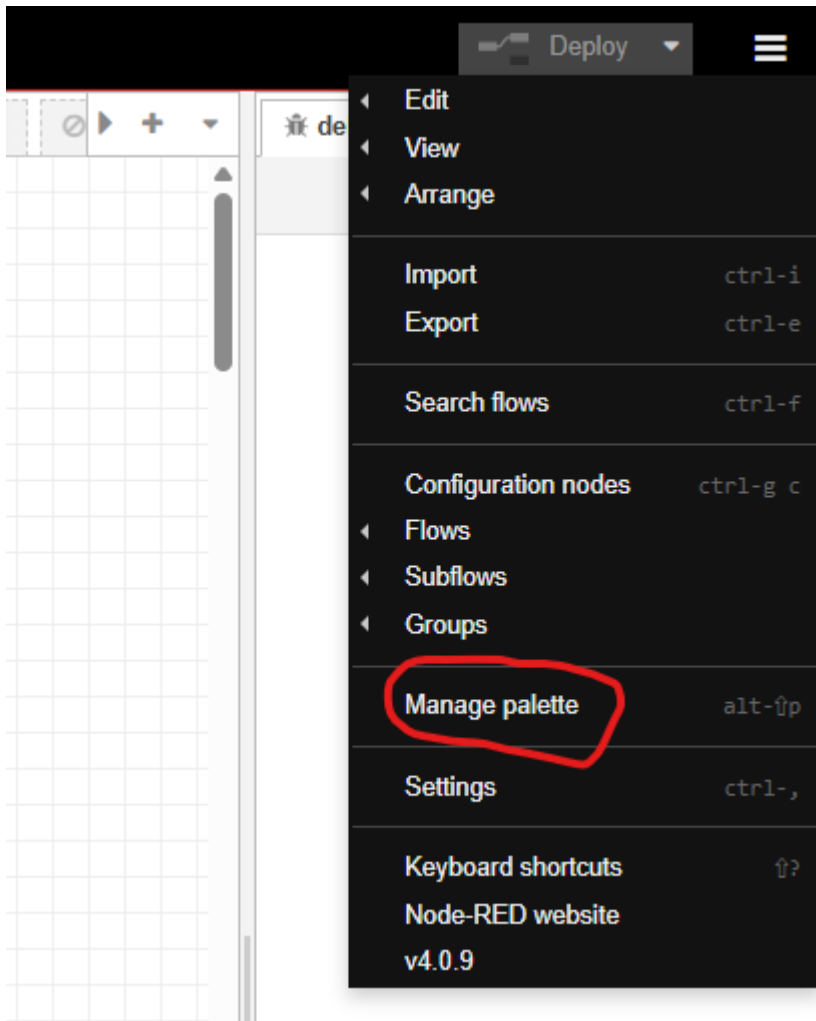
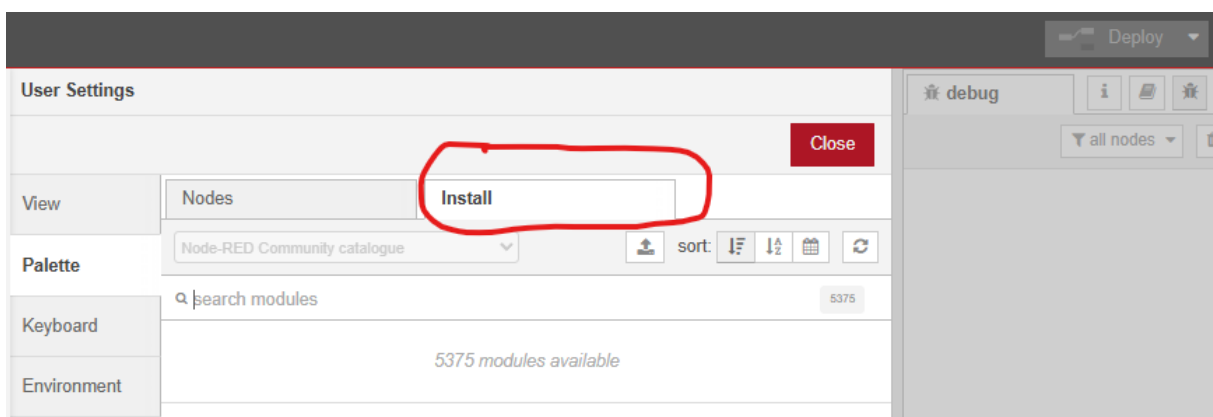


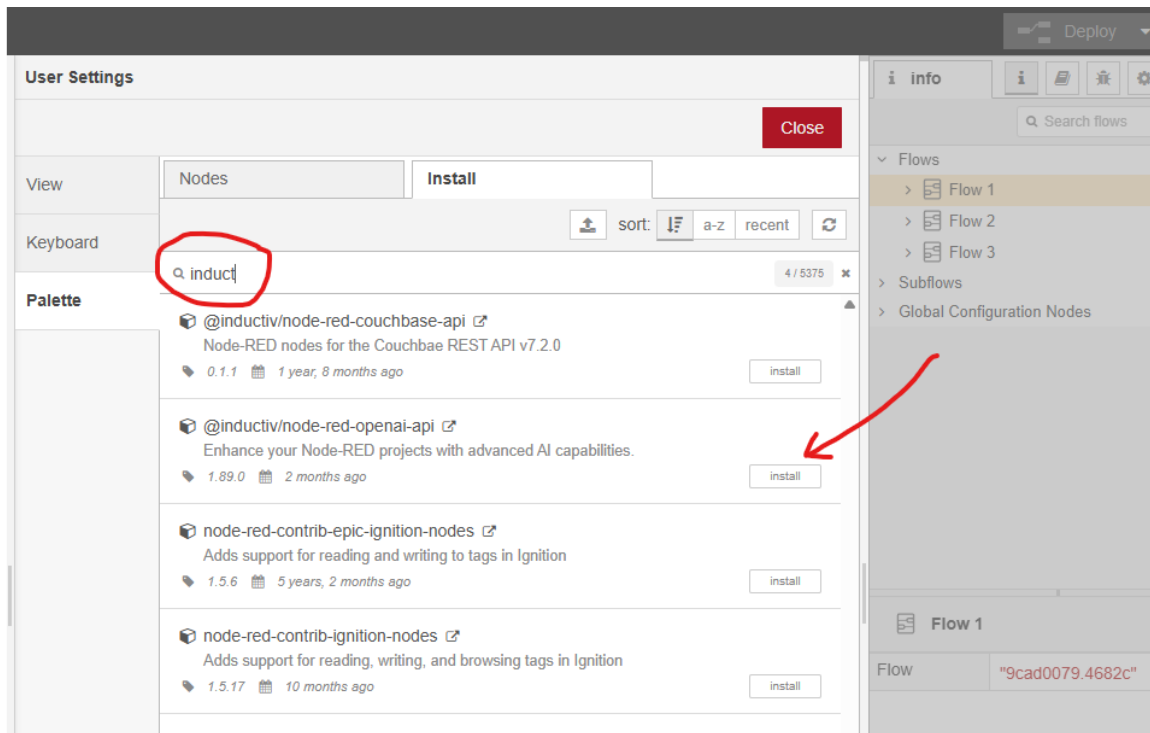
Installatie "OpenAI" node in Node-red, en maken van simpele flow gebruik makend van LM_Studio local server.



Select "Manage Palette"



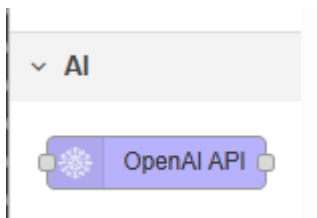
Select Tab "install"



Type in het zoek veld “induct”. Dan laat ie een aantal pakketten zien en kies dan de “openai” pakket.

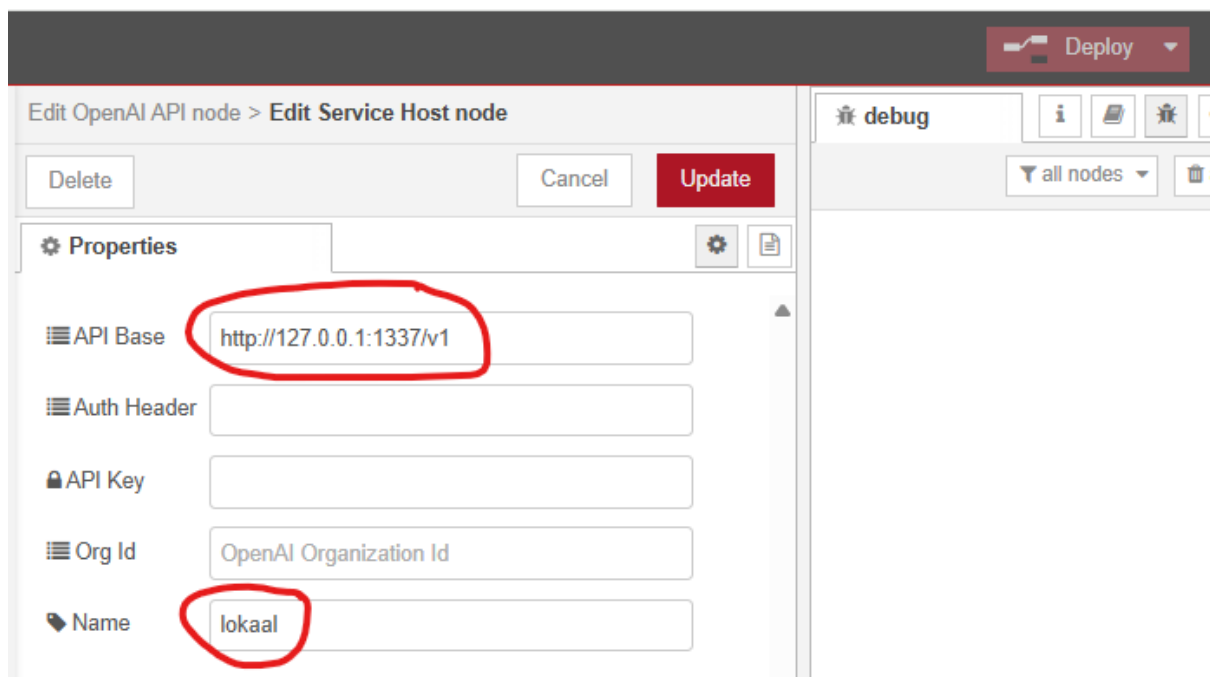
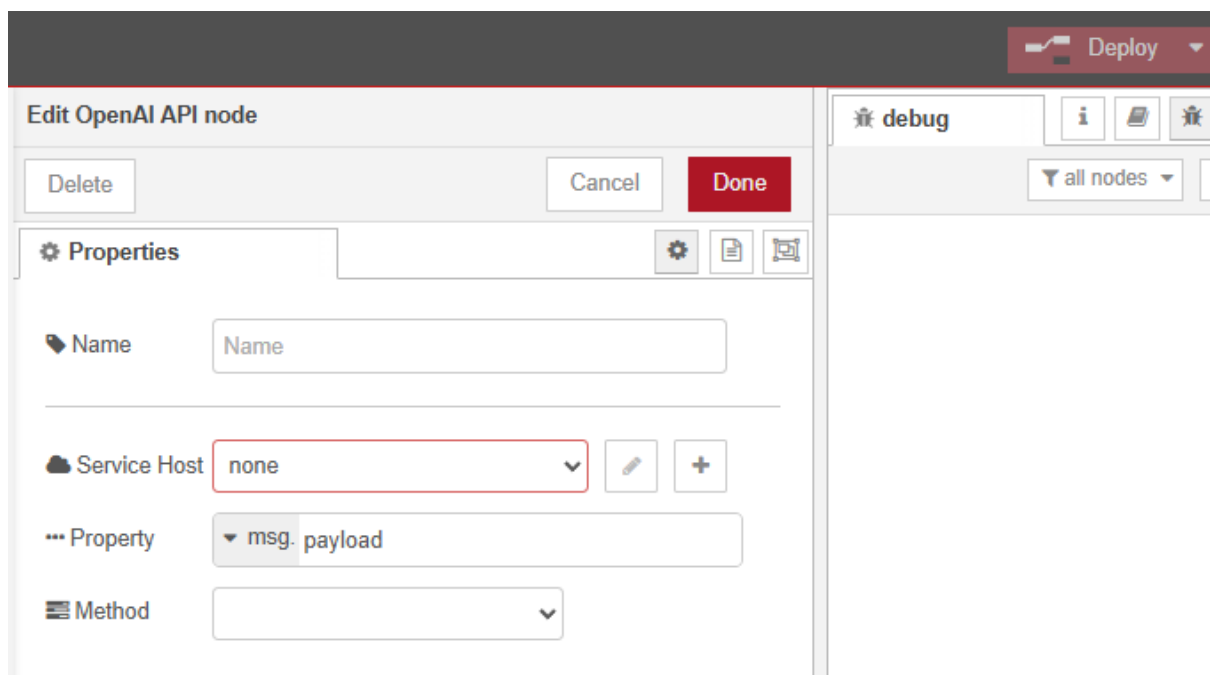
Type install. Nu wordt openai node geïnstalleerd.

Je krijgt nu onderstaande icon in je node lijst aan de linker kant van je scherm.



Sleep deze node in een nieuwe flow. Dan dubbelklik op de node in de flow :

Specificeer "Service Host", klik op de + icon.



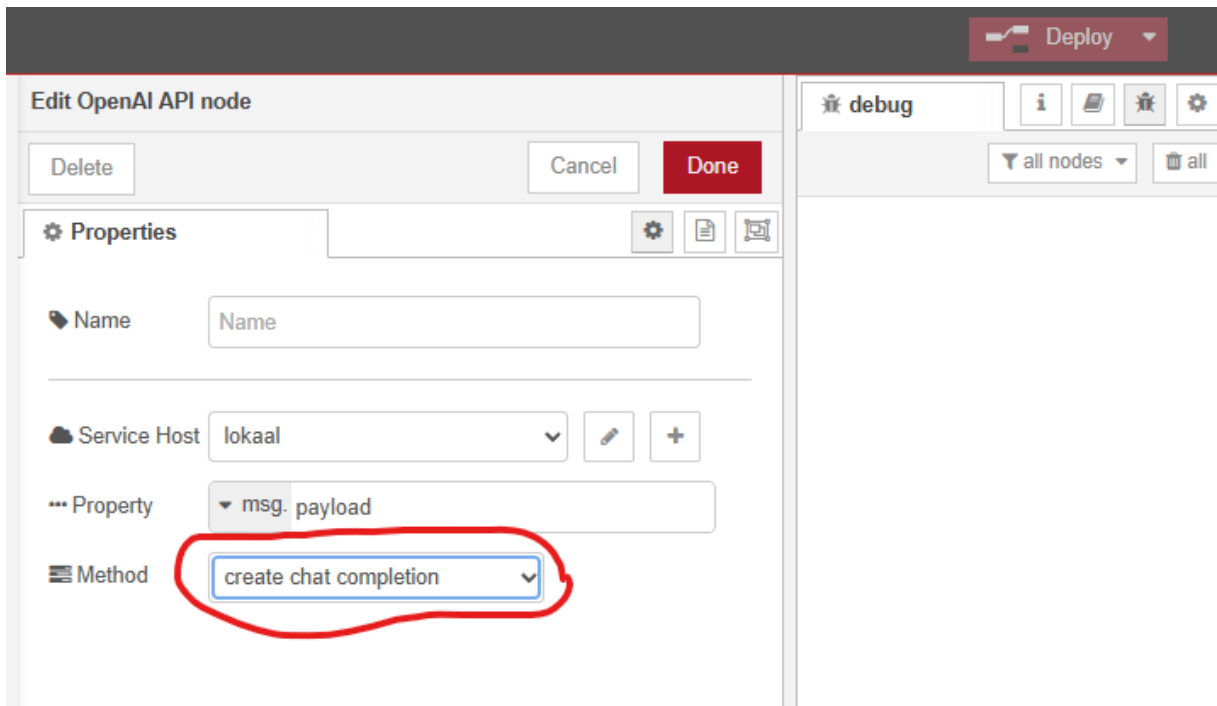
Vul API-Base in voor verbinding naar Lm_Studio in server mode

Geef het een naam "lokaal", maakt niet uit welke naam.

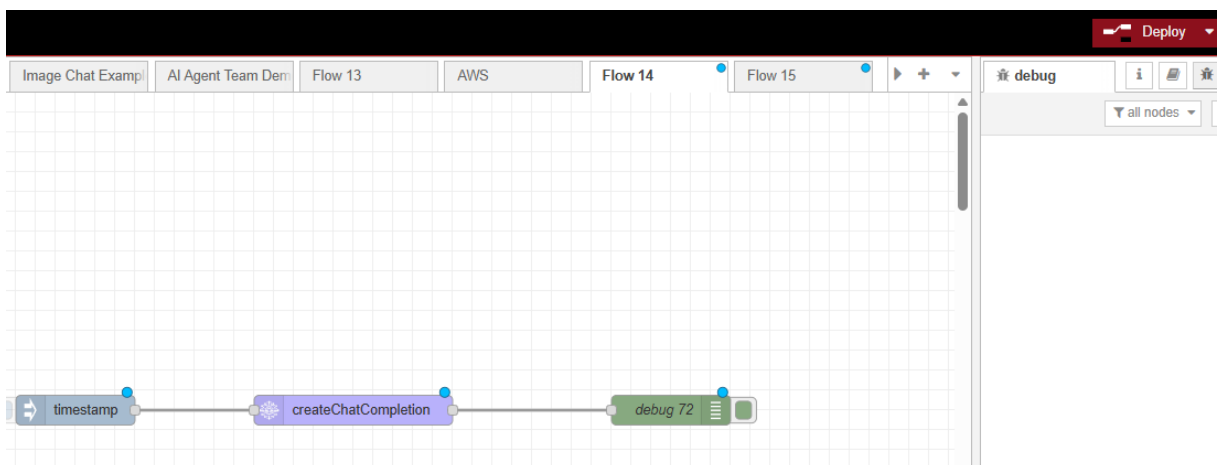
Klik op update.

In Method selecteer “create chat completion” Method.

Klik on done

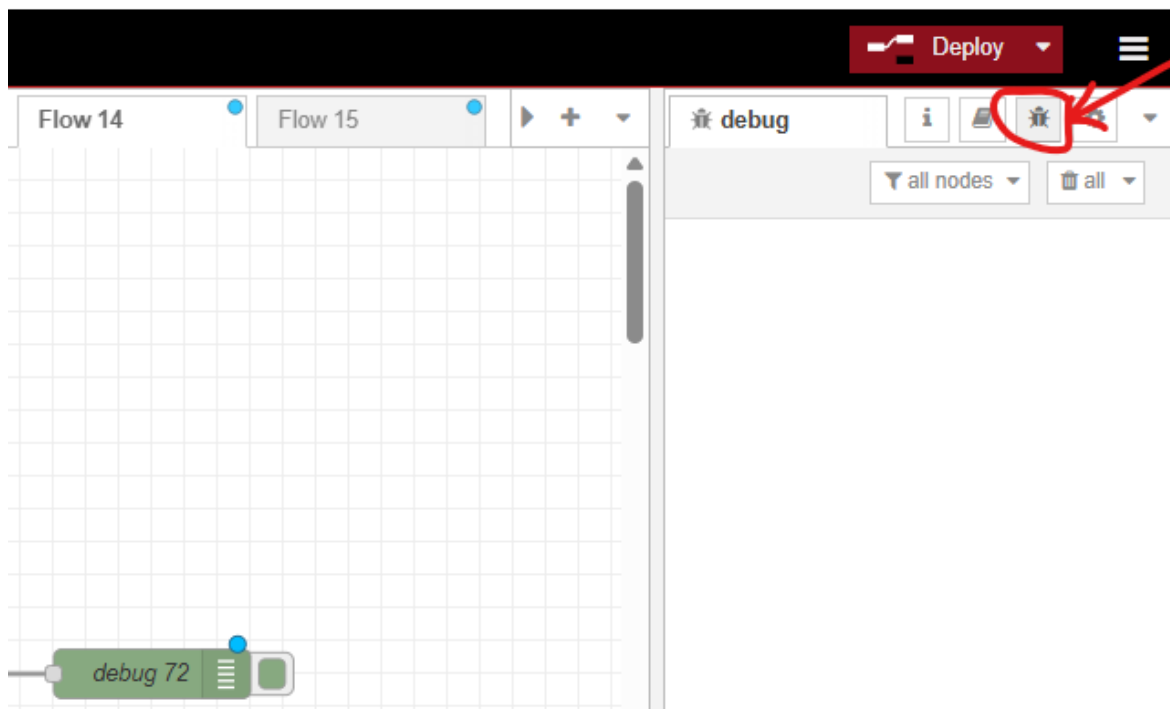


Sleep vanuit node lijst een “inject node” en een “debug node” in de flow. Maak de verbinding zoals in onderstaand voorbeeld

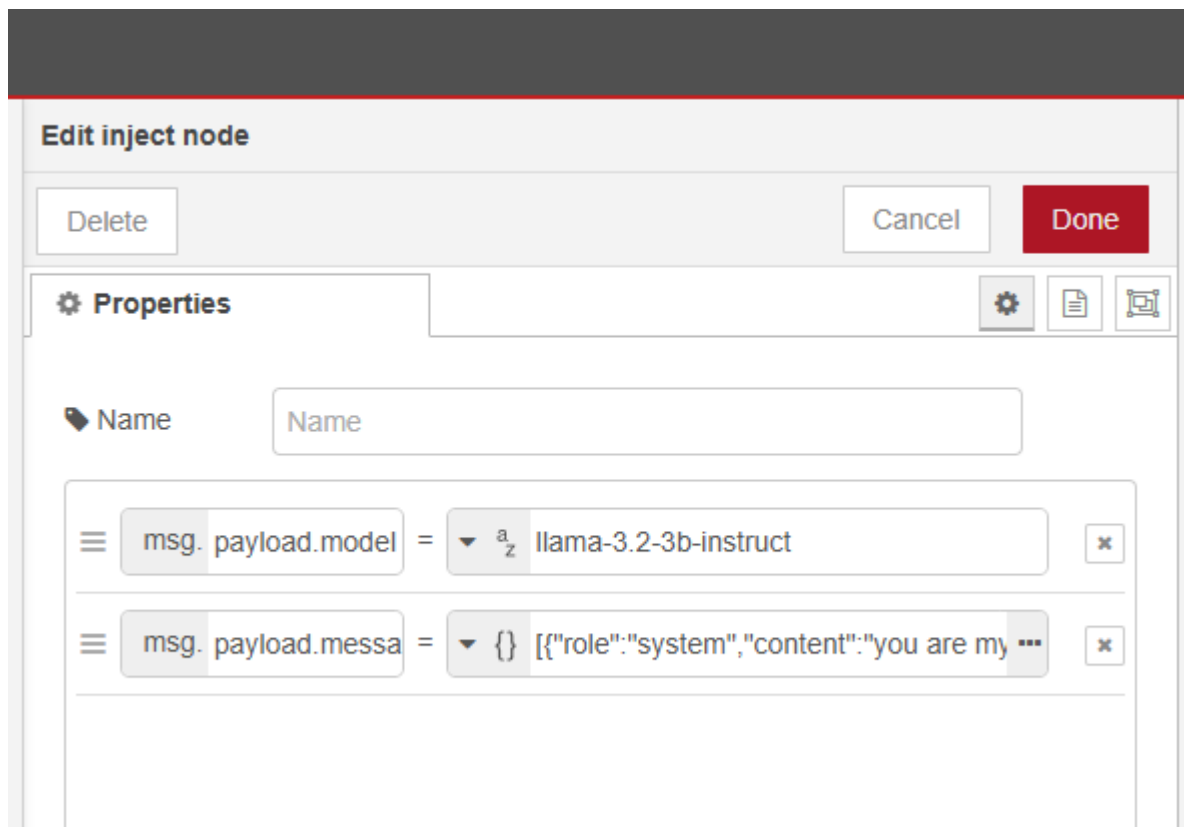


Klik on Deploy.

Zorg ervoor dat de debug window is geselecteerd.



Dubbel klik op de inject node :



Vul een model in die je hebt geïnstalleerd in LM_Studio.

Vergeet niet om msg.payload te veranderen naar msg.payload.model

```
msg.payload.messa = { } [{"role":"system","content":"you are my ..."}]
```

Klik op de

Selecteer de voor het model "String"

Klik on +add

Verander msg.payload naar msg.payload.messages

```
msg.payload.messa = { } [{"role":"system","content":"you are my ..."}]
```

Klik op de

The screenshot shows the 'Edit inject node' dialog with a 'Properties' section. The 'Name' field is empty. Below it, there are two property entries. The first is 'msg.payload.model' with a dropdown menu showing 'llama-3.2-3b-instruct'. The second is 'msg.payload.messa' with a dropdown menu showing a JSON object: `{ } [{"role":"system","content":"you are my ..."}]`. A red circle highlights the 'JSON' option in the dropdown menu. Other options in the menu include 'flow.', 'global.', 'string', 'number', 'boolean', 'buffer', 'timestamp', 'expression', 'env variable', and 'msg.'.

Klik op de 3 puntjes en copy paste onderstaand tekst.

```
[
  {
    "role": "system",
    "content": "you are my assistent"
  },
  {
    "role": "user",
    "content": "Good morning"
  }
]
```

Klik on "done"

Klik on "Deploy"

Dubbelklik op de debug node. (groen)

Verander zoals screenshot :

The screenshot shows the 'Edit debug node' window. The 'Output' dropdown is set to 'msg.payload.choices[0].message.content', which is circled in red. The 'To' section has 'debug window' selected. The 'Name' field is 'debug 17'. The right panel shows the debug console output: '16-5-2025, 10:32:46 node: debug 17' and 'msg.payload.choices[0].message.content : string[124]' followed by the message: 'Good morning! How can I help you today? Would you like some assistance with anything in particular, or just a friendly chat?'.

Klik op done

Klik op Deploy

Klaar... Als je op de button klikt bij inject node, krijg je aan de rechterkant het antwoord van LLM.

