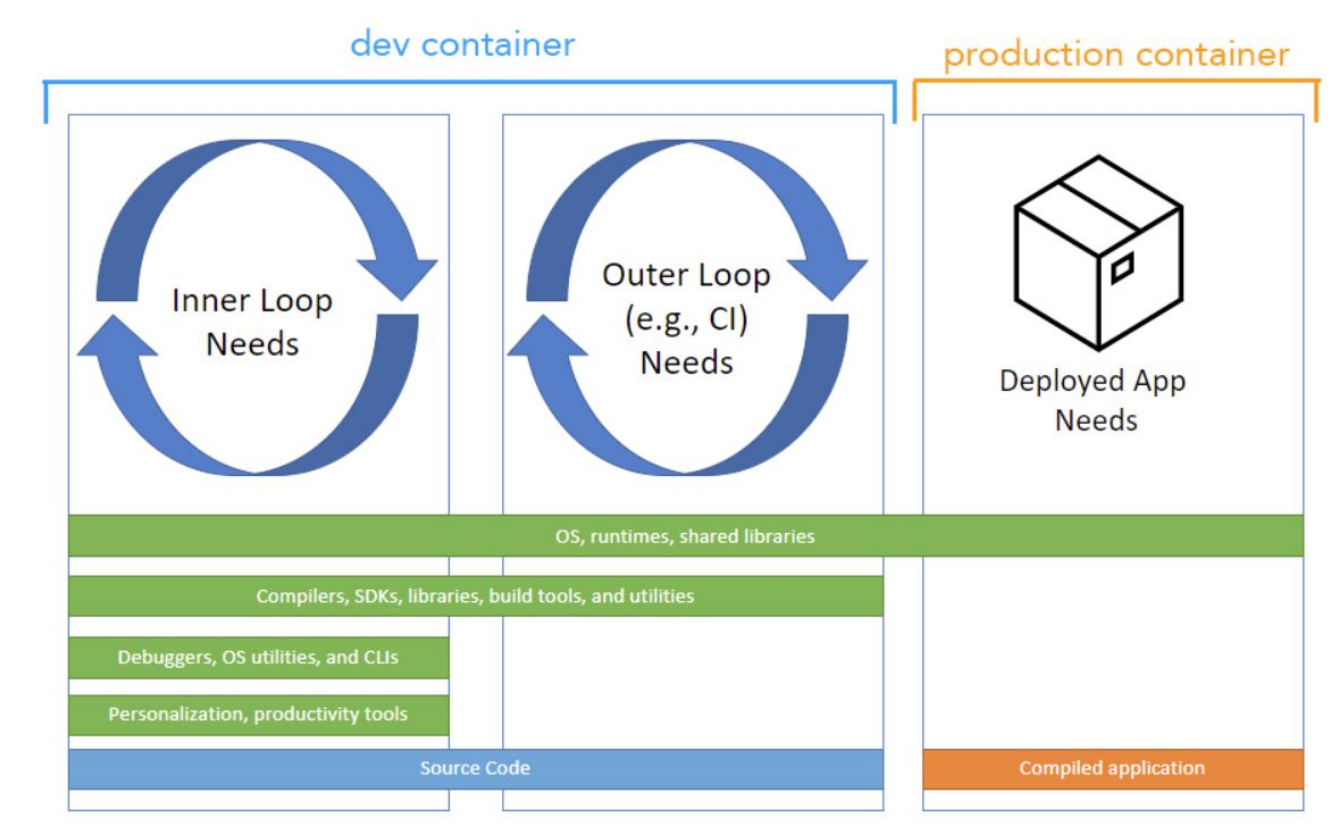
Flogo - Devcontainers
&
Azure Functions

What’s a devcontainer, and why use them?

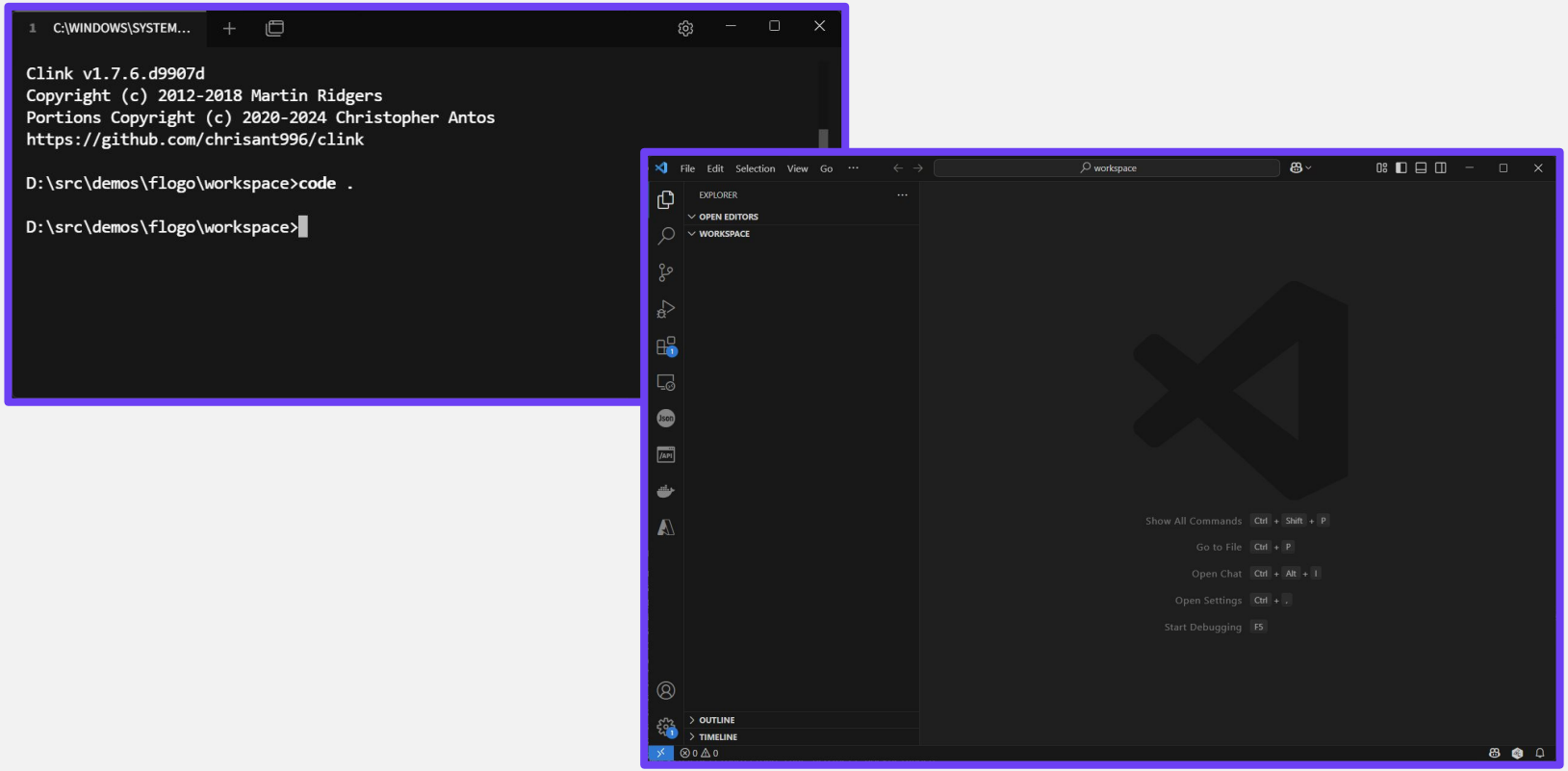
- A standardised portable environment.
- It’s a containerised setup for developers.
- Provides consistency across environments.
- Quick to setup and use.
- Isolation helps avoid conflicts between projects.
- Can be run in cloud environments.
- Can be customised to suit your project’s needs.
- Many pre-built images exist for popular toolchains.
- https://containers.dev/

TL;DR

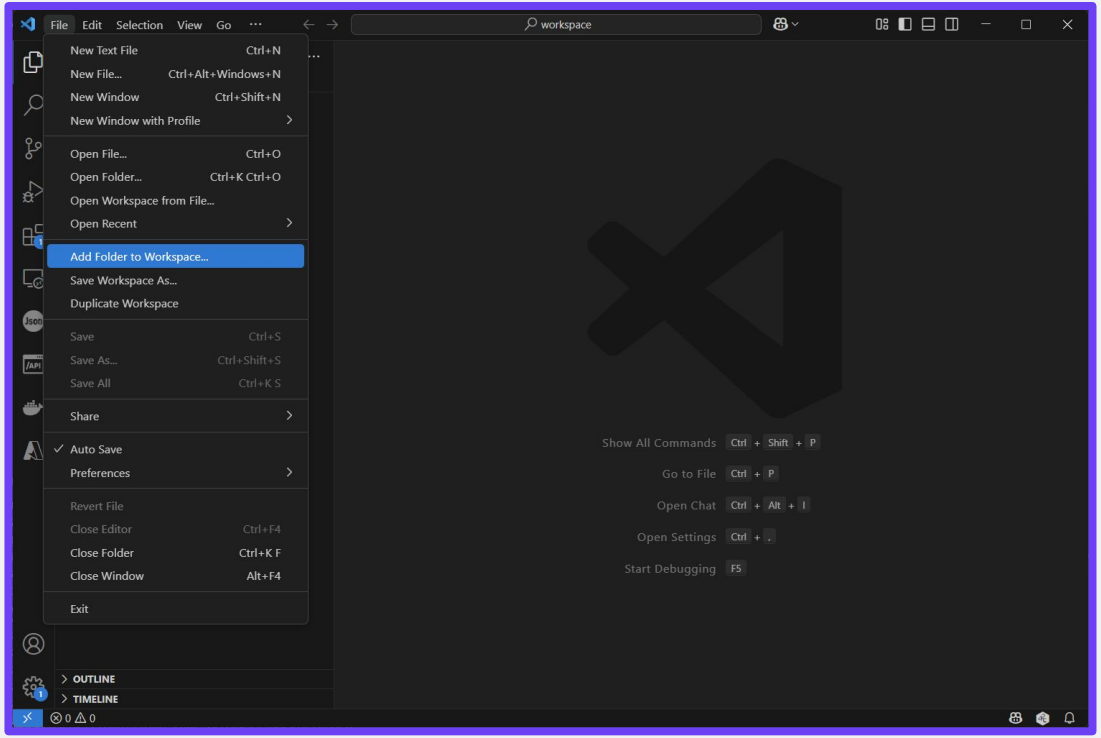
- You can run Flogo development in a container



Launch Visual Studio Code from workspace folder



Add Workspace Folder to Workspace



Create devcontainer.json

*Create a folder **.devcontainer** and add a new file **devcontainer.json** inside*

devcontainer.json 2

.devcontainer > devcontainer.json > ...

```
{
    "name": "FlogoDevContainer",

    "image": "mcr.microsoft.com/devcontainers/go:1-1.23-bookworm",

    "features": {
        "ghcr.io/devcontainers/features/azure-cli": {},
        "ghcr.io/devcontainers/features/docker-in-docker:2": {},
        "ghcr.io/jlaundry/devcontainer-features/azure-functions-core-tools": {},
        "ghcr.io/azure/azure-dev/azd:0" :{},
    },

    "forwardPorts": [
        8080,
        9999
    ],

    "customizations": {
        "vscode": {
            "extensions": [
                "golang.go",
                "ms-vscode.azurecli",
                "ms-azuretools.vscode-docker",
                "ms-azuretools.vscode-azurefunctions",
                "ms-azuretools.vscode-azureresourcegroups",
                "ms-azuretools.vscode-azurestorage",
                "${containerWorkspaceFolder}/.devcontainer/extensions/flogo-vscode-linux-x64-1.2.0-836.vsix"
            ]
        }
    }
}
```

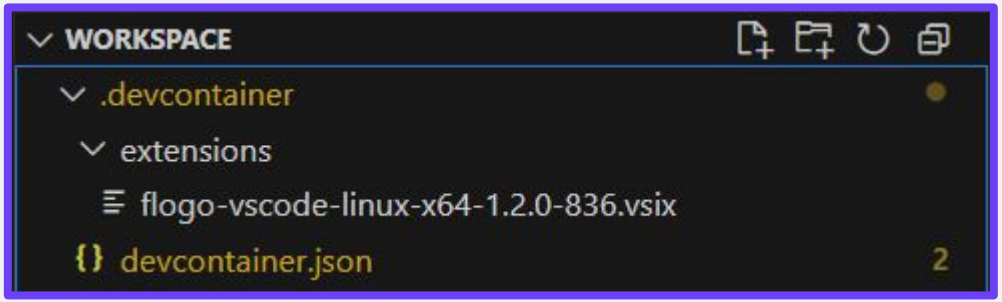
Features of the container

Extensions to VSCode

- Azure CLI
- Docker-in-docker
- Azure Functions Core Tools
- Azure Developer CLI
- Flogo Extensions

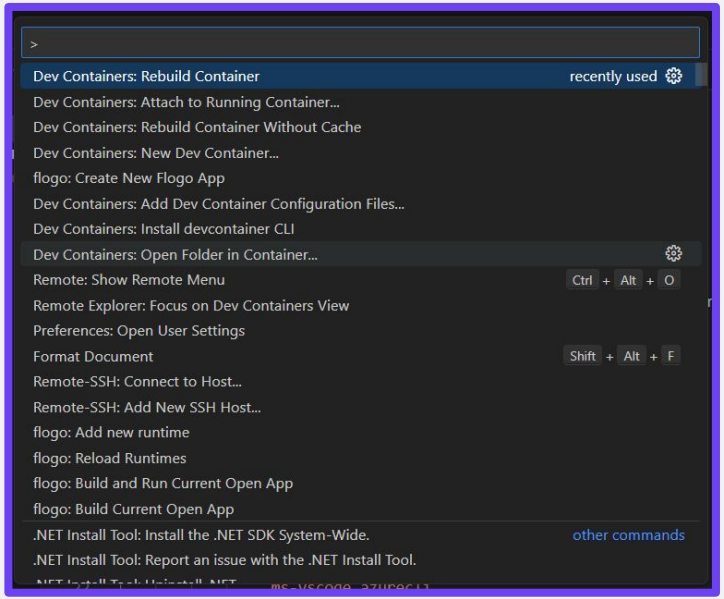
Add Flogo VSCode Extension

- Create a new folder ‘extensions’ within .devcontainer folder.
- Copy linux variant of Flogo VSCode Extension to extensions folder.
- Workspace should look something like this:



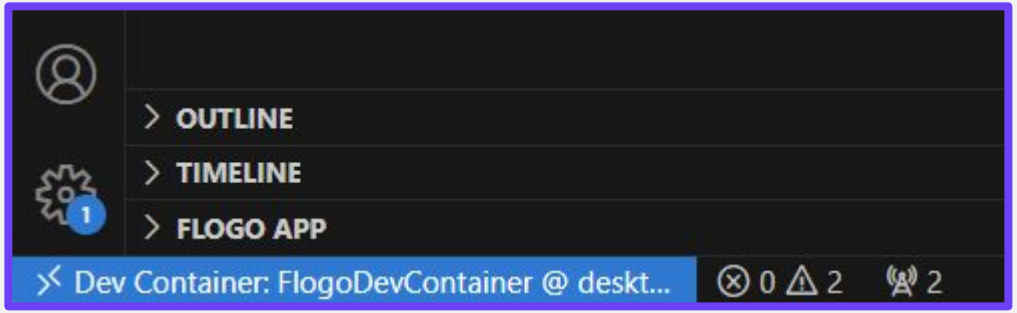
Build the Dev Container

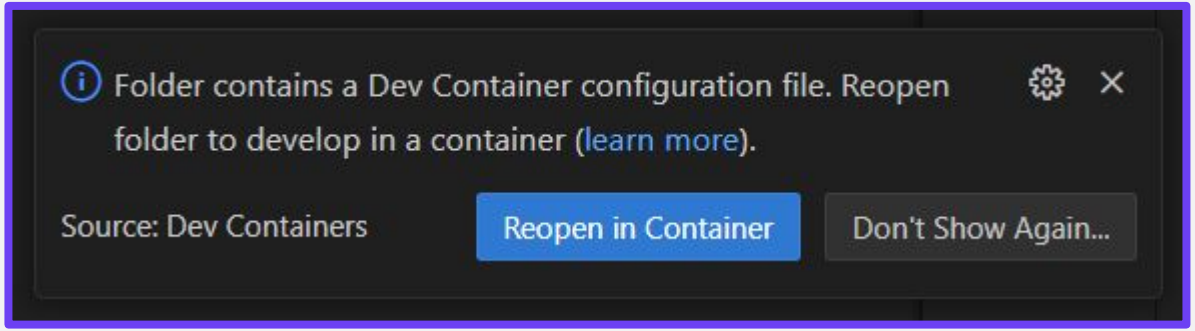
- CTRL+LEFT-SHIFT+P brings up command window
- Select ‘Dev Containers: Rebuild Container’

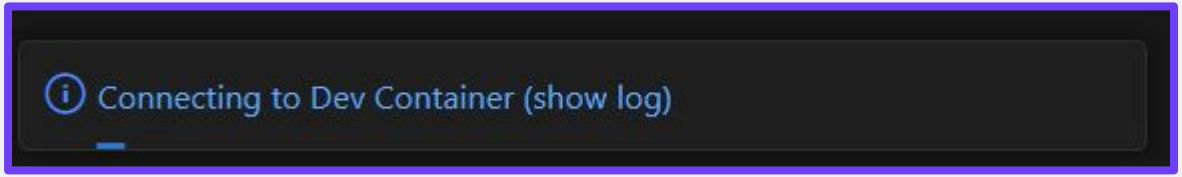


Build the Dev Container

Once the dev container is built VSCode will automatically connect to your remote container session running inside your newly created container:

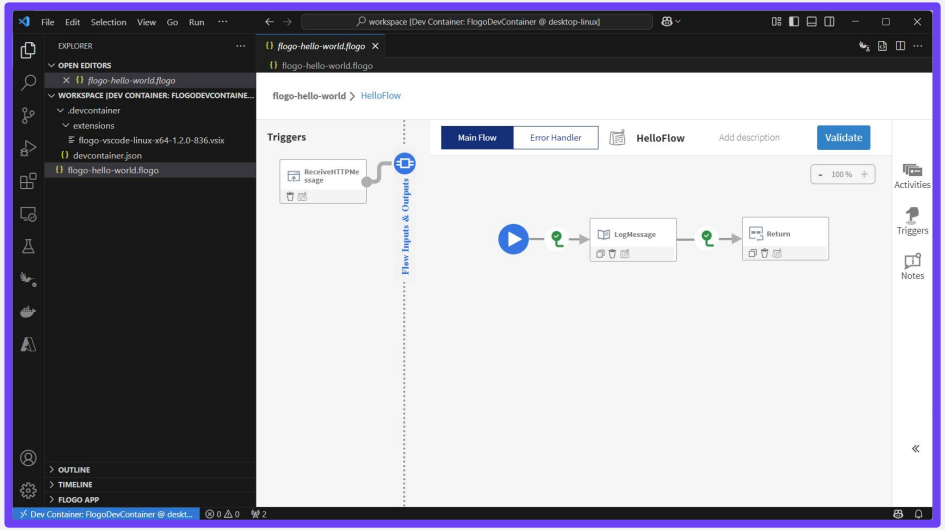






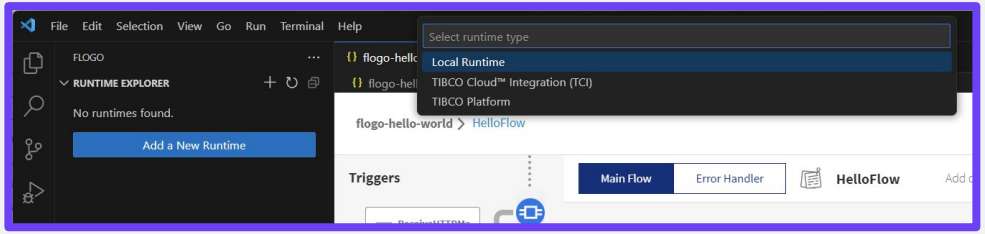
Create your Flogo Application

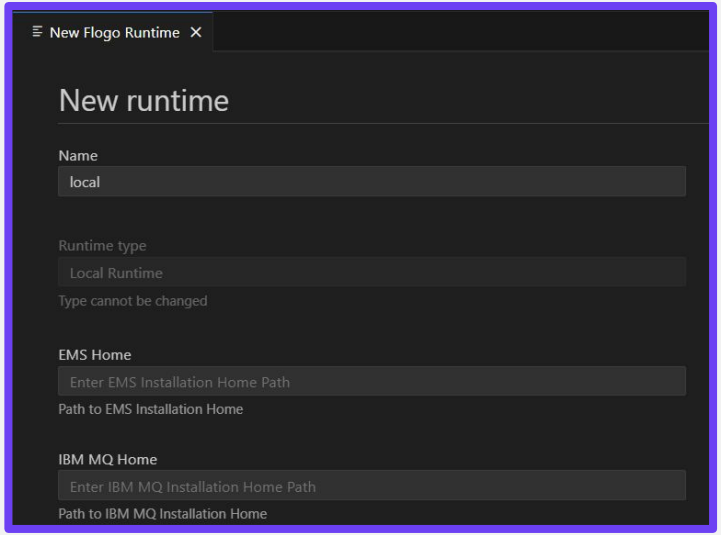
- Create new Flogo Application in the root of your workspace folder.
- Trigger
  - ReceiveHTTPMessage, use port 80 or 8080



Add a New Runtime

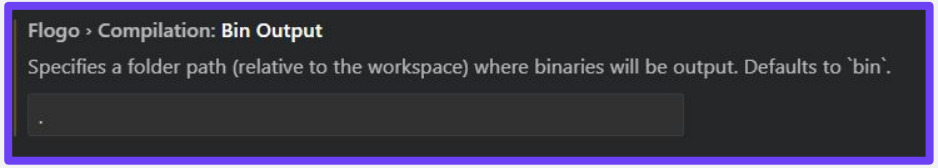
- Use local runtime to build binary application
  - OS: Linux, Architecture: amd64
- Select Flogo toolbar, Runtime Explorer, Add a New Runtime..



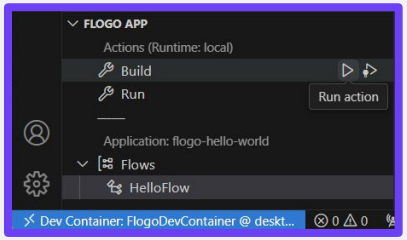


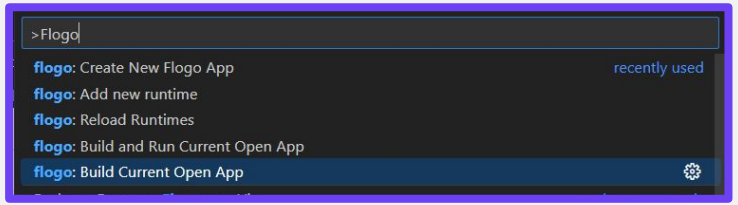
Build your Flogo Application

- Use local runtime to build binary application
  - OS: Linux, Architecture: amd64
- Change Settings Flogo->Compilation:Bin Output to write build to workspace root folder.



- Make sure to have your flogo application selected in explorer
- CTRL+LEFT-SHIFT+P or use the FLOGO APP Explorer window to run the build





Create a new Function App

- Use Azure Function Core Tools to create function app
  - func init --worker-runtime custom --docker

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS 2    AZURE

vscode ➜ /workspaces/workspace $ func init --worker-runtime custom --docker
Writing .gitignore
Writing host.json
Writing local.settings.json
Writing /workspaces/workspace/.vscode/extensions.json
Writing Dockerfile
Writing .dockerignore
vscode ➜ /workspaces/workspace $
```

Create a new Function

- Use Azure Function Core Tools to create function
  - func new --name flogo-hello-world-func --template "HTTP trigger"

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS 2    AZURE

vscode ➜/workspaces/workspace $ func new --name flogo-hello-world-func --template "HTTP trigger"
Select a number for template:HTTP trigger
Function name: [HttpTrigger] Writing /workspaces/workspace/flogo-hello-world-func/function.json
The function "flogo-hello-world-func" was created successfully from the "HTTP trigger" template.
vscode ➜/workspaces/workspace $
```

Add start script

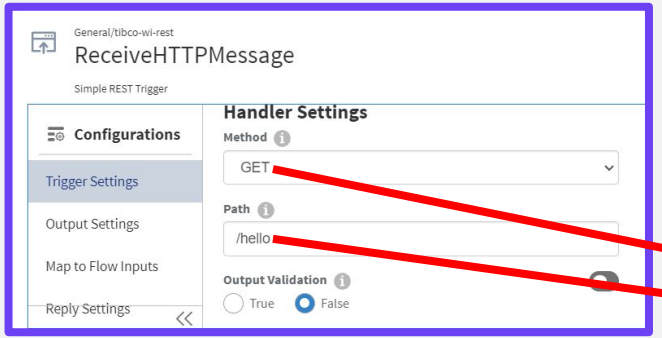
- Add a new file named ‘start.sh’ in the root of your workspace project folder.
- Set execute permissions
  - chmod +x start.sh

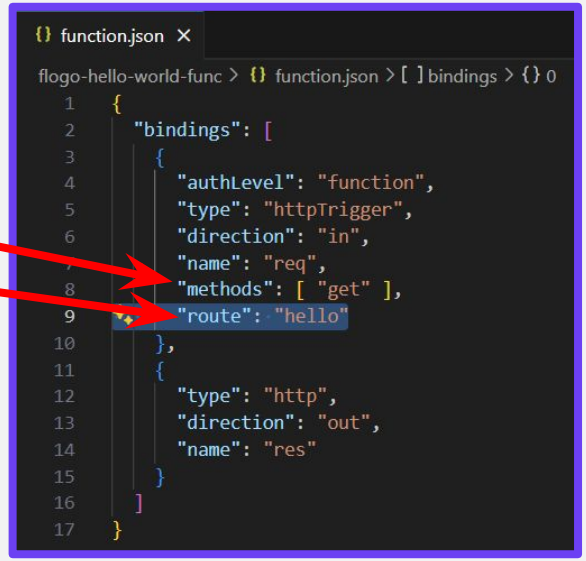
```
$ start.sh
```

```
$ start.sh
1  #!/usr/bin/env sh
2  echo "Starting function..."
3  PORT=${FUNCTIONS_CUSTOMHANDLER_PORT} ./flogo-hello-world
```

Modify function.json

- Found in folder created by ‘func new --name’
- Modify bindings





Modify Dockerfile

- Modify the generated Dockerfile to the following:

Dockerfile

Dockerfile > ...

```
FROM mcr.microsoft.com/azure-functions/dotnet:4
ENV AzureWebJobsScriptRoot=/home/site/wwwroot \
    AzureFunctionsJobHost__Logging__Console__IsEnabled=true

COPY .  "/home/site/wwwroot"
```

Modify host.json

- Modify
  - “defaultExecutablePath” : “start”
  - “enableForwardingHttpRequest” : true
- Add
  - “extensions” : { “http” : { “routePrefix” : ”” }}

host.json

host.json > ...

```
1  {
2    "version": "2.0",
3    "logging": {
4      "applicationInsights": {
5        "samplingSettings": {
6          "isEnabled": true,
7          "excludedTypes": "Request"
8        }
9      }
10   },
11   "extensionBundle": {
12     "id": "Microsoft.Azure.Functions.ExtensionBundle",
13     "version": "[4.*, 5.0.0)"
14   },
15   "customHandler": {
16     "description": {
17       "defaultExecutablePath": "",
18       "workingDirectory": "",
19       "arguments": []
20     }
21   }
22 }
```

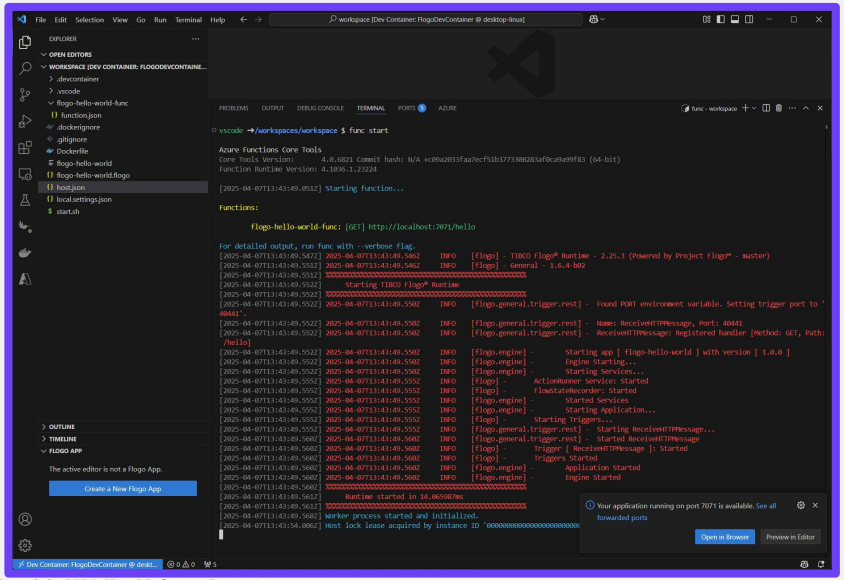
host.json

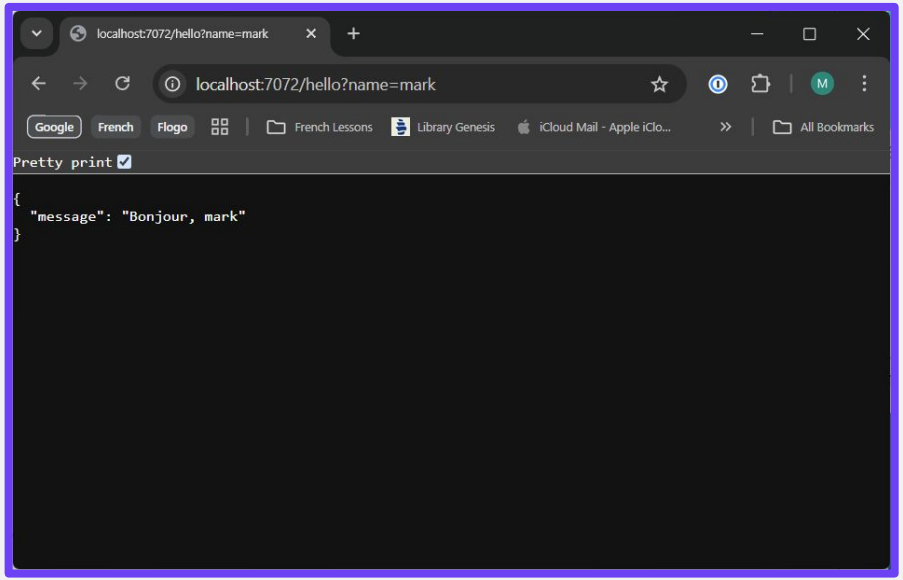
/workspaces/workspace/host.json

```
1  {
2    "version": "2.0",
3    "logging": {
4      "applicationInsights": {
5        "samplingSettings": {
6          "isEnabled": true,
7          "excludedTypes": "Request"
8        }
9      }
10   },
11   "extensionBundle": {
12     "id": "Microsoft.Azure.Functions.ExtensionBundle",
13     "version": "[4.*, 5.0.0)"
14   },
15   "customHandler": {
16     "description": {
17       "defaultExecutablePath": "start.sh",
18       "workingDirectory": "",
19       "arguments": []
20     },
21     "enableForwardingHttpRequest": true
22   },
23   "extensions": {
24     "http": {
25       "routePrefix": ""
26     }
27   }
28 }
```

Take it for a test drive

- Use Azure Function Core Tools to start a local runtime host and load the function project
  - [func start]





Build and Push Docker Image

1. Build docker image

   docker build -t flogo-hello-world:1.0

2. Tag for Azure Container Registry

   docker tag flogo-hello-world:1.0 presalesemeauk.azurecr.io/flogo-hello-world:1.0

3. Login to Azure

   az login

4. Login to Azure Container Registry

   az acr login --name presalesemeauk.azurecr.io

5. Push to Azure Container Registry

   docker push presalesemeauk.azurecr.io/flogo-hello-world:1.0

Create Azure Function App

- Use Azure CLI to create Azure Function App:

  az functionapp create --resource-group PresalesEMEAUK --os-type Linux --consumption-plan-location westeurope --runtime custom --functions-version 4 --name my-flogo-hello-world-app --storage-account emeaukfuncappstorage

Deploy Azure Function

- Zip application

  zip -r app.zip . -x “./.devcontainer/*”

- Deploy application

  az functionapp deployment source config-zip --resource-group presalesemeauk --name my-flogo-hello-world-app --src app.zip

File Edit Selection View Go Run Terminal Help

The extension 'Azure Resources' wants to access a Microsoft account

Select an account for 'Azure Resources' to use or Esc to cancel

Mark.Mussett@corp.cloud.com

mmussett@tibco.com

Sign in to another account

AZURE

RESOURCES Remote

Waiting for Azure sign-in...

WORKSPACE Local

Attached Storage Accounts

Attached Database Accounts (Postgres)

Local Project workspace

Attached Database Accounts

MongoDB Accounts

ACCOUNTS & TENANTS

Waiting for Azure sign-in...

HELP AND FEEDBACK

Dockerfile

```
1  FRO
2  ENV
3
4
5  COPY .  "/home/site/wwwroot"
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS 4 AZURE

bash - workspace

```
vscode ➜ /workspaces/workspace $ az functionapp create --resource-group PresalesEMEAUK --os-type Linux --consumption-plan-location westeurope --runtime cu
stom --functions-version 4 --name flogo-hello-world-app --storage-account emeaukfuncappstorage
Your Linux function app 'flogo-hello-world-app', that uses a consumption plan has been successfully created but is not active until content is published u
sing Azure Portal or the Functions Core Tools.
Application Insights "flogo-hello-world-app" was created for this Function App. You can visit https://portal.azure.com/#resource/subscriptions/342afe1e-b8
9b-4ab7-9ea8-3a341889f8b2/resourceGroups/PresalesEMEAUK/providers/microsoft.insights/components/flogo-hello-world-app/overview to view your Application In
sights component
{
  "availabilityState": "Normal",
  "clientAffinityEnabled": false,
  "clientCertEnabled": false,
  "clientCertExclusionPaths": null,
  "clientCertMode": "Required",
  "cloningInfo": null,
  "containerSize": 0,
  "customDomainVerificationId": "2B27447DDDC7E2EA0BF83D8AD854974651D1E9AE63CA146D6D176C114D2DE0CA",
  "dailyMemoryTimeQuota": 0,
  "daprConfig": null,
  "defaultHostName": "flogo-hello-world-app.azurewebsites.net",
  "enabled": true,
  "enabledHostNames": [
    "flogo-hello-world-app.azurewebsites.net",
    "flogo-hello-world-app.scm.azurewebsites.net"
  ],
  "endToEndEncryptionEnabled": false,
  "extendedLocation": null,
  "hostNameSslStates": [
    {
      "certificateResourceId": null,
      "hostType": "Standard",
      "ipBasedSslResult": null,
      "ipBasedSslState": "NotConfigured",
      "name": "flogo-hello-world-app.azurewebsites.net",
      "sslState": "Disabled",
      "thumbprint": null,
      "toUpdate": null,
      "toUpdateIpBasedSsl": null,
      "virtualIPv6": null,
      "virtualIp": null
```

Dev Container: FlogoDevContainer @ deskt... 0 0 4 Ln 5, Col 29 Spaces: 4 UTF-8 LF {} Dockerfile

File Edit Selection View Go Run Terminal Help

workspace [Dev Container: FlogoDevContainer @ desktop-linux]

AZURE

RESOURCES Remote

- azrpsemeafr-PresalesEMEAFrance
  - App Services
  - Azure AI Foundry
  - Azure Arc-enabled machines
  - Azure Cosmos DB
  - Azure Cosmos DB for MongoDB (vCore)
  - Container Apps
  - Durable Task Scheduler
  - Function App
    - adeccodemoaz
    - flogo-hello-world
    - flogo-h
      - Funct
      - Appli
      - Files
      - Logs
      - Deplo

Browse Website
Ask @azure
Deploy to Function App...
Configure Deployment Source...
Start
Stop
Restart
Delete Function App...
Start Streaming Logs
Stop Streaming Logs
Edit Tags...
View Properties
Open in Portal
Refresh

WORKSPACE Loc

- Attached
- Attached
- Local Proj
- Attached
- MongoDB

ACCOUNTS & TEN

- Mark.Mus
  - Cloud Software Group abd3ae6e-c344-4...
- mmussett@tibco.com
  - TIBCO Software Inc. cde6fa59-abb3-497...

HELP AND FEEDBACK

Dockerfile

```
FROM mcr.microsoft.com/azure-functions/dotnet:4
ENV AzureWebJobsScriptRoot=/home/site/wwwroot \
    AzureFunctionsJobHost__Logging__Console__IsEnabled=true

COPY . "/home/site/wwwroot"
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS 4 AZURE

bash - workspace

```
vscode ➜ /workspaces/workspace $ az functionapp create --resource-group PresalesEMEAUK --os-type Linux --consumption-plan-location westeurope --runtime cu
stom --functions-version 4 --name flogo-hello-world-app --storage-account emeaukfuncappstorage
Your Linux function app 'flogo-hello-world-app', that uses a consumption plan has been successfully created but is not active until content is published u
sing Azure Portal or the Functions Core Tools.
lication Insights "flogo-hello-world-app" was created for this Function App. You can visit https://portal.azure.com/#resource/subscriptions/342afe1e-b8
4ab7-9ea8-3a341889f8b2/resourceGroups/PresalesEMEAUK/providers/microsoft.insights/components/flogo-hello-world-app/overview to view your Application In
hts component

availabilityState": "Normal",
clientAffinityEnabled": false,
clientCertEnabled": false,
clientCertExclusionPaths": null,
clientCertMode": "Required",
cloningInfo": null,
containerSize": 0,
customDomainVerificationId": "2B27447DDDC7E2EA0BF83D8AD854974651D1E9AE63CA146D6D176C114D2DE0CA",
dailyMemoryTimeQuota": 0,
daprConfig": null,
defaultHostName": "flogo-hello-world-app.azurewebsites.net",
enabled": true,
enabledHostNames": [
  "flogo-hello-world-app.azurewebsites.net",
  "flogo-hello-world-app.scm.azurewebsites.net"
,
endToEndEncryptionEnabled": false,
extendedLocation": null,
hostNameSslStates": [
  {
    "certificateResourceId": null,
    "hostType": "Standard",
    "ipBasedSslResult": null,
    "ipBasedSslState": "NotConfigured",
    "name": "flogo-hello-world-app.azurewebsites.net",
    "sslState": "Disabled",
    "thumbprint": null,
    "toUpdate": null,
    "toUpdateIpBasedSsl": null,
    "virtualIPv6": null,
    "virtualIp": null
```

Dev Container: FlogoDevContainer @ deskt... ⊗ 0 ⚠ 0 4

Ln 5, Col 29 Spaces: 4 UTF-8 LF {} Dockerfile