[**1. Redis Introduction** 1](#_Toc173063809)

[**2. Caching** 3](#_Toc173063810)

[**3. What is Redis** 4](#_Toc173063811)

[**4. Setup Docker for Redis** 5](#_Toc173063812)

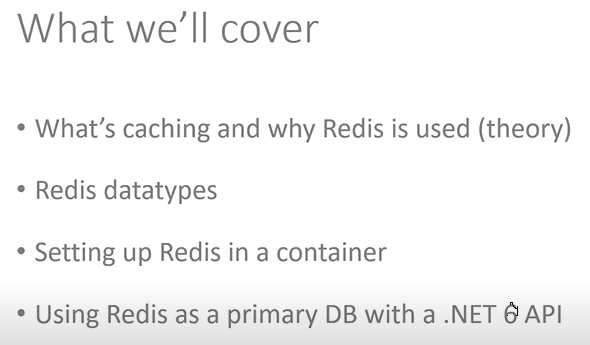
[**5. Redis Command Line Tool (REDIS – CLI)** 7](#_Toc173063813)

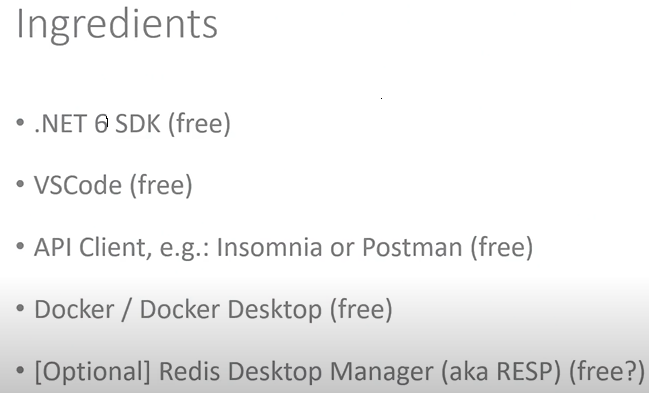
[**6. Redis Data Type** 7](#_Toc173063814)

[**7. Project Setup** 8](#_Toc173063815)

# **1. Redis Introduction**







A white background with black text

Description automatically generated

A screenshot of a data type

Description automatically generated

A white background with black text

Description automatically generated

# **2. Caching**

Case 1.

A diagram of a data flow

Description automatically generated

Case 2.

A diagram of a data processing process

Description automatically generated

Case 3.

A diagram of a data flow

Description automatically generated

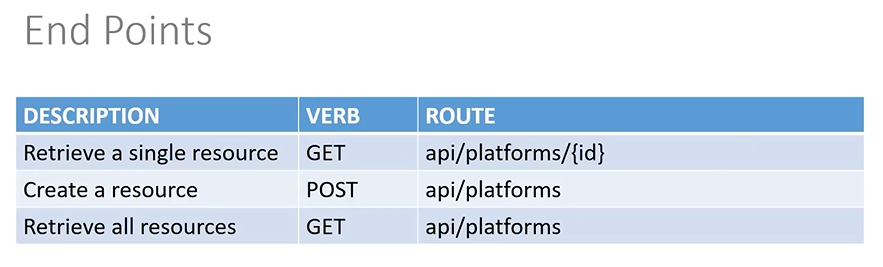
# **3. What is Redis**

A screenshot of a computer

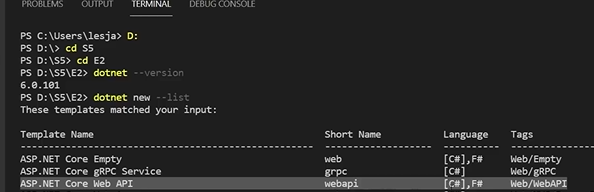
Description automatically generated

A diagram of a software company

Description automatically generated



Create ASP.NET Core Web API in VS Code



A black background with white text

Description automatically generated

PS D:\ss\E2> code RedisAPI -r (It open the project)

# **4. Setup Docker for Redis**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

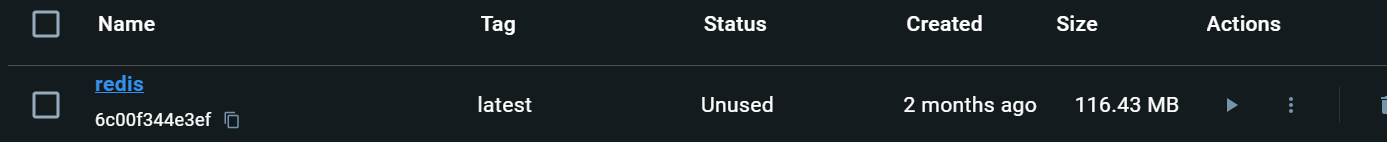
1. Create docker compose file. Define the image and container name along with above ports.

Pull the redis image from the docker hub and run the below command in Visual Studio Developer Command Prompt.

A blue and white screen with black text

Description automatically generated

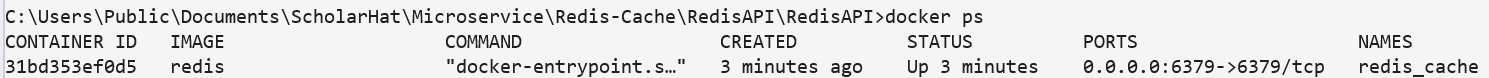
1. Pulled redis image is showing in the docker desktop.



1. Run the container using below command in Visual Studio Developer command prompt to run the container. **(docker compose down)** to **remove** and **(docker compose stop)** to **stop** the container.



1. Run the below command, it shows running container name. ( As we can see in Docker Desktop also)



A black rectangle with white dots

Description automatically generated

# **5. Redis Command Line Tool (REDIS – CLI)**

1. Run the below command in Visual Studio Developer Command Prompt.







Check weather it running or not by using the below command. (enter ping, and gives PONG as response)

A number on a white background

Description automatically generated

# **6. Redis Data Type**

A screenshot of a phone

Description automatically generated

1. Run the below command



A close up of a text

Description automatically generated





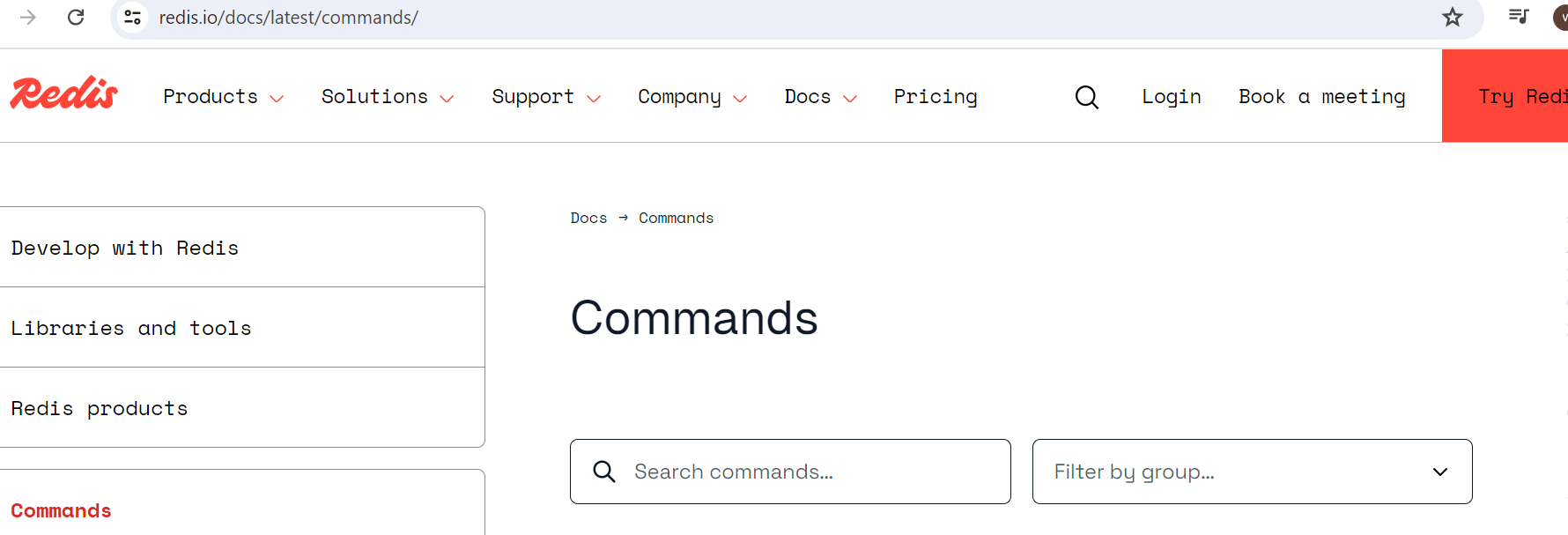




A black text with a arrow pointing to the left

Description automatically generated with medium confidence

Please check the web site for reference (<https://redis.io/commands>)

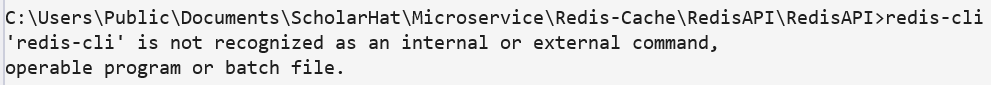


1. Exit from the Redis command prompt by using (**exit**) command.
2. First exit command represent exit connection from the redis database.
3. Second exit command exit from the interactive terminal command of redis container.

A close-up of a computer code

Description automatically generated

Once you exit and check for redis-cli command, it gives you below error.



# **7. Project Setup**

1. Intstall the below packages

A screenshot of a computer program

Description automatically generated A screenshot of a computer program

Description automatically generated

A close up of a computer screen

Description automatically generated

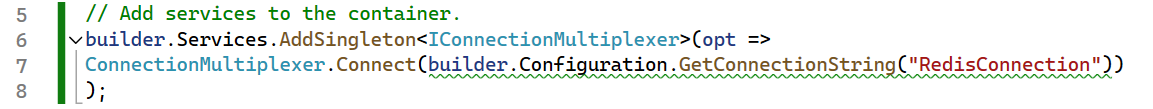
1. We are going to use Package number 2 for IConnectionMultiplexer approach. Define the connection string in appsettings.Development.json and add its services to the container in Programm.cs file.

"ConnectionStrings": {

"RedisConnection": "127.0.0.1:6379"

}

Programm.cs



1. Add a Model
2. Create a Repository under Data folder (Interface and its implementation.)
3. Add a Services to the container for repository.
4. Add a PlatformsController.
5. Run the container using below command and redis command.

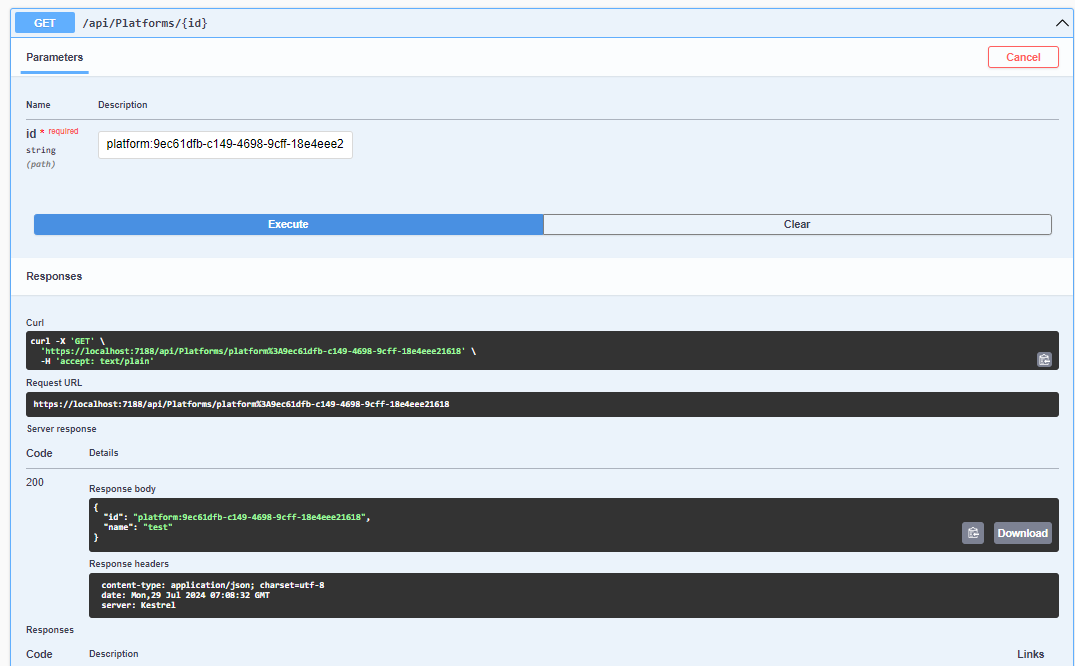
A close-up of a white background

Description automatically generated

8. Install the Redis Insight (exe) and Test connection. Enter the details like credentials and Database name. Keep default Database name. Select the database and find the data using key type (here we use string type as key)

A screenshot of a computer

Description automatically generated



A screenshot of a computer

Description automatically generated

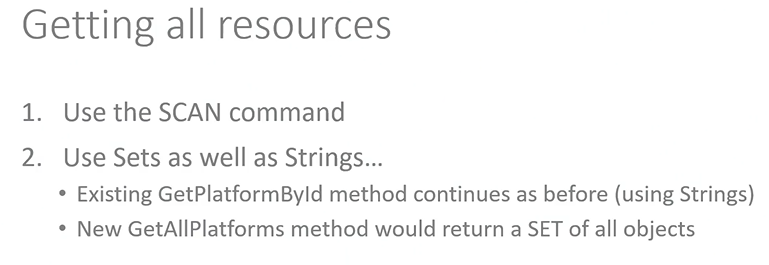
A screenshot of a computer

Description automatically generated

1. Add some platform like Docker, Kubernetes, Teraform using post method.

A screenshot of a computer

Description automatically generated



A number and numbers on a white background

Description automatically generated

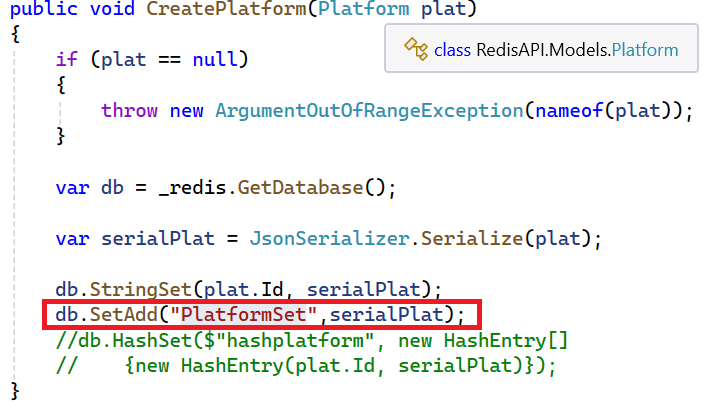
A screenshot of a phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated

Add the below highlighted code in repository implementation.



Then post the data as “name” : “Exchage” as we did earlier. The same can be see in Redis Insight.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

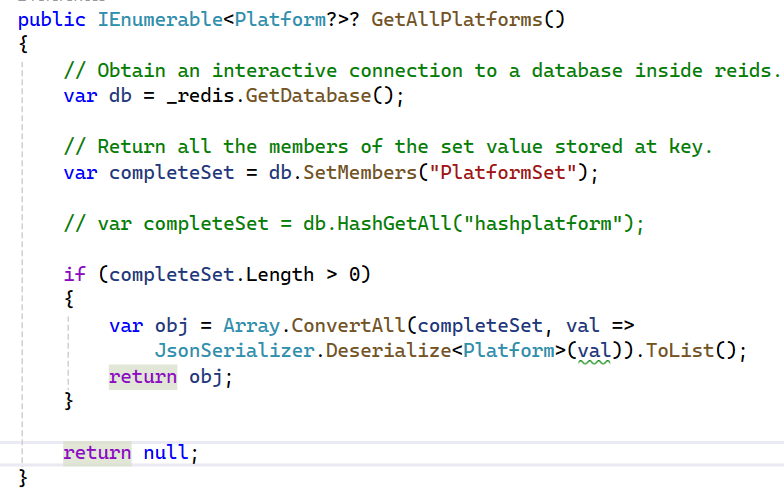
Description automatically generated

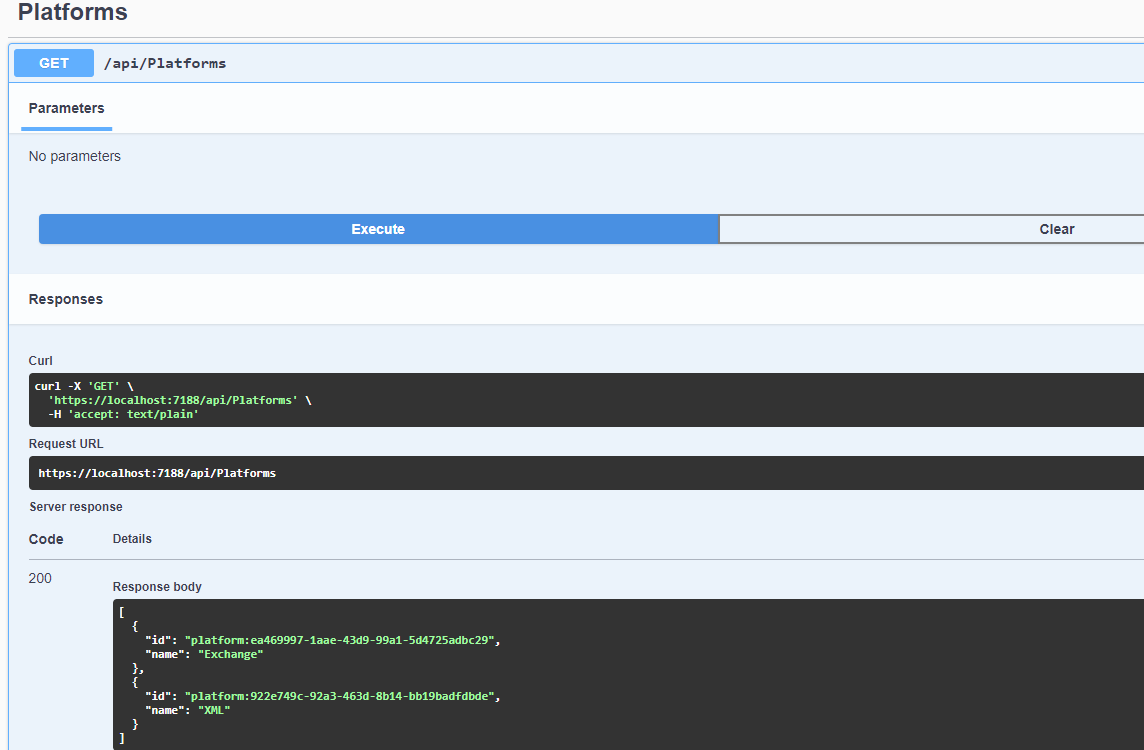
A screenshot of a computer

Description automatically generated

Post the data as (Name: “XML”)







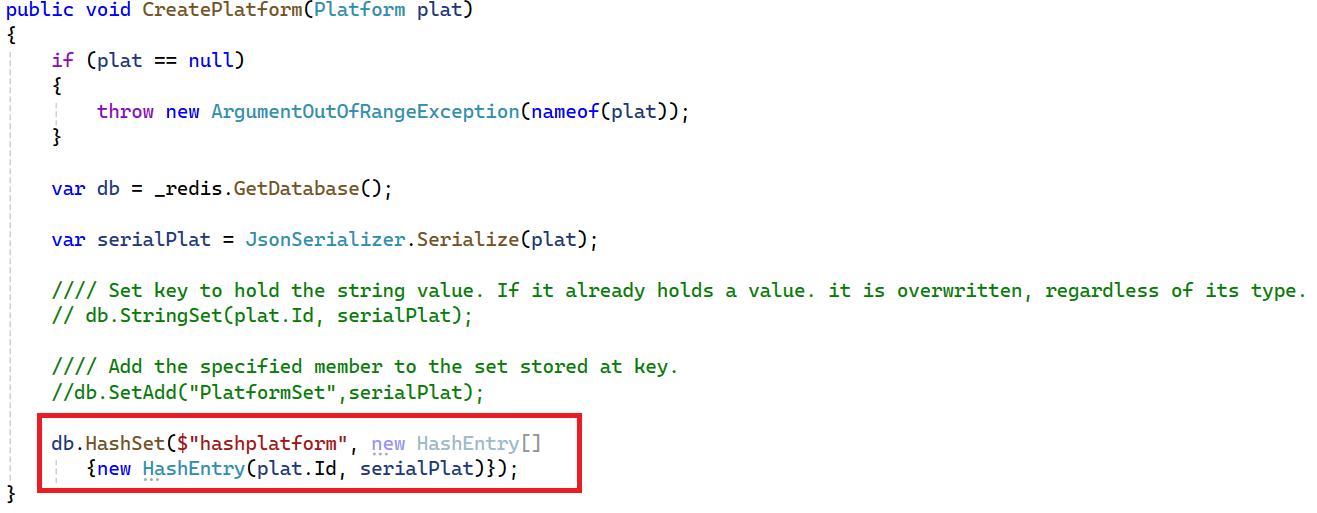
A white background with text

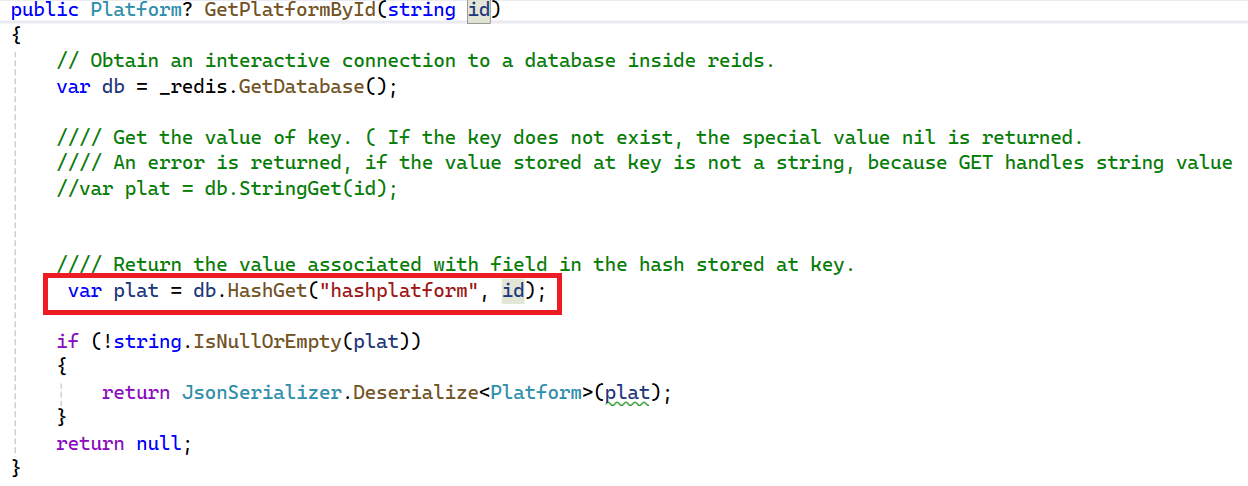
Description automatically generated

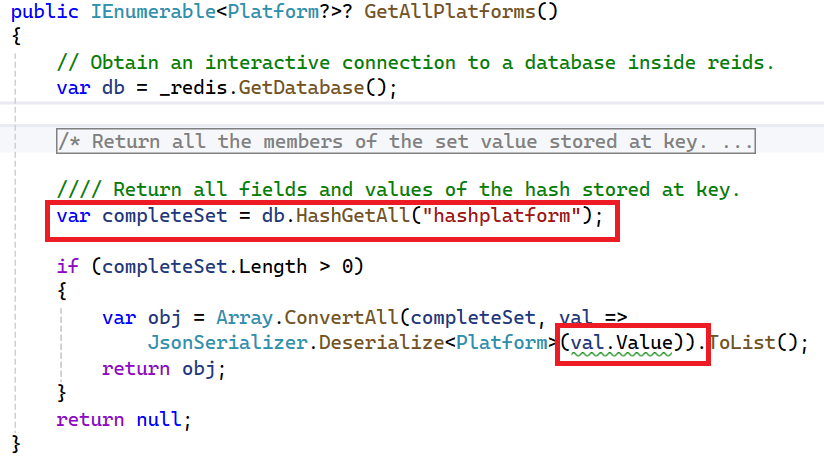
A screenshot of a computer

Description automatically generated

In Repository add the below code for hash







Post the data (“name” : “Redis”) and (“name”: “Python”)

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

We Scan the database 0 by using scan 0 command in Redis insight cli command or Visual Studio Developer Command Prompt.

A screenshot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generated

Stop the docker container

**>docker ps** (Get list of running containers)

**>docker compose stop** ( Stop the docker containers)