Creation of function

Steps needed to deploy a python Azure Function.

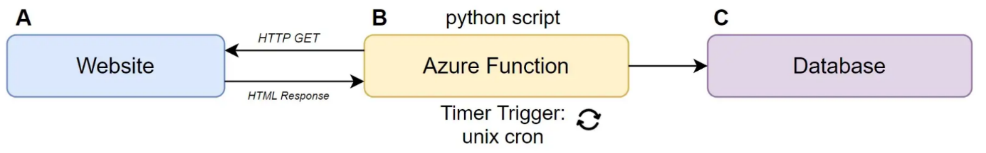
You need to install [Azure CLI](https://learn.microsoft.com/en-us/cli/azure/install-azure-cli-windows?tabs=azure-cli) and [Azure Functions Core Tools](https://learn.microsoft.com/en-us/azure/azure-functions/functions-run-local?tabs=v4%2Cwindows%2Ccsharp%2Cportal%2Cbash#install-the-azure-functions-core-tools)

|  |  |  |
| --- | --- | --- |
| 1 | az login | Login to azure (u will be switched to browser) |
| 2 | func init ProcessFeeFile --python | creates a sub-folder with some files |
| 3 | cd ProcessFeeFile | switch to new sub-folder |
| 4 | conda create -n azure\_func\_env | If not done before |
| 5 | conda activate azure\_func\_env |  |
| 6 | conda install pip | to be able to install py packages |
| 7 | set AZ\_LOCATION=westeurope  set AZ\_RESOURCE\_GROUP\_NAME=rg-itr-test-001  set AZ\_PLAN=FREE  set AZ\_FUNC\_NAME=funcProcessFeeFile  set AZ\_FUNC\_STORAGE\_NAME=assetmanagerfees | set variables |
| 8 | func new ^  --name %AZ\_FUNC\_NAME% ^  --template "HTTP trigger" ^  --authlevel "anonymous" | this creates a subfolder in subfolder |
| 9 | edit requirements.txt | file appear in the folder after step 8 |
| 10 | pip install -r requirements.txt |  |
| 11 | ***code development stage*** |  |
| 12 | func start | try locally |
| 13 | az account set --subscription firefly-westeurope-prod | If needed: switch to another subscription |
| 14 | az group create ^  --name %AZ\_RESOURCE\_GROUP\_NAME% ^  --location %AZ\_LOCATION% | If needed: creating resource group |
| 15 | az storage account create ^  --resource-group %AZ\_RESOURCE\_GROUP\_NAME% ^  --name %FUNC\_STORAGE\_NAME% ^  --sku Standard\_LRS | If needed: creating storage account |
| 16 | az functionapp create ^  --consumption-plan-location %AZ\_LOCATION% ^  --resource-group %AZ\_RESOURCE\_GROUP\_NAME% ^  --runtime python ^  --runtime-version 3.9 ^  --functions-version 4 ^  --name %AZ\_FUNC\_NAME% ^  --os-type linux ^  --storage-account %AZ\_FUNC\_STORAGE\_NAME% | Create Azure function |
| 17 | func azure functionapp publish %AZ\_FUNC\_NAME% | Deploy Azure function |

Very detailed on create locally in VS Code Python Azure functions and deploy + deploying API in Azure functions app ([article](https://medium.com/@nowitsanurag/deploying-api-in-azure-functions-app-f6d7feaa4308))

Azure Functions HTTP trigger ([microsoft](https://learn.microsoft.com/en-us/azure/azure-functions/functions-bindings-http-webhook-trigger))

Deploy Function which periodically sends GET HTTP request to website via VS Code and CLI ([article](https://python.plainenglish.io/serverless-python-bot-using-azure-function-90da5de713f7))





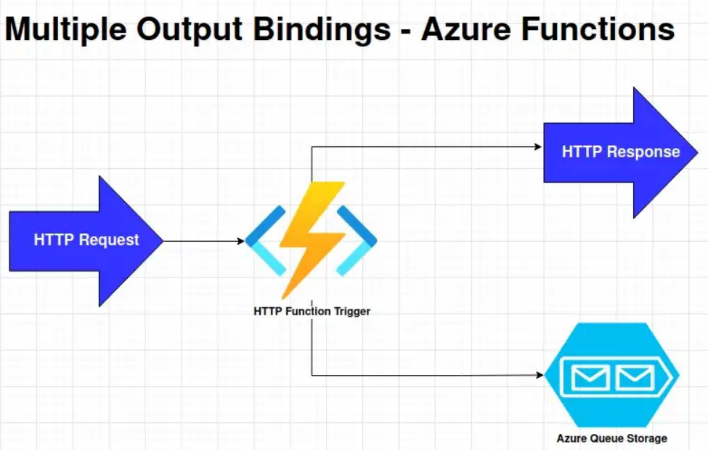
Azure Functions triggers and bindings concepts: <https://learn.microsoft.com/en-us/azure/azure-functions/functions-triggers-bindings>

Another example of deploying a Function from VS Code with more info on Bindings ([Article](https://medium.com/geekculture/getting-started-with-azure-functions-19e96257ec0f))



Deploying Function with several Bindigs ([article](https://rajesh-r6r.medium.com/efficient-azure-functions-with-multiple-bindings-772d58853494))

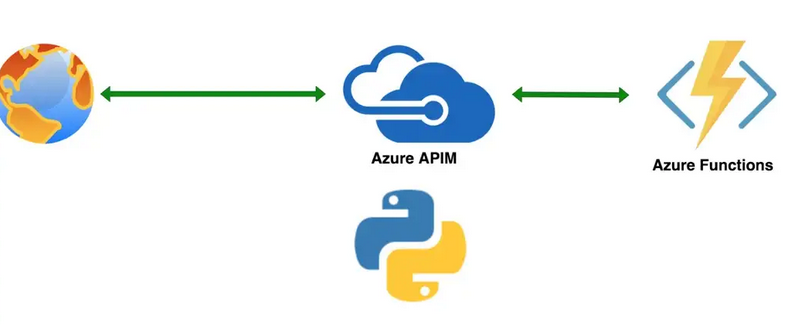




Link Azure Function with a static web app (CLI command). This adds entry level auth of the AZ function.

Automate user notifications in MS Teams using Webhooks, Azure Service Bus and Azure Function App ([article](https://medium.com/@error505/automate-user-notifications-in-microsoft-teams-using-incoming-webhooks-azure-service-bus-and-azure-ac59bf28b6eb))

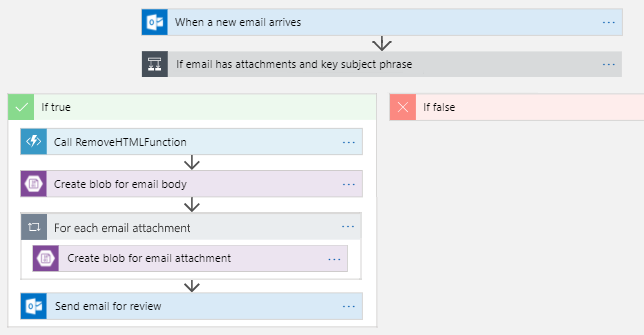
Add API gateway for your Python APIs Running on Azure Functions ([article](https://medium.com/bb-tutorials-and-thoughts/how-to-add-api-gateway-for-your-python-apis-running-on-azure-functions-d26aa33583fb))



Additional [article](https://medium.com/bb-tutorials-and-thoughts/configuring-auth-2-0-in-apim-for-python-apis-running-on-azure-functions-63d074744bcf) of the same author about adding Oauth 2.0

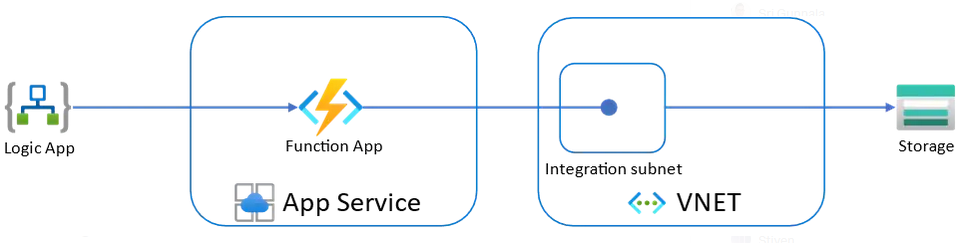
Functions + Logic Apps:

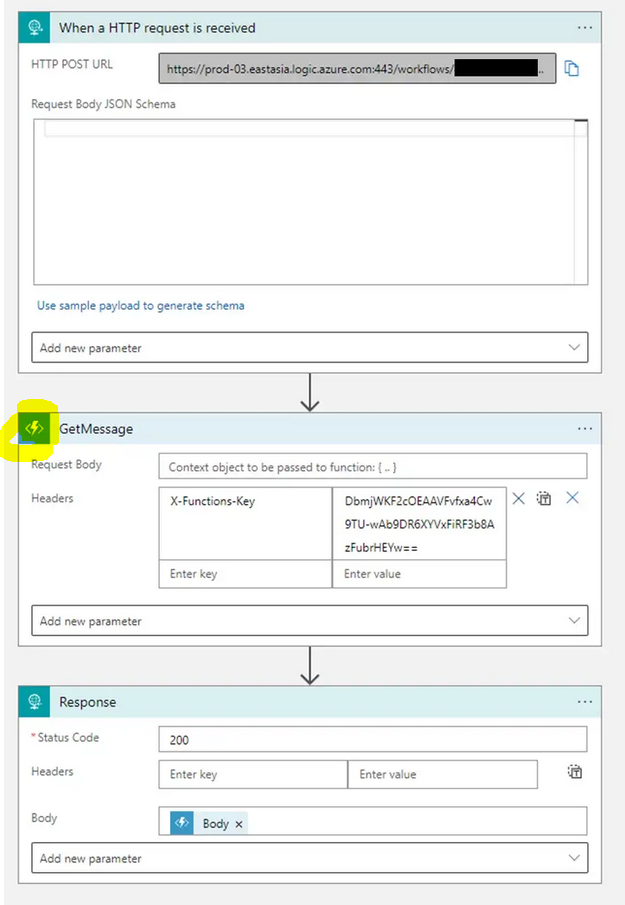
[Process emails by using Azure Logic Apps, Azure Functions, and Azure Storage](https://learn.microsoft.com/en-us/azure/logic-apps/tutorial-process-email-attachments-workflow)



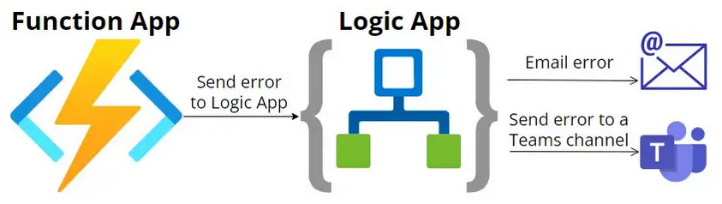
Azure Logic Apps access to Function App with Managed Identity ([article](https://medium.com/@wilsonyeung_41612/azure-logic-apps-access-to-function-app-with-managed-identity-12081e290535)):

consumption plan is that it doesn’t support VNET integration: the consumption plan logic app is hosted on a multi-tenant environment, which means you are sharing the host server with many other logic app users.

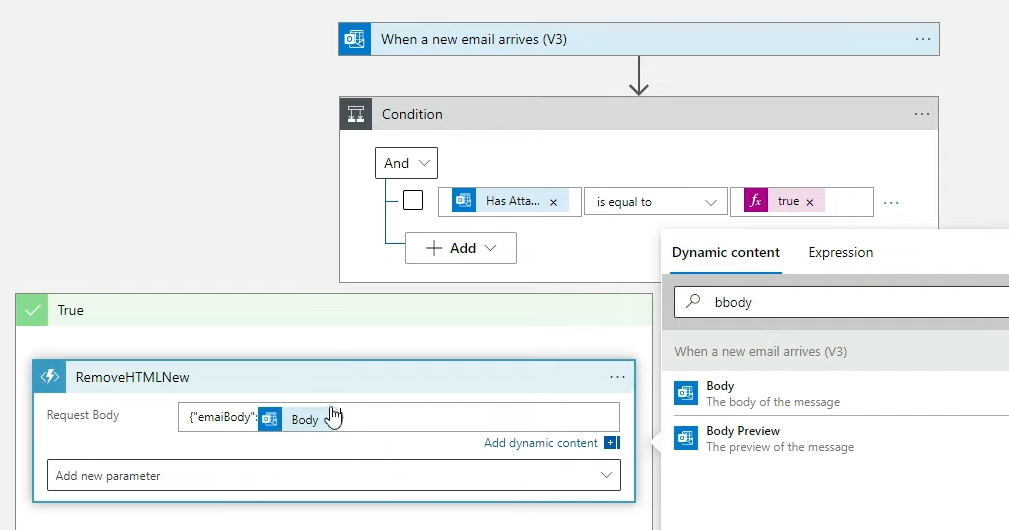


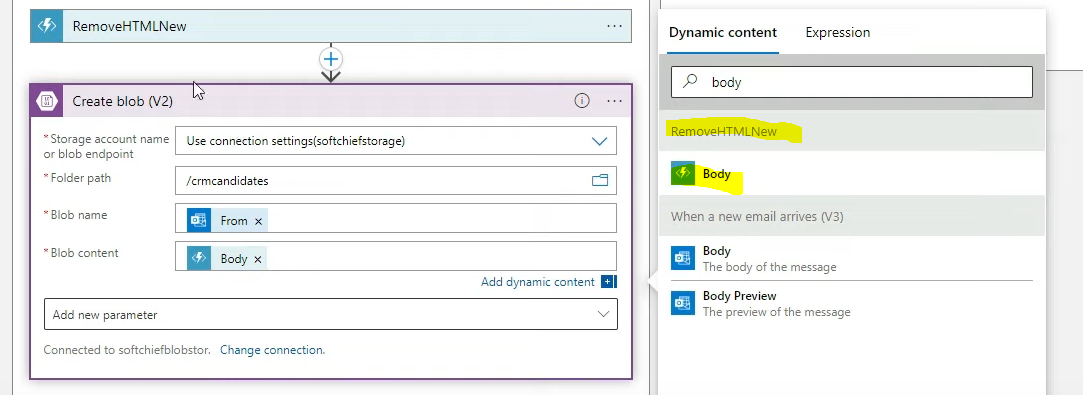


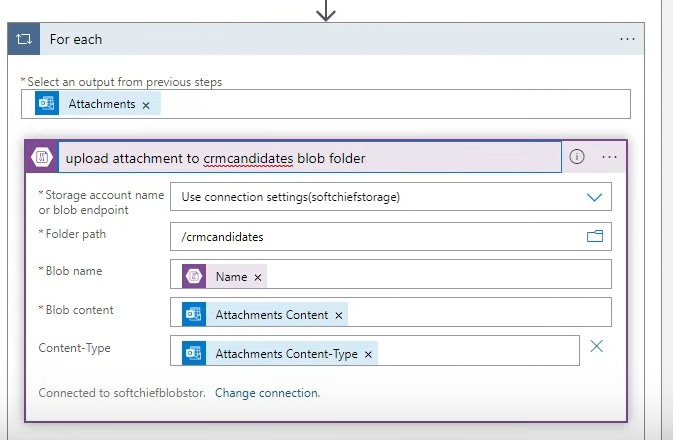
Catch an error in Azure Function and then email or send it to an MS Teams channel via Azure Logic App ([article](https://medium.com/@yamchi/how-to-catch-an-error-in-an-azure-function-and-email-or-send-it-to-an-ms-teams-channel-via-azure-418c7712b26))



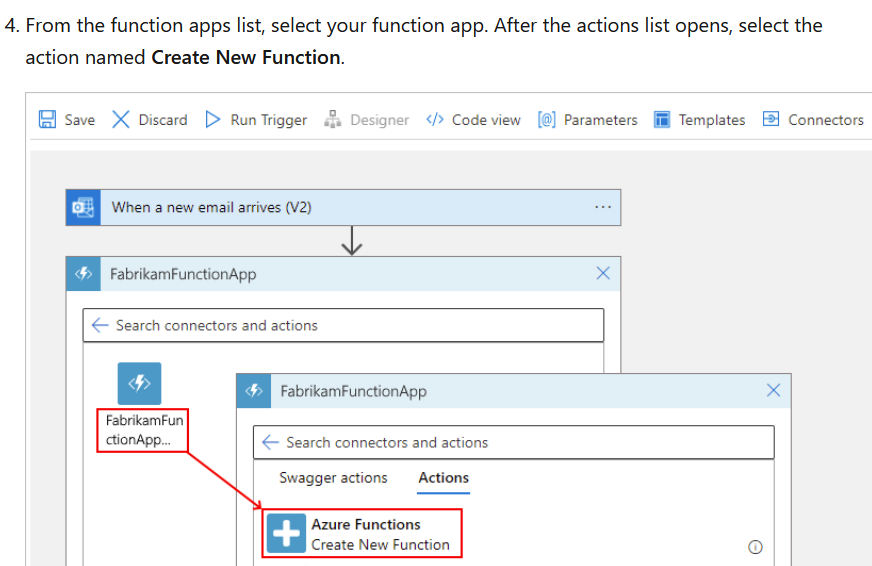
Logic App: Email => get email => process body of email with **Function** => store processed body in **Blob =>** store attachments in blob ([video](https://www.youtube.com/watch?v=Y_zcXHCHJuM&list=PLU0ornf0hw6c-9eXePWwrgguRST9lsOFt&index=1))

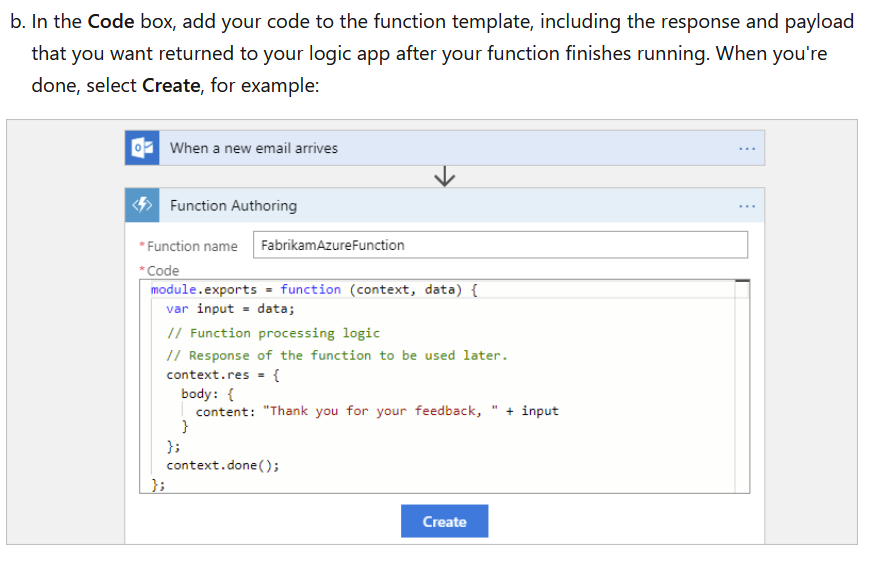




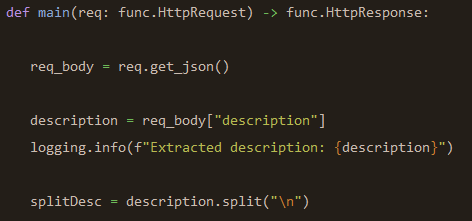


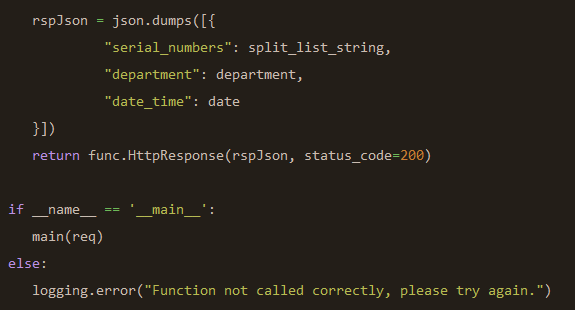
Create and run code from workflows in Azure Logic Apps using Azure Functions ([article](https://learn.microsoft.com/en-us/azure/logic-apps/logic-apps-azure-functions?tabs=consumption)):



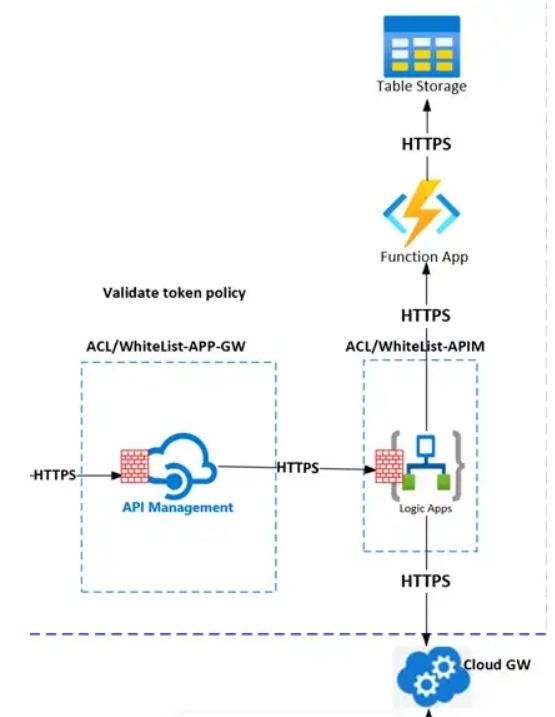


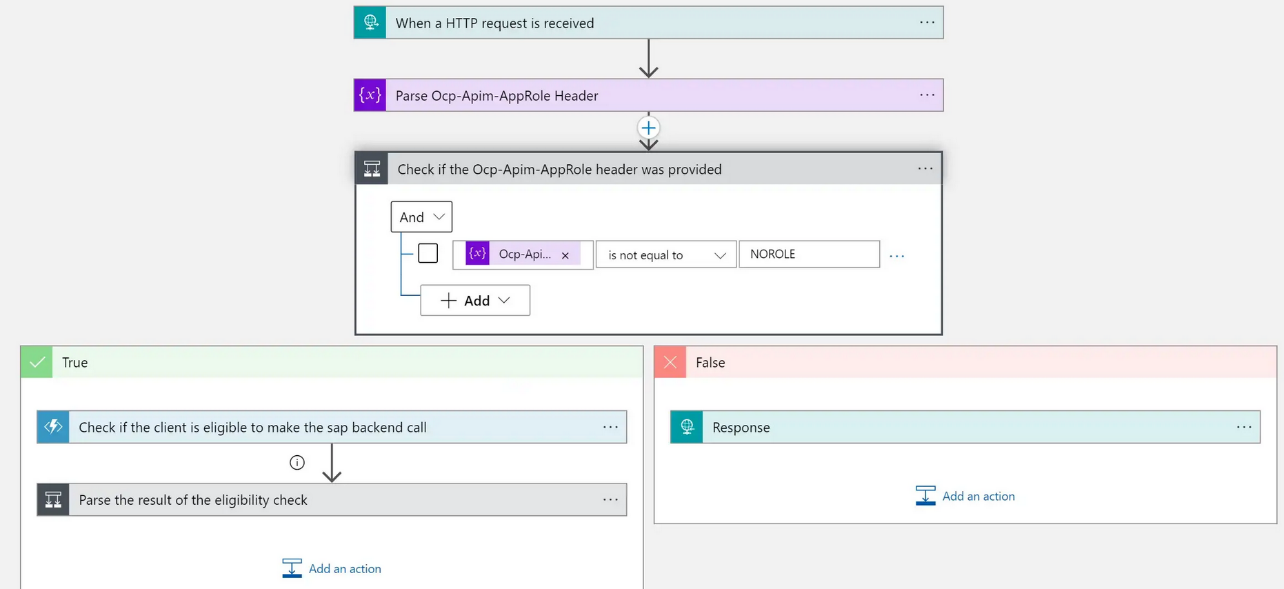
Logic Apps pass JSON payloads to Function App ([article](https://automationadmin.com/2021/01/function-apps-with-logic-apps/))

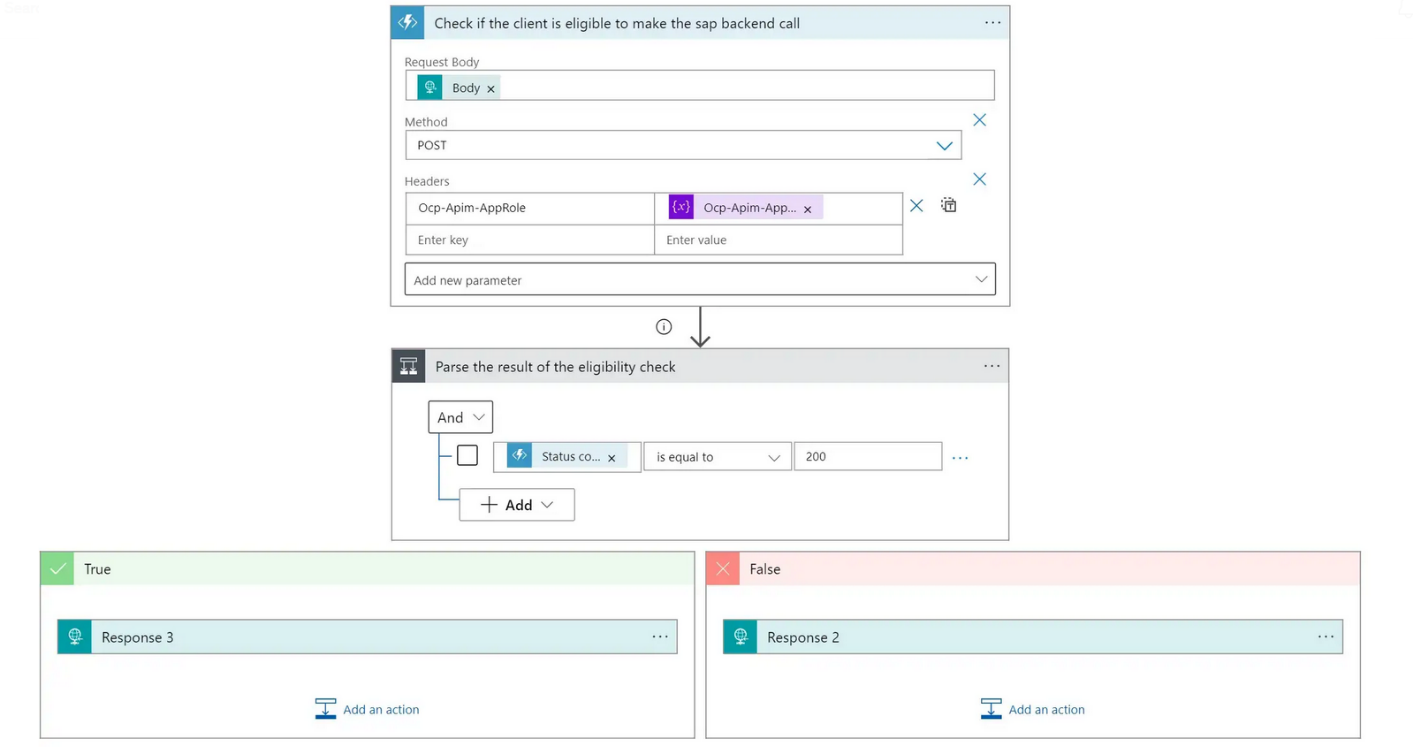


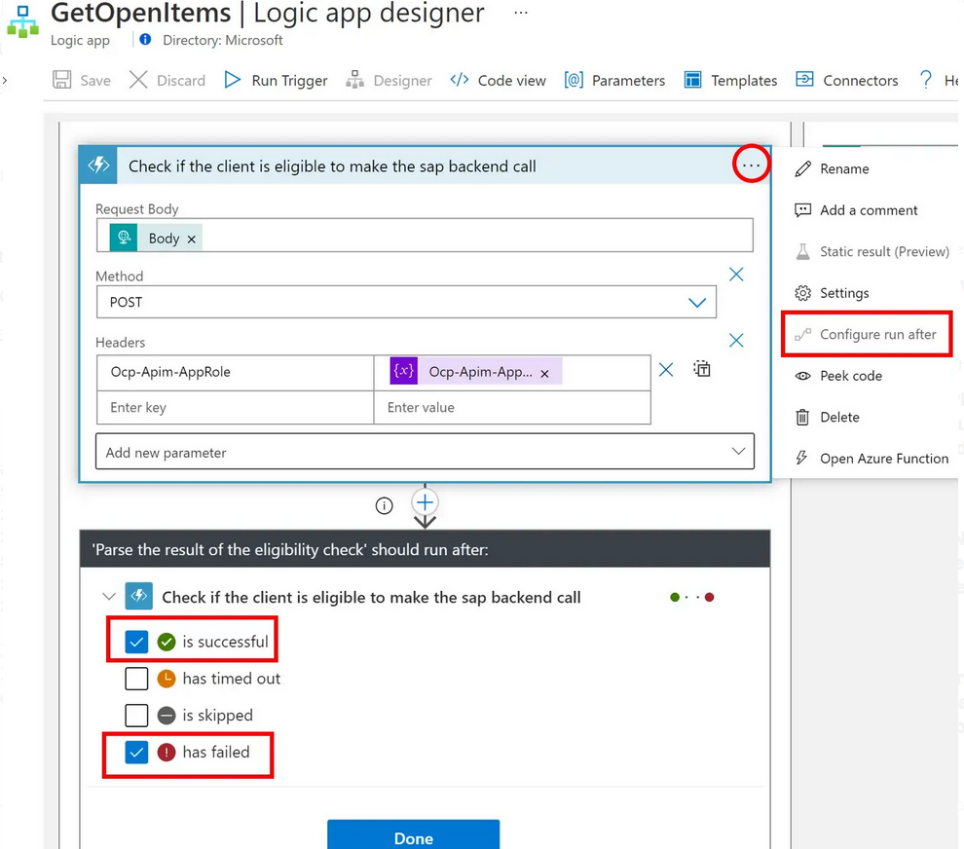


Function checks if Logic App is allowed to access Table Storage



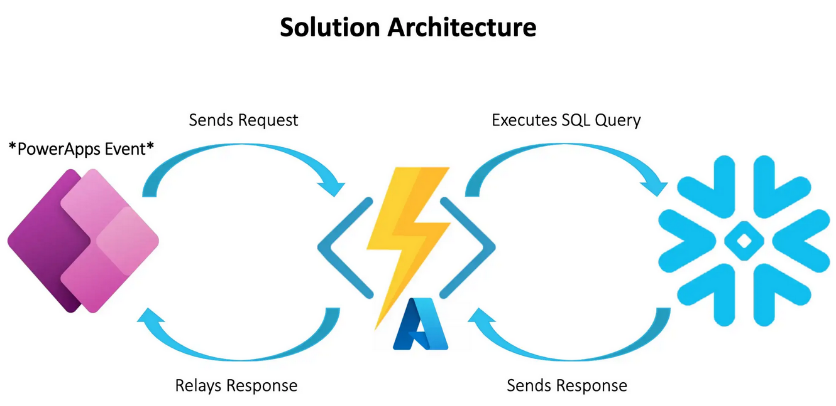






Power App + Logic Apps:

Power APP calls Azure Function via Custom Connector ([article](https://medium.com/@onepoint.au/how-to-connect-microsoft-powerapps-to-snowflake-8c6db40931a3)):



Connect Power App to Logic App ([article](https://learn.microsoft.com/en-us/azure/logic-apps/call-from-power-automate-power-apps)):

