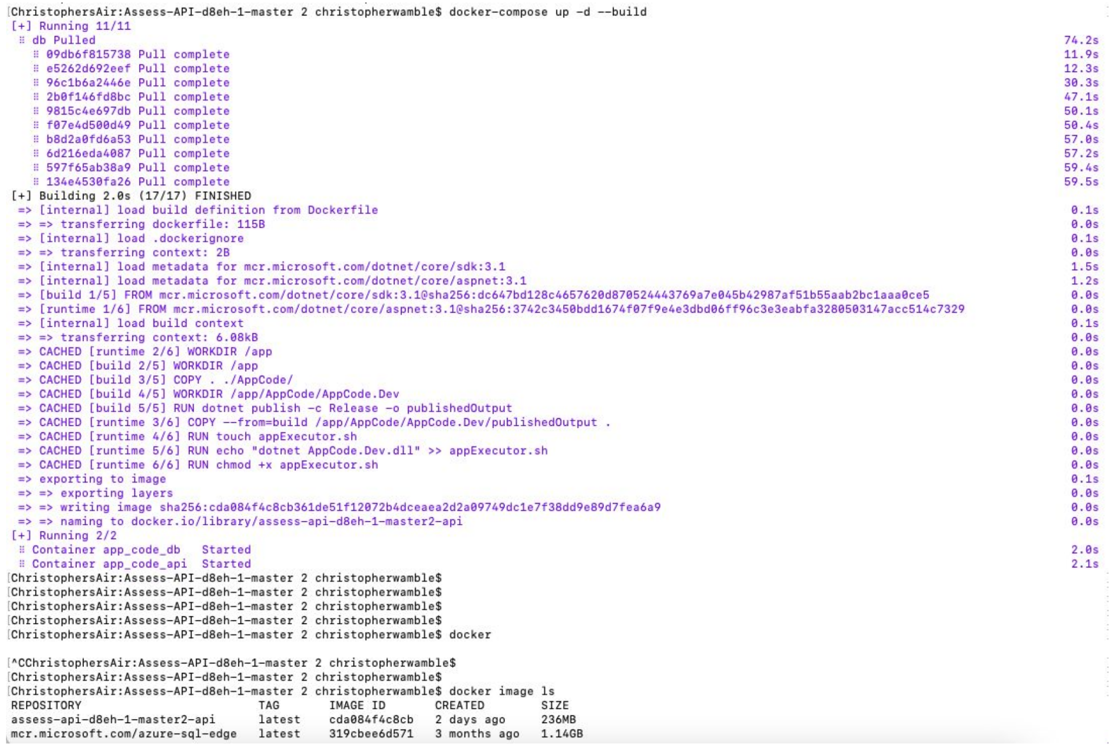
The Explanation

What I tried

Installing docker

Navigate to project folder and run \`docker-compose up -d --build\`

Confirmed build completed successfully and that image and container are up and running

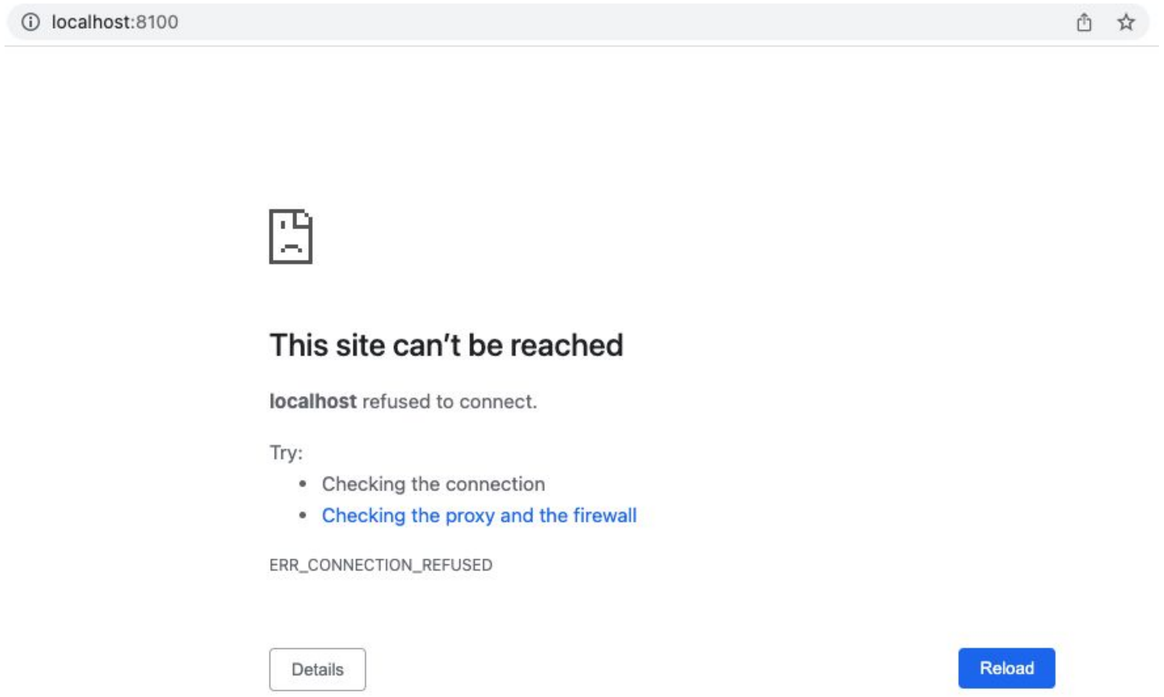


```
ChristophersAir:Assess-API-d8eh-1-master 2 christopherwamble$ docker-compose up -d --build
[+] Running 11/11
 ⠿ db Pulled                                                                                                                       74.2s
   ⠿ 09db6f815738 Pull complete                                                                                                    11.9s
   ⠿ e5262d692eef Pull complete                                                                                                    12.3s
   ⠿ 96c1b6a2446e Pull complete                                                                                                    30.3s
   ⠿ 2b0f146fd8bc Pull complete                                                                                                    47.1s
   ⠿ 9815c4e697db Pull complete                                                                                                    50.1s
   ⠿ f07e4d500d49 Pull complete                                                                                                    50.4s
   ⠿ b8d2a0fd6a53 Pull complete                                                                                                    57.0s
   ⠿ 6d216eda4087 Pull complete                                                                                                    57.2s
   ⠿ 597f65ab38a9 Pull complete                                                                                                    59.4s
   ⠿ 134e4530fa26 Pull complete                                                                                                    59.5s
[+] Building 2.0s (17/17) FINISHED
 => [internal] load build definition from Dockerfile                                                                                0.1s
 => => transferring dockerfile: 115B                                                                                                0.0s
 => [internal] load .dockerignore                                                                                                   0.1s
 => => transferring context: 2B                                                                                                     0.0s
 => [internal] load metadata for mcr.microsoft.com/dotnet/core/sdk:3.1                                                              1.5s
 => [internal] load metadata for mcr.microsoft.com/dotnet/core/aspnet:3.1                                                           1.2s
 => [build 1/5] FROM mcr.microsoft.com/dotnet/core/sdk:3.1@sha256:dc647bd128c4657620d870524443769a7e045b42987af51b55aab2bc1aaa0ce5   0.0s
 => [runtime 1/6] FROM mcr.microsoft.com/dotnet/core/aspnet:3.1@sha256:3742c3450bdd1674f07f9e4e3dbd06ff96c3e3eabfa3280503147acc514c7329 0.0s
 => [internal] load build context                                                                                                   0.1s
 => => transferring context: 6.08kB                                                                                                 0.0s
 => CACHED [runtime 2/6] WORKDIR /app                                                                                               0.0s
 => CACHED [build 2/5] WORKDIR /app                                                                                                 0.0s
 => CACHED [build 3/5] COPY . ./AppCode/                                                                                            0.0s
 => CACHED [build 4/5] WORKDIR /app/AppCode/AppCode.Dev                                                                             0.0s
 => CACHED [build 5/5] RUN dotnet publish -c Release -o publishedOutput                                                             0.0s
 => CACHED [runtime 3/6] COPY --from=build /app/AppCode/AppCode.Dev/publishedOutput .                                               0.0s
 => CACHED [runtime 4/6] RUN touch appExecutor.sh                                                                                   0.0s
 => CACHED [runtime 5/6] RUN echo "dotnet AppCode.Dev.dll" >> appExecutor.sh                                                        0.0s
 => CACHED [runtime 6/6] RUN chmod +x appExecutor.sh                                                                                0.0s
 => exporting to image                                                                                                              0.1s
 => => exporting layers                                                                                                             0.0s
 => => writing image sha256:cda084f4c8cb361de51f12072b4dceaea2d2a09749dc1e7f38dd9e89d7fea6a9                                        0.0s
 => => naming to docker.io/library/assess-api-d8eh-1-master2-api                                                                    0.0s
[+] Running 2/2
 ⠿ Container app_code_db   Started                                                                                                  2.0s
 ⠿ Container app_code_api  Started                                                                                                  2.1s
ChristophersAir:Assess-API-d8eh-1-master 2 christopherwamble$
ChristophersAir:Assess-API-d8eh-1-master 2 christopherwamble$
ChristophersAir:Assess-API-d8eh-1-master 2 christopherwamble$
ChristophersAir:Assess-API-d8eh-1-master 2 christopherwamble$
ChristophersAir:Assess-API-d8eh-1-master 2 christopherwamble$ docker

^CChristophersAir:Assess-API-d8eh-1-master 2 christopherwamble$
ChristophersAir:Assess-API-d8eh-1-master 2 christopherwamble$
ChristophersAir:Assess-API-d8eh-1-master 2 christopherwamble$ docker image ls
REPOSITORY                         TAG       IMAGE ID       CREATED        SIZE
assess-api-d8eh-1-master2-api      latest    cda084f4c8cb   2 days ago     236MB
mcr.microsoft.com/azure-sql-edge   latest    319cbee6d571   3 months ago   1.14GB
```

What I tried

Launch http://localhost:8100

No such page, unable to view Swagger.UI documentation



localhost:8100

This site can't be reached

localhost refused to connect.

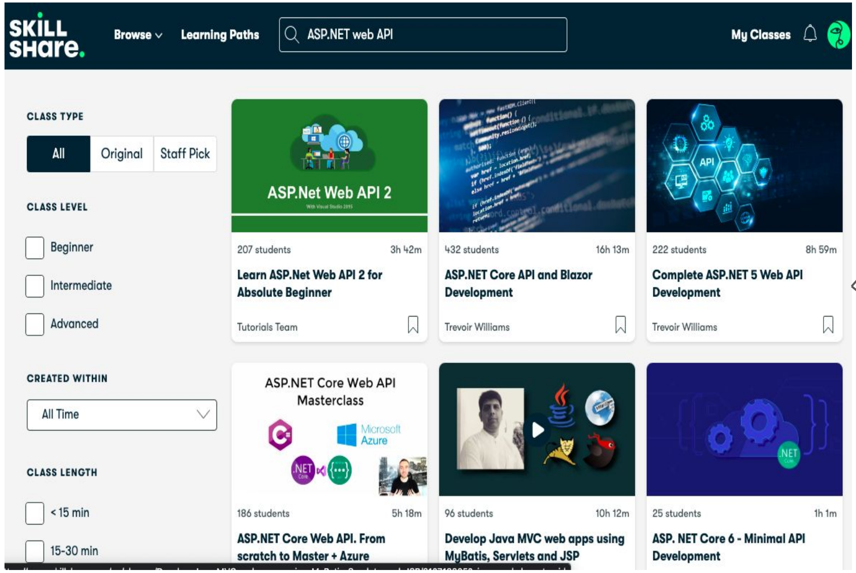
Try:

- Checking the connection
- Checking the proxy and the firewall

ERR_CONNECTION_REFUSED

What I tried

Since I’m not experienced in coding ASP.NET or coding APIs, I reviewed ASP.NET tutorials from my personal skillshare account to get a better understanding.



SKILL SHARE. Browse Learning Paths ASP.NET web API My Classes

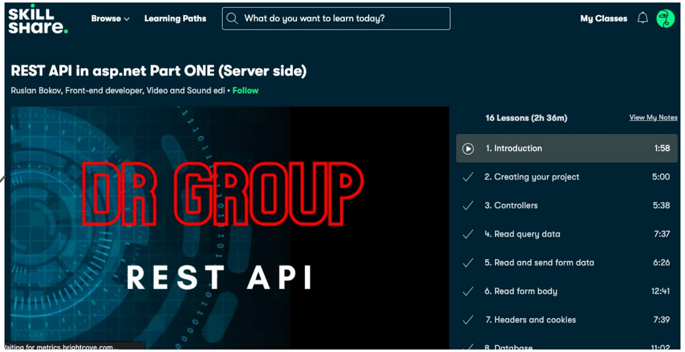
CLASS TYPE: All | Original | Staff Pick

CLASS LEVEL: Beginner, Intermediate, Advanced

CREATED WITHIN: All Time

CLASS LENGTH: < 15 min, 15-30 min

Students	Length	Title	Teacher
207 students	3h 42m	Learn ASP.Net Web API 2 for Absolute Beginner	Tutorials Team
432 students	16h 13m	ASP.NET Core API and Blazor Development	Trevoir Williams
222 students	8h 59m	Complete ASP.NET 5 Web API Development	Trevoir Williams
186 students	5h 18m	ASP.NET Core Web API. From scratch to Master + Azure	
96 students	10h 12m	Develop Java MVC web apps using MyBatis, Servlets and JSP	
25 students	1h 1m	ASP. NET Core 6 - Minimal API Development	



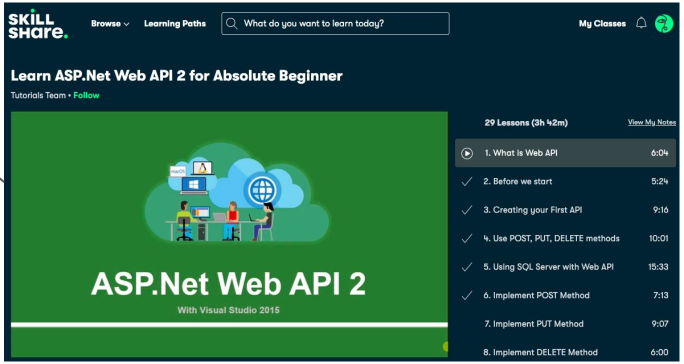
SKILL SHARE. Browse Learning Paths What do you want to learn today? My Classes

REST API in asp.net Part ONE (Server side)

Ruslan Bokov, Front-end developer, Video and Sound edi • Follow

16 Lessons (2h 36m) View My Notes

Lesson	Length
1. Introduction	1:58
2. Creating your project	5:00
3. Controllers	5:38
4. Read query data	7:37
5. Read and send form data	6:26
6. Read form body	12:41
7. Headers and cookies	7:39
8. Database	11:02



SKILL SHARE. Browse Learning Paths What do you want to learn today? My Classes

Learn ASP.Net Web API 2 for Absolute Beginner

Tutorials Team • Follow

29 Lessons (3h 42m) View My Notes

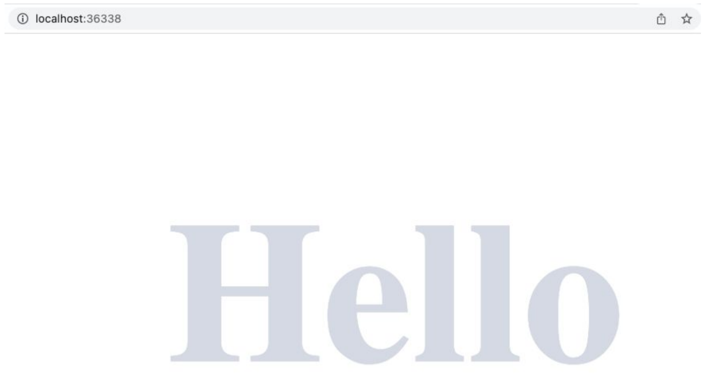
Lesson	Length
1. What is Web API	6:04
2. Before we start	5:24
3. Creating your First API	9:16
4. Use POST, PUT, DELETE methods	10:01
5. Using SQL Server with Web API	15:33
6. Implement POST Method	7:13
7. Implement PUT Method	9:07
8. Implement DELETE Method	6:00

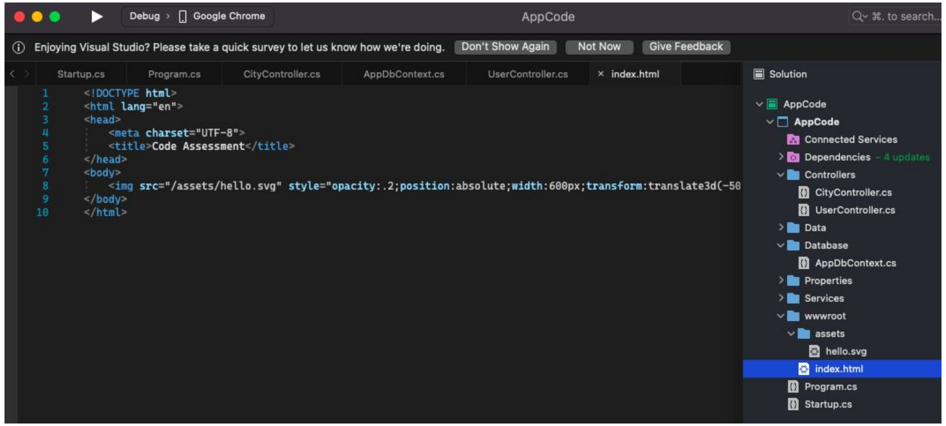
What Worked

Downloaded Visual Studios for Mac (Intel Chip), version 2022

Opened the project, compiled it, ran it

Able to get generated webpage in the project successfully





Enjoying Visual Studio? Please take a quick survey to let us know how we're doing.

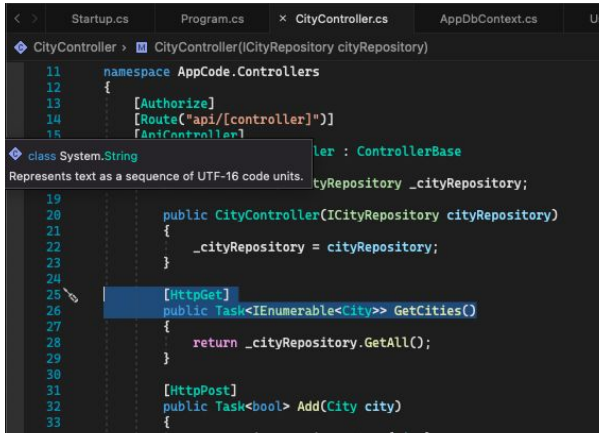
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Code Assessment</title>
</head>
<body>
    <img src="/assets/hello.svg" style="opacity:.2;position:absolute;width:600px;transform:translate3d(-50
</body>
</html>
```

But also, what didn’t work

Attempted to append the API call for GetCities

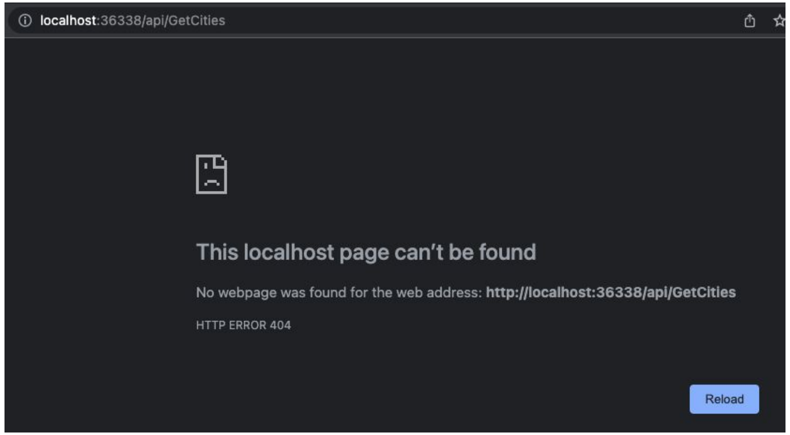
Expected to see a list of existing cities, or a response that there are no cities.

Response was localhost page can’t be found


Startup.cs
Program.cs
CityController.cs
AppDbContext.cs
CityController › CityController(ICityRepository cityRepository)

```
namespace AppCode.Controllers
{
    [Authorize]
    [Route("api/[controller]")]
    [ApiController]
                                    ler : ControllerBase

                                    tyRepository _cityRepository;

        public CityController(ICityRepository cityRepository)
        {
            _cityRepository = cityRepository;
        }

        [HttpGet]
        public Task<IEnumerable<City>> GetCities()
        {
            return _cityRepository.GetAll();
        }

        [HttpPost]
        public Task<bool> Add(City city)
        {
```

class System.String
Represents text as a sequence of UTF-16 code units.



This localhost page can’t be found

No webpage was found for the web address: **http://localhost:36338/api/GetCities**

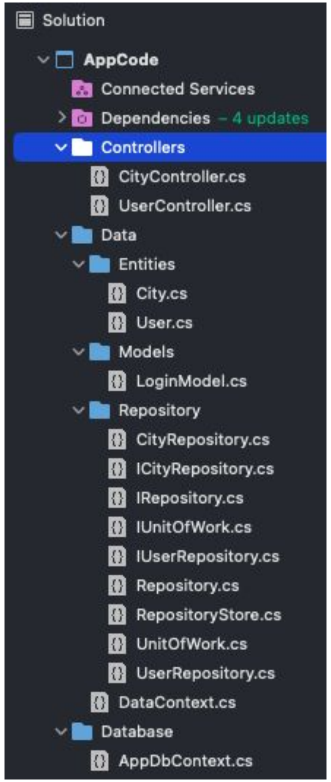
HTTP ERROR 404

What I tried

While going through the tutorial walkthroughs on Skillshare, I attempted to compare any useful information with the code I was presented.

It appears to me I would need to write out code that is missing from some of the methods provided. This may include having to add Route headers and going down a deeper rabbit hole of understanding the relationship of the Controllers with the Data Entities, Models, and Repositories, and how they all funnel into the Database for storage, modification, and retrieval.

I feel at this point, without an experienced programmer to help guide me through the coding, it would take me more than the allotted time of 1 week to complete this project.



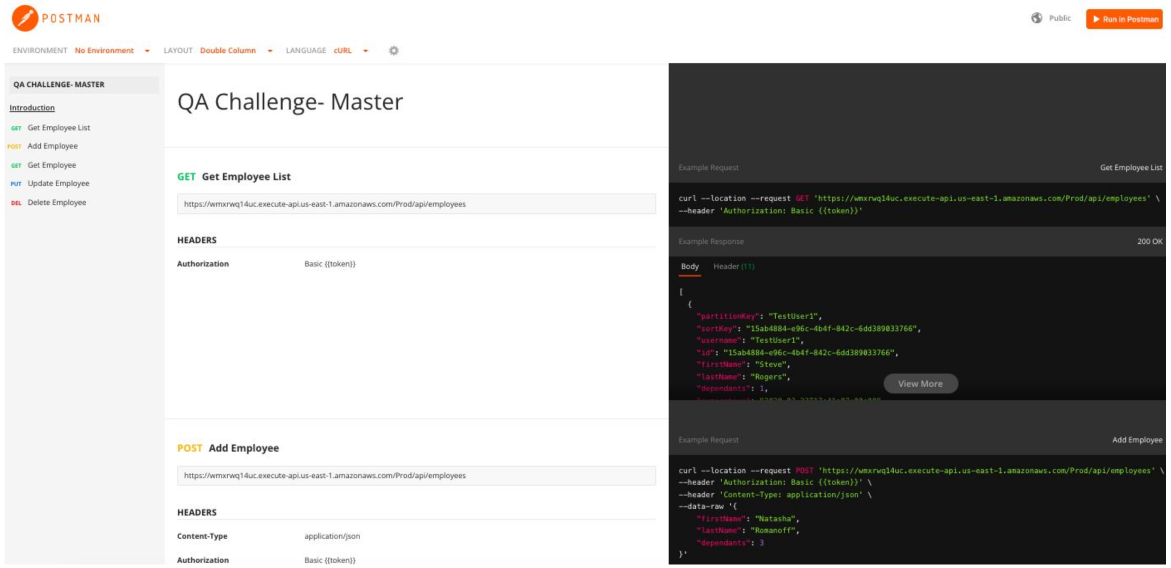
```
Solution
  AppCode
    Connected Services
    Dependencies – 4 updates
    Controllers
      CityController.cs
      UserController.cs
    Data
      Entities
        City.cs
        User.cs
      Models
        LoginModel.cs
      Repository
        CityRepository.cs
        ICityRepository.cs
        IRepository.cs
        IUnitOfWork.cs
        IUserRepository.cs
        Repository.cs
        RepositoryStore.cs
        UnitOfWork.cs
        UserRepository.cs
      DataContext.cs
    Database
      AppDbContext.cs
```

What I would do in a Perfect World

Accessible API

Easy access to API documentation

Given an auth token or an API call to extract a token



QA Challenge- Master

GET Get Employee List

https://wmxrwq14uc.execute-api.us-east-1.amazonaws.com/Prod/api/employees

HEADERS

Authorization	Basic {{token}}

Example Request — Get Employee List

```
curl --location --request GET 'https://wmxrwq14uc.execute-api.us-east-1.amazonaws.com/Prod/api/employees' \
--header 'Authorization: Basic {{token}}'
```

Example Response — 200 OK

Body Header (11)

```
[
  {
    "partitionKey": "TestUser1",
    "sortKey": "15ab4884-e96c-4b4f-842c-6dd389033766",
    "username": "TestUser1",
    "id": "15ab4884-e96c-4b4f-842c-6dd389033766",
    "firstName": "Steve",
    "lastName": "Rogers",
    "dependants": 1,
```

POST Add Employee

https://wmxrwq14uc.execute-api.us-east-1.amazonaws.com/Prod/api/employees

HEADERS

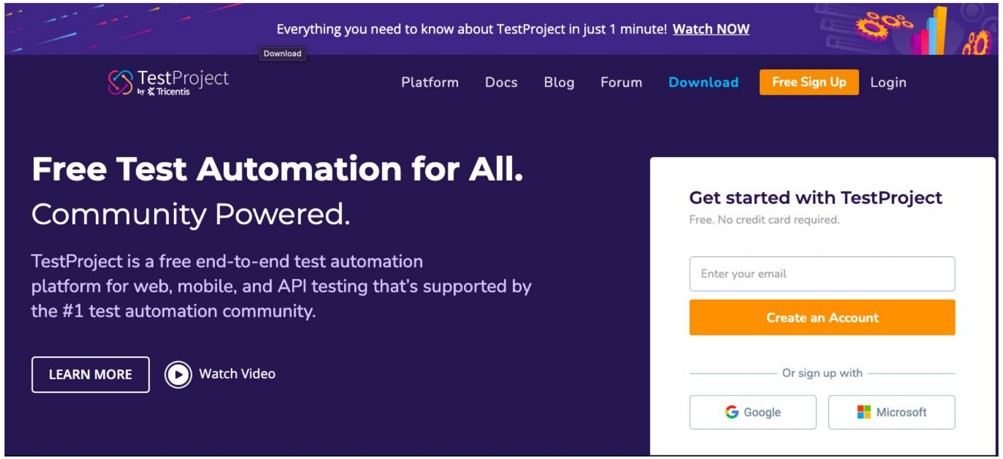
Content-Type	application/json
Authorization	Basic {{token}}

Example Request — Add Employee

```
curl --location --request POST 'https://wmxrwq14uc.execute-api.us-east-1.amazonaws.com/Prod/api/employees' \
--header 'Authorization: Basic {{token}}' \
--header 'Content-Type: application/json' \
--data-raw '{
    "firstName": "Natasha",
    "lastName": "Romanoff",
    "dependants": 3
}'
```

Tool I would use

TestProject.io utilizing its RESTful API Client Addon microservices

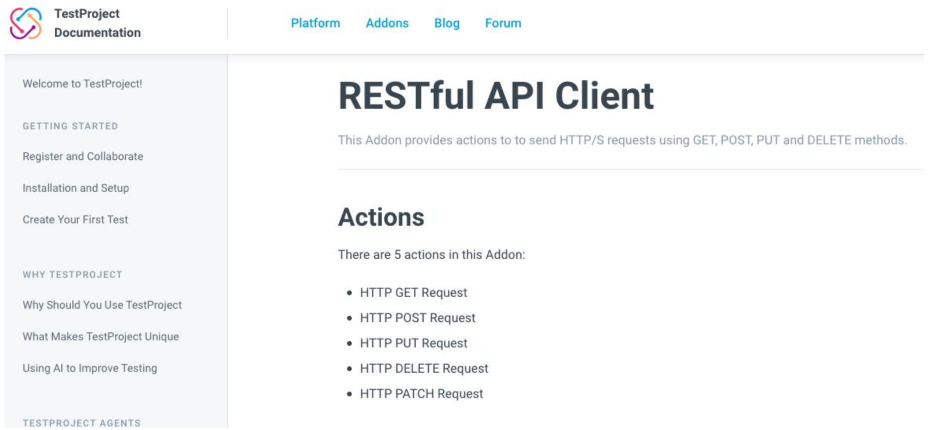


Everything you need to know about TestProject in just 1 minute! Watch NOW

Free Test Automation for All.
Community Powered.

TestProject is a free end-to-end test automation platform for web, mobile, and API testing that's supported by the #1 test automation community.

Get started with TestProject
Free. No credit card required.



RESTful API Client

This Addon provides actions to to send HTTP/S requests using GET, POST, PUT and DELETE methods.

Actions

There are 5 actions in this Addon:

- HTTP GET Request
- HTTP POST Request
- HTTP PUT Request
- HTTP DELETE Request
- HTTP PATCH Request

https://docs.testproject.io/testproject-addons/available-addons/restful-api-client-addon

Task 1: Create User Account

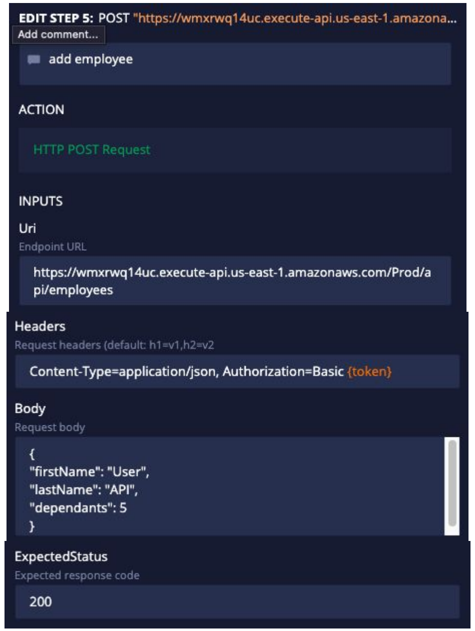
API for Create User Account would have a format like the following:

- POST Add Employee => POST Add User
- firstName => Name (String)
- lastName => Email (String)
- Dependents => Password (String)

Create a step to invoke the call and confirm a 200 status

Create additional steps to test edge cases:

- Errors from incorrect input (int instead of String)
- Submit empty values and confirm an error code response
- Character limits



EDIT STEP 5: POST "https://wmxrwq14uc.execute-api.us-east-1.amazona...

Add comment...

add employee

ACTION

HTTP POST Request

INPUTS

Uri
Endpoint URL

https://wmxrwq14uc.execute-api.us-east-1.amazonaws.com/Prod/api/employees

Headers
Request headers (default: h1=v1,h2=v2

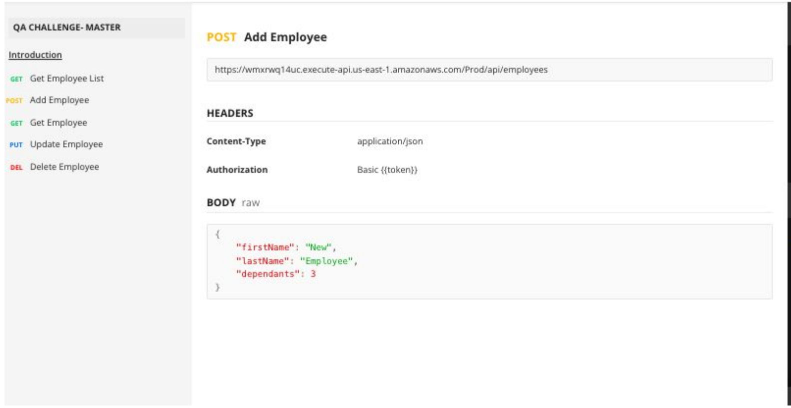
Content-Type=application/json, Authorization=Basic {token}

Body
Request body

```
{
"firstName": "User",
"lastName": "API",
"dependants": 5
}
```

ExpectedStatus
Expected response code

200



QA CHALLENGE- MASTER

Introduction

- GET Get Employee List
- POST Add Employee
- GET Get Employee
- PUT Update Employee
- DEL Delete Employee

POST Add Employee

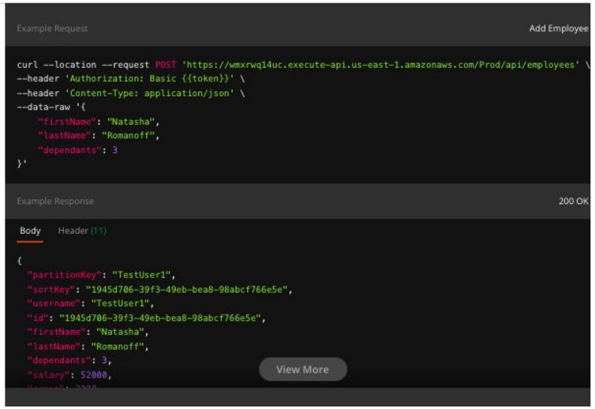
https://wmxrwq14uc.execute-api.us-east-1.amazonaws.com/Prod/api/employees

HEADERS

Content-Type	application/json
Authorization	Basic {{token}}

BODY raw

```
{
    "firstName": "New",
    "lastName": "Employee",
    "dependants": 3
}
```



Example Request — Add Employee

```
curl --location --request POST 'https://wmxrwq14uc.execute-api.us-east-1.amazonaws.com/Prod/api/employees' \
--header 'Authorization: Basic {{token}}' \
--header 'Content-Type: application/json' \
--data-raw '{
    "firstName": "Natasha",
    "lastName": "Romanoff",
    "dependants": 3
}'
```

Example Response — 200 OK

Body Header (11)

```
{
  "partitionKey": "TestUser1",
  "sortKey": "1945d706-39f3-49eb-bea8-98abcf766e5e",
  "username": "TestUser1",
  "id": "1945d706-39f3-49eb-bea8-98abcf766e5e",
  "firstName": "Natasha",
  "lastName": "Romanoff",
  "dependants": 3,
  "salary": 52000,
```

View More

Task 1A: Login Authentication

API for Login User would have a format like the following:

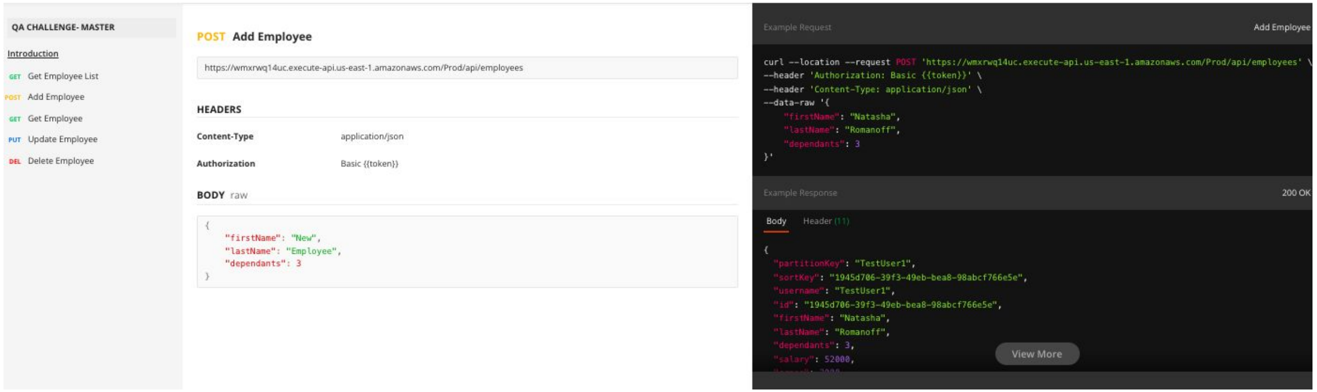
- POST Add Employee => POST Login User
- firstName => Username (String)
- lastName => Password (String)

Create a step to invoke the call and confirm a 200 status for an existing user

Create a step to invoke the call and confirm an error code for a non-existent user

Create additional steps to test edge cases:

- Errors from incorrect input (int instead of String)
- Submit empty values and confirm an error code response
- Character limits



QA CHALLENGE- MASTER

Introduction

- GET Get Employee List
- POST Add Employee
- GET Get Employee
- PUT Update Employee
- DEL Delete Employee

POST Add Employee

https://wmxrwq14uc.execute-api.us-east-1.amazonaws.com/Prod/api/employees

HEADERS

Content-Type	application/json
Authorization	Basic {{token}}

BODY raw

```
{
    "firstName": "New",
    "lastName": "Employee",
    "dependants": 3
}
```

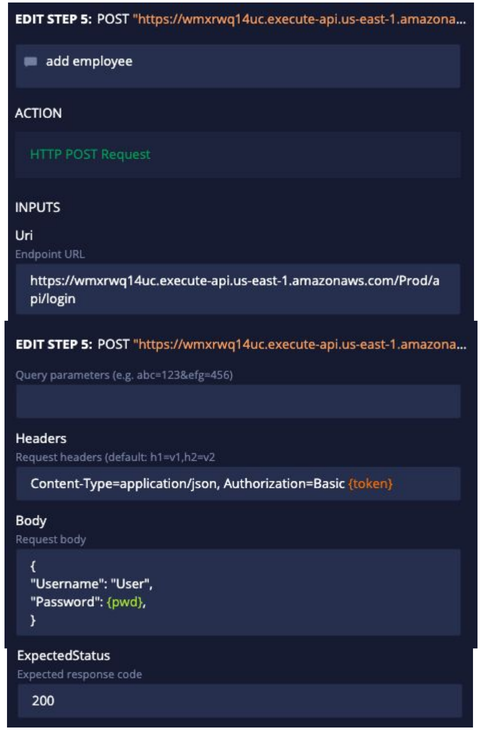
Example Request — Add Employee

```
curl --location --request POST 'https://wmxrwq14uc.execute-api.us-east-1.amazonaws.com/Prod/api/employees' \
--header 'Authorization: Basic {{token}}' \
--header 'Content-Type: application/json' \
--data-raw '{
    "firstName": "Natasha",
    "lastName": "Romanoff",
    "dependants": 3
}'
```

Example Response — 200 OK

Body Header (11)

```
{
  "partitionKey": "TestUser1",
  "sortKey": "1945d706-39f3-49eb-bea8-98abcf766e5e",
  "username": "TestUser1",
  "id": "1945d706-39f3-49eb-bea8-98abcf766e5e",
  "firstName": "Natasha",
  "lastName": "Romanoff",
  "dependants": 3,
  "salary": 52000,
```



EDIT STEP 5: POST "https://wmxrwq14uc.execute-api.us-east-1.amazona...

add employee

ACTION

HTTP POST Request

INPUTS

Uri
Endpoint URL

https://wmxrwq14uc.execute-api.us-east-1.amazonaws.com/Prod/api/login

EDIT STEP 5: POST "https://wmxrwq14uc.execute-api.us-east-1.amazona...

Query parameters (e.g. abc=123&efg=456)

Headers
Request headers (default: h1=v1,h2=v2

Content-Type=application/json, Authorization=Basic {token}

Body
Request body

```
{
"Username": "User",
"Password": {pwd},
}
```

ExpectedStatus
Expected response code

200

Task 2: Remove City

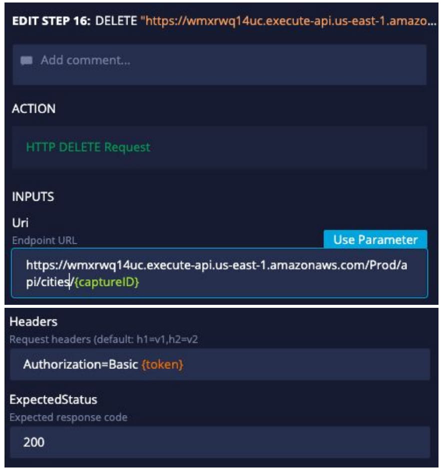
API for Delete City would have a format like the following:

- DEL Delete Employee => DEL Delete City
- Use route '/api/cities/{id}'

Create a step to invoke the call and confirm a 200 status that a city was created

Create additional steps to test edge cases:

- No id in url
- Enter characters as an id
- Enter non-existent id



EDIT STEP 16: DELETE "https://wmxrwq14uc.execute-api.us-east-1.amazo...

Add comment...

ACTION

HTTP DELETE Request

INPUTS

Uri
Endpoint URL

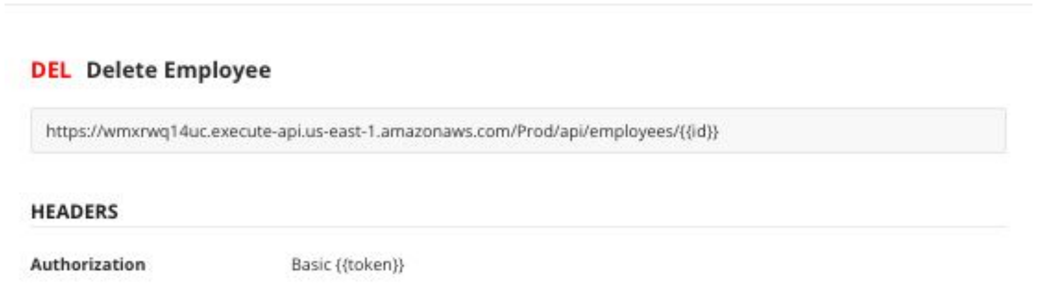
https://wmxrwq14uc.execute-api.us-east-1.amazonaws.com/Prod/api/cities/{captureID}

Headers
Request headers (default: h1=v1,h2=v2

Authorization=Basic {token}

ExpectedStatus
Expected response code

200



DEL Delete Employee

https://wmxrwq14uc.execute-api.us-east-1.amazonaws.com/Prod/api/employees/{{id}}

HEADERS

Authorization	Basic {{token}}

Task 2: Retrieve City

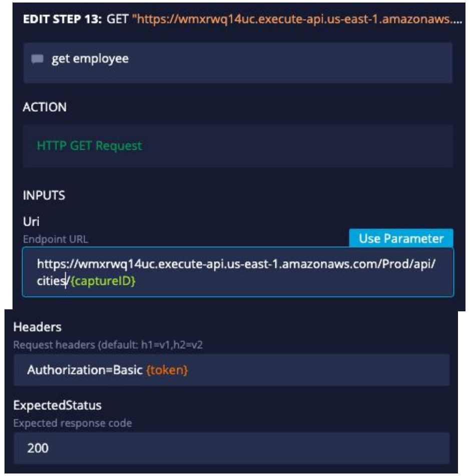
API for Retrieve City would have a format like the following:

- GET Get Employee => GET Get City
- Use route '/api/cities/{id}'

Create a step to invoke the call and confirm a 200 status that a city was created

Create additional steps to test edge cases:

- No id in url
- Enter characters as an id
- Enter non-existent id



EDIT STEP 13: GET "https://wmxrwq14uc.execute-api.us-east-1.amazonaws....

get employee

ACTION

HTTP GET Request

INPUTS

Uri
Endpoint URL

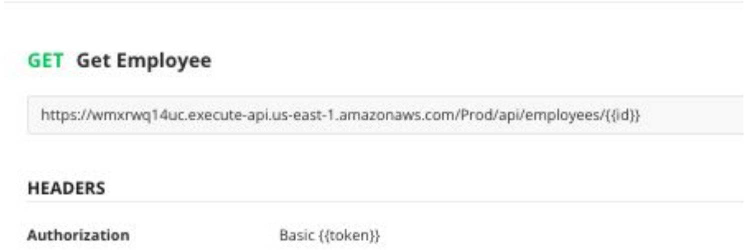
https://wmxrwq14uc.execute-api.us-east-1.amazonaws.com/Prod/api/cities/{captureID}

Headers
Request headers (default: h1=v1,h2=v2

Authorization=Basic {token}

ExpectedStatus
Expected response code

200



GET Get Employee

https://wmxrwq14uc.execute-api.us-east-1.amazonaws.com/Prod/api/employees/{{id}}

HEADERS

Authorization Basic {{token}}