vsCode Demo

Start LTM-1 VM

Start LTM-2 VM

Client w VS Code

* F5 FAST extension
* YAML extension

1. **Einführung**

**Visual Studio Code** is a code editor redefined and optimized for building and debugging modern web and cloud applications.

Zeige GIT https://github.com/de1chk1nd/VScodeDemo

Zeige VS Code Oberfläche

Zeige github Integration und Ordnerstruktur

<F1> – git: clone

Zeige extensions

* [https://marketplace.visualstudio.com/items?itemName=DumpySquare.vscode-f5-fast](https://mcas-proxyweb.mcas.ms/certificate-checker?login=false&originalUrl=https%3A%2F%2Fmarketplace.visualstudio.com.mcas.ms%2Fitems%3FitemName%3DDumpySquare.vscode-f5-fast)

**Alt:** Tool wurde weiterentwickelt. Früher war es für das Device Management (bigip hinzufügen, PWD Caching) gedacht, ATC Discovery, etc.

**Neu:**

* + ATC Management (Install RPM, pull from git)
    - DO
    - AS3 Call
    - FAST Templates
    - TS
  + YAML <> JSON Converter
  + base64 de- & encoding
  + (better) support from pushing from editor (right-click)
  + Open API “Support” (API Calls)

1. **Basis Connection**

* F5 plugin im Tool zeigen
  + Information im unteren Teil (blau)
* Mit einer F5 Verbinden (LTM-1) [anklicken menu; blau: F5-FAST > Connect]
  + (rightclick device) Show Device Info
  + (rightclick login provider)

1. **AS3**

***Zeigen***

Verbundene F5 im unteren Bereich zeigen (bigip, Version, etc)

AS3 Menu links zeigen (TASKS & TENANTS)

<right-click> “hostname” im blauen Balken (Device Info – viele Wege führen nach Rom 😉 )

* Im json file <ctrl-a>, [as3] eingeben, template holen
* SchemaFile ist eingebunden
  + ***Hover Over Content*** Menupunkte -> Erklärung
  + ***Change Content*** -> error message
  + ***Add Content*** einfügen -> autocomplete

***Anlegen & Testen***

Add Virtual Server : 10.1.10.150

Add Node : 212.227.165.10

* CTRL-A – right-click – „POST AS3”
* Browser: 10.1.10.150
* Zeigen TMUI

***Verifizieren & löschen***

[linkes menu]

AS3, Tenants, [Sample\_01]

* + AS3 – Call AS3 Call
  + <right-click>, expended AS3 Call mit default Werten
  + <right-click>, delete tenant

1. **FAST Templates**

***Zeigen***

Verbundene F5 im unteren Bereich zeigen (bigip, Version, etc)

FAST Menu links zeigen (TASKS & TEMPLATES, TEMPLATE SETS)

***Arbeiten mit Template Sets***

Auf git wechseln und in den „goodTemplate“ Folder wechseln

* Zeigen des Inhalts und der Readme Datei

<right click folder> and „POST Template SET“

Auf das F5 plugin gehen > TEMPLATE SETs

Auf das F5 plugin gehen > TEMPLATE

***Templates sind hochgeladen und man kann mit denen arbeiten.***

***Deploy & Test Services***

GIT – README

* App1 - Markieren <right-click> Deploy Fast App

Add Virtual Server : 10.1.10.151

Add Node : 212.227.165.10

* App2 - Markieren <right-click> Deploy Fast App

Add Virtual Server : 10.1.10.152

Add Node : 212.227.165.10

Löschen der Apps: <right click [deployed App] > [delete FAST App]

***Rendering Output***

Select App Template in GIT (app3, app4) > [***F1***] > F5-Fast: Render HTML Preview

***DELETE TEMPLATE SET***

1. **RPM Management**

***Connect to F5-LTM-2***

Install packages: DO, AS3, FAST, TS

***F1: F5: Install Update RPM***

*Comm-Flow: VScode -> git -> download if not cached -> upload to f5*

Un-Install packages: DO, AS3, FAST, TS

***F1: F5: Un-Install RPM***

*Show iApp LX package installation from git*

*Install DO*

GET Device Info: DO, example

1. **OPEN API stuff …**

Git: “***makeHTTPsTesting.md***”

Make anything “API”

Disconnect

* Cool Feature – “context aware” -> missing host

> right click: Make HTTP request

***HTTP, YAML, JSON***

**### simple url as string**

```

/mgmt/tm/sys/clock

```

**### simple url in yaml format**

```

url: /mgmt/tm/sys/clock

```

**### simple url in json format**

```

{

  "url": "/mgmt/tm/sys/clock"

}

```

**### url post in yaml**

```

url: /mgmt/shared/authn/login

method: POST

body:

    username: admin

    password: f5twister!

```

**### url post in yaml**

```

url: /mgmt/tm/sys/config

method: POST

body:

    command: save

```

***External Calls***

**### Simple get**

```

https://api.chucknorris.io/jokes/random

```

**### simple get in yaml**

```

url: https://api.chucknorris.io/jokes/random

```

**### simple get in yaml - shows broken destination**

```

url: https://broken.extra.io/whah

```

**### simple POST in yaml**

```

url: 'https://postman-echo.com/post'

method: POST

data:

  hi: yo

```

**### simple POST in json**

```

{

    "url": "https://postman-echo.com/post",

    "method": "POST",

    "data": {

        "hi": "yo"

    }

}

```

**### raw get to external F5 using basic auth - DEV**

```

{

  "url": "https://192.168.200.131/mgmt/tm/sys/ntp",

  "auth": {

      "username": "admin",

      "password": "yayPassword!",

      "sendImmediately": "true"

  }

}

```

***Convert JSON <-> YAML & base64 de- encode***

Show extension: YAML

Settings: YAML schemas -> f5

Show YAML declaration -> show schema validation

* Hover over
* Add wrong text

Show NAP syntax

* Convert to YAML

Show “cert import”

1. **Troubleshooting**

Falschen Call erstellen > abschiessen

ID merken

ID suchen