

# Opdracht 4

---

## Cloud Technology & Security

Qing Scholten

---

### Opdracht 1: Http trigger function

1.

```
const { app } = require('@azure/functions');

app.http('Function2', {
  methods: ['GET', 'POST'],
  authLevel: 'anonymous',
  handler: async (request, context) => {
    context.log(`Http function processed request for url "${request.url}"`);

    const parameter = request.query.get('parameter');
    if (parameter === null) {
      context.log("Error! Parameter ontbreekt!");
      return {
        status: 500,
        body: "Error! Je hebt parameter niet meegegeven!"
      }
    }
    return { body: `Hello, je parameter is ${parameter}!` };
  }
});
```

2. Het await keyword is aanwezig omdat het proces asynchroon de gegevens van het HTTP-verzoek inleest. Het voert de functie pas uit als de benodigde data binnen is. Door de asynchrone functionaliteit blijft de server efficiënt werken zonder dat het geblokkeerd wordt doordat het wacht op data.

3.

- az login
- az group create --name Practicum4 --location westeurope
- az storage account create --name practicum4 --location westeurope --resource-group Practicum4 --sku Standard\_LRS --allow-blob-public-access false
- az functionapp create --resource-group Practicum4 --consumption-plan-location westeurope --runtime node --runtime-version 20 --functions-version 4 --name practicum4 --storage-account practicum4
- func azure functionapp publish practicum4 De logmeldingen verschijnen in de cli nadat de command "func azure functionapp logstream practicum4" ingevoerd is.

4. Iedereen op het internet met de URL van de site waar de functie aan vast zit.

---

## Opdracht 2: REST api met azure function

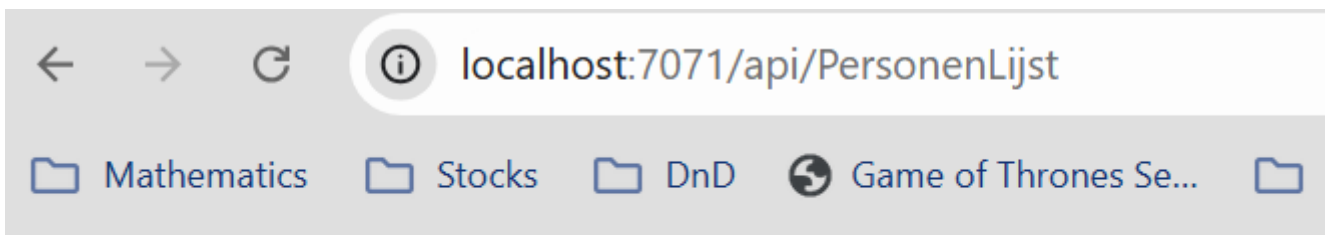
1.

```
const { app } = require('@azure/functions');

app.http('PersonenLijst', {
  methods: ['GET'],
  authLevel: 'anonymous',
  handler: async (request, context) => {
    context.log(`Http function processed request for url "${request.url}"`);

    const jsonlist = {Personen:
      [{Naam: "Kwak"},
      {Naam: "Kwik"}]};
    const textlist = JSON.stringify(jsonlist);

    return { body: `Hello, ${textlist}!` };
  }
});
```



**Hello, {"Personen":[{"Naam":"Kwak"}, {"Naam":"Kwik"}]}!**

2.

```
const { app } = require('@azure/functions');

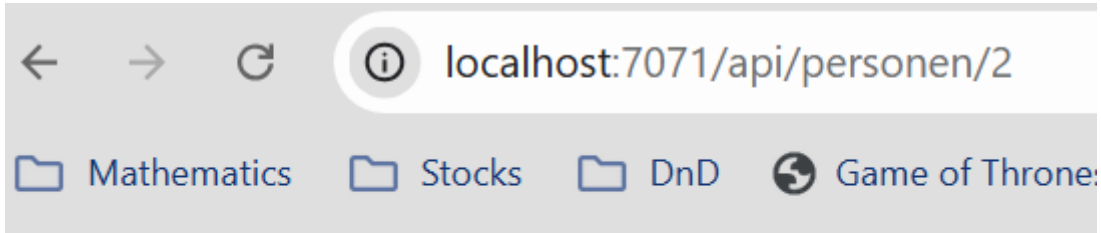
app.http('PersonenLijst', {
  methods: ['GET'],
  authLevel: 'anonymous',
  route: 'personen/{id:int?}',
  handler: async (request, context) => {
    context.log(`Http function processed request for url "${request.url}"`);

    var id = request.params.id - 1;
    const jsonlist = {Personen:
      [{Naam: "Kwak"},
      {Naam: "Kwik"},
      {Naam: "Kwek"}]};
    const persoon = jsonlist.Personen[id];
    if (isNaN(id)) {
      var textlist = JSON.stringify(jsonlist);
    }
  }
});
```

```

    }
    else if (id > jsonlist.Personen.length-1) {
        var textlist = "Deze persoon bestaat niet!";
    }
    else {
        var textlist = JSON.stringify(persoon);
    }
    return { body: `${textlist}` };
}
});

```



```

{"Naam": "Kwik"}

```

3.

```

const { app } = require('@azure/functions');
const jsonlist = {Personen:
  [{Naam: "Kwak"},
  {Naam: "Kwik"},
  {Naam: "Kwek"}]};

app.http('PersonenLijst', {
  methods: ['GET'],
  authLevel: 'anonymous',
  route: 'personen/{id:int?}',
  handler: async (request, context) => {
    context.log(`Http function processed request for url "${request.url}"`);

    var id = request.params.id - 1;
    var persoon = jsonlist.Personen[id];
    if (isNaN(id)) {
      context.log(`Verzoek tot opvragen gehele personenlijst.`)
      var textlist = JSON.stringify(jsonlist);
    }
    else if (id > jsonlist.Personen.length || id < 0) {
      context.log(`Verzoek tot opvragen van niet bestaand persoon!`)
      return {
        status: 404,
        body: "404! Deze persoon bestaat niet!"
      }
    }
    else {
      context.log(`Verzoek tot opvragen van persoon ${id}.`)

```

```
        var textlist = JSON.stringify(persoon);
    }
    return { body: `${textlist}` };
}
});
```

GET http://localhost:7071/api/personen/4 Send

Params Authorization Headers (7) Body Scripts Settings Cookies

Query Params

Key	Value	Description	Bulk Edit
Key	Value	Description	

Body Cookies Headers (4) Test Results 404 Not Found 17 ms 179 B Save Response

Pretty Raw Preview Visualize Text

1 404! Deze persoon bestaat niet!

4.

```
app.http('PersonenToevoegen', {
  methods: ['POST'],
  authLevel: 'anonymous',
  route: 'personen',
  handler: async (request, context) => {
    context.log(`HTTP function processed request for url "${request.url}"`);
    try {
      var bod = await request.text();
      var nieuwpersoon = JSON.parse(bod);
      if (!nieuwpersoon.hasOwnProperty("Naam")) {
        context.log("Poging tot toevoeging persoon zonder naam.");
        return {
          status: 404
        }
      }
    }
    else {
      jsonlist.Personen.push(nieuwpersoon);
      context.log(`Persoon toegevoegd:
${JSON.stringify(nieuwpersoon)}`);
      return {
        status: 200,
        body: `Persoon succesvol toegevoegd:
${JSON.stringify(nieuwpersoon)}`
      }
    }
  }
})
```

```
        catch (e) {
            context.log(`Fout bij het verwerken van het toevoegen van een
persoon`, e);
            return {
                status: 500,
                body: 'Interne serverfout bij toevoegen van persoon.'
            }
        }
    }
});
```

POST http://localhost:7071/a • GET http://localhost:7071/ap • +

q.scholten.qs@gmail.com

Word Klan

HTTP http://localhost:7071/api/personen

Save Share

POST http://localhost:7071/api/personen

Send

Params Authorization Headers (9) Body Scripts Settings

Body

none form-data x-www-form-urlencoded raw binary GraphQL JSON

JSON

1 { "Naam": "Donald" }

Body Cookies Headers (4) Test Results

200 OK • 12 ms • 188 B • Save Response

Pretty Raw Preview Visualize Text

1 Persoon succesvol toegevoegd: { "Naam": "Donald" }

POST http://localhost:7071/a • GET http://localhost:7071/ap • +

No environment

HTTP http://localhost:7071/api/personen

Save Share

GET http://localhost:7071/api/personen

Send

Params Authorization Headers (7) Body Scripts Settings

Body

Query Params

Key	Value	Description
Key	Value	Description

Body Cookies Headers (4) Test Results

200 OK • 11 ms • 221 B • Save Response

Pretty Raw Preview Visualize Text

1 { "Personen": [ { "Naam": "Kwak" }, { "Naam": "Kwik" }, { "Naam": "Kwek" }, { "Naam": "Donald" } ] }

5.

```
app.http('PersonenAanpassen', {
  methods: ['PUT'],
  authLevel: 'function',
```

```
route: 'personen/{id:int?}',
handler: async (request, context) => {
  context.log(`HTTP function processed request for url "${request.url}"`);
  try {
    var id = request.params.id - 1;
    if (isNaN(id) || id > jsonlist.Personen.length || id < 0) {
      context.log(`Poging tot wijzigen van niet bestaand persoon.`)
      return {
        status: 400,
        body: 'Persoon met dit id bestaat niet.'
      }
    }
    var bod = await request.json();
    var naam = bod.Naam;
    if (naam == null) {
      context.log("Poging tot verandering persoon zonder naam.")
      return {
        status: 400,
        body: 'Geen naam ingegeven.'
      }
    }
    else {
      var nieuwpersoon = {Naam: naam};
      jsonlist.Personen[id] = nieuwpersoon;
      context.log(`Persoon aangepast: ${JSON.stringify(nieuwpersoon)}`);
      return {
        status: 200
      }
    }
  }
  catch (e) {
    context.log(`Fout bij het verwerken van het veranderen van een
persoon`, e);
    return {
      status: 500,
      body: 'Interne serverfout bij veranderen van persoon.'
    }
  }
});
```

PUT http://localhost:7071/ap GET http://localhost:7071/ap +

q.scholten.qs@gmail.com

Wind Klan

Save Share

http://localhost:7071/api/personen/1

PUT http://localhost:7071/api/personen/1 Send

Params Authorization Headers (9) Body Scripts Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

1 { "Naam": "Donald" }

Body Cookies Headers (3) Test Results

200 OK 11 ms 92 B Save Response

Pretty Raw Preview Visualize Text

1

PUT http://localhost:7071/ap GET http://localhost:7071/ +

q.scholten.qs@gmail.com

Wind Klan

Save Share

http://localhost:7071/api/personen

GET http://localhost:7071/api/personen Send

Params Authorization Headers (7) Body Scripts Settings

Query Params

Key	Value	Description
Key	Value	Description

Body Cookies Headers (4) Test Results

200 OK 9 ms 205 B Save Response

Pretty Raw Preview Visualize Text

1 { "Personen": [ { "Naam": "Donald" }, { "Naam": "Kwik" }, { "Naam": "Kwek" } ] }

6.

```
app.http('PersonenVerwijderen', {
  methods: ['DELETE'],
  authLevel: 'anonymous',
  route: 'personen/{id:int?}',
  handler: async (request, context) => {
    context.log(`HTTP function processed request for url "${request.url}"`);
    try {
      var id = request.params.id - 1;
      if (isNaN(id) || id > jsonlist.Personen.length || id < 0) {
        context.log(`Poging tot verwijderen van niet bestaand persoon.`)
        return {
          status: 400,
          body: 'Persoon met dit id bestaat niet.'
        }
      }
    }
  }
})
```

```
        }
    }
    else {
        var verwijderpersoon = jsonlist.Personen[id];
        jsonlist.Personen.splice(id,1);
        context.log(`Persoon verwijderd:
${JSON.stringify(verwijderpersoon)}}`);
        return {
            status: 200
        }
    }
}
catch (e) {
    context.log(`Fout bij het verwerken van het verwijderen van een
persoon`, e);
    return {
        status: 500,
        body: 'Interne serverfout bij verwijderen van persoon.'
    }
}
}
});
```



DEL http://localhost:7071/ap

GET http://localhost:7071/ap

+

No environment

http://localhost:7071/api/personen/2

Save

Share

DELETE

http://localhost:7071/api/personen/2

Send

Params

Authorization

Headers (7)

Body

Scripts

Settings

Cookies

Query Params

Key	Value	Description	Bulk Edit
Key	Value	Description	

Body

Cookies

Headers (3)

Test Results

200 OK

15 ms

92 B

Save Response

Pretty

Raw

Preview

Visualize

Text

1

DEL http://localhost:7071/ap

GET http://localhost:7071/ap

+

No environment

http://localhost:7071/api/personen

Save

Share

GET

http://localhost:7071/api/personen

Send

Params

Authorization

Headers (7)

Body

Scripts

Settings

Cookies

Query Params

Key	Value	Description	Bulk Edit
Key	Value	Description	

Body

Cookies

Headers (4)

Test Results

200 OK

14 ms

187 B

Save Response

Pretty

Raw

Preview

Visualize

Text

1

{ "Personen": [{ "Naam": "Kwak" }, { "Naam": "Kwek" }] }

7.

POSThttps://practicum4.azure...GEThttp://localhost:7071/ap...

https://practicum4.azurewebsites.net/api/personen

POSThttps://practicum4.azurewebsites.net/api/personen

ParamsAuthorizationHeaders (9)BodyScriptsSettings

noneform-datax-www-form-urlencodedrawbinaryGraphQLJSON

1{"Naam": "Donald"}

BodyCookiesHeaders (6)Test Results

200 OK472 ms400 B

PrettyRawPreviewVisualizeText

1Persoon succesvol toegevoegd: {"Naam": "Donald"}

POSThttps://practicum4.azure...GEThttps://practicum4.azure...

https://practicum4.azurewebsites.net/api/personen

GEThttps://practicum4.azurewebsites.net/api/personen

ParamsAuthorizationHeaders (7)BodyScriptsSettings

Query Params

Key	Value	Description
Key	Value	Description

BodyCookiesHeaders (6)Test Results

200 OK714 ms409 B

PrettyRawPreviewVisualizeText

1{"Personen": [{"Naam": "Kwak"}, {"Naam": "Kwik"}, {"Naam": "Kwek"}, {"Naam": "Donald"}]}

PUThttps://practicum4.azure...GEThttps://practicum4.azure...

https://practicum4.azurewebsites.net/api/personen/1

PUThttps://practicum4.azurewebsites.net/api/personen/1

ParamsAuthorizationHeaders (9)BodyScriptsSettings

noneform-datax-www-form-urlencodedrawbinaryGraphQLJSON

1{"Naam": "Dagobert"}

BodyCookiesHeaders (3)Test Results

200 OK25 ms143 B

PrettyRawPreviewVisualizeText

1

PUT https://practicum4.azure

GET https://practicum4.azure

+

https://practicum4.azurewebsites.net/api/personen

Save

Share

GET

https://practicum4.azurewebsites.net/api/personen

Send

Params

Authorization

Headers (7)

Body

Scripts

Settings

Cookies

Query Params

Key	Value	Description	Bulk Edit
Key	Value	Description	

Body

Cookies

Headers (6)

Test Results

200 OK • 571 ms • 414 B • Save Response

Pretty

Raw

Preview

Visualize

Text

1 { "Personen": [{"Naam": "Dagobert"}, {"Naam": "Kwik"}, {"Naam": "Kwek"}, {"Naam": "Donald"}]}

DEL https://practicum4.azure

GET https://practicum4.azure

+

https://practicum4.azurewebsites.net/api/personen/1

Save

Share

DELETE

https://practicum4.azurewebsites.net/api/personen/1

Send

Params

Authorization

Headers (7)

Body

Scripts

Settings

Cookies

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

Beautify

1

Body

Cookies

Headers (3)

Test Results

200 OK • 55 ms • 143 B • Save Response

Pretty

Raw

Preview

Visualize

Text

1

DEL https://practicum4.azure

GET https://practicum4.azure

+

https://practicum4.azurewebsites.net/api/personen

Save

Share

GET

https://practicum4.azurewebsites.net/api/personen

Send

Params

Authorization

Headers (7)

Body

Scripts

Settings

Cookies

Query Params

Key	Value	Description	Bulk Edit
Key	Value	Description	

Body

Cookies

Headers (6)

Test Results

200 OK • 22 ms • 406 B • Save Response

Pretty

Raw

Preview

Visualize

Text

1 { "Personen": [{"Naam": "Kwik"}, {"Naam": "Kwek"}, {"Naam": "Donald"}]}

- 1. Door de authorization level te veranderen van 'anonymous' naar 'function' of 'admin' is er een specifieke key nodig om de API te kunnen gebruiken.
- 2. Er is een parameter "code" toegevoegd in de link die gebruikt wordt voor het GET request.

DEL https://practicum4.azure

GET https://practicum4.azure

+

No environment

https://practicum4.azurewebsites.net/api/personen?code=onhtlcvis6qUu4526Lqb49wJtc2xo1uxybesCB7ZbldwAzFugiFeag==

Save

Share

GET

https://practicum4.azurewebsites.net/api/personen?code=onhtlcvis6qUu4526Lqb49wJtc2xo1uxybesCB7ZbldwAzFugiFeag==

Send

Params

Authorization

Headers (7)

Body

Scripts

Settings

Cookies

Query Params

<input checked="" type="checkbox"/>	Key	Value	Description	Bulk Edit
<input checked="" type="checkbox"/>	code	onhtlcvis6qUu4526Lqb49wJtc2xo1uxybesCB7ZbldwAzFugiFeag==		
	Key	Value	Description	

Body

Cookies

Headers (6)

Test Results

200 OK

313 ms

400 B

Save Response

Pretty

Raw

Preview

Visualize

Text

1

{ "Personen": [ { "Naam": "Kwak" }, { "Naam": "Kwik" }, { "Naam": "Kwek" } ] }