41 |
| 2023-01-25 | 10.06.31 | WATER | 80 |
| 2023-01-25 | 10.06.35 | BOIL | 43 |