

DOT NET CORE-AZURE

MINI-PROJECT

Create a **Web API Project** to store Product Information. Use Entity Framework to store the product information in the database. The user should be able to perform all the CRUD Operations. Configure **GET, POST, PUT and DELETE**.

The Product Entity should have the following properties:

- Product ID
- ProductName
- Price
- Brand
- Manufacture Date
- Expiration Date

Use Data Annotations to

- Mark the Primary Key
- Make ProductName Mandatory
- Make Price a Number

Create a jQuery and AJAX Client to consume the Web API and show the result.

Azure Hosting:

- Host the web api in azure and consume the same using jQuery Client.
- Configure Scale out by adding rules for custom scaling
- Configure Deployment slots for staging and production
- Configure Application Insights for the project
- Configure Swagger for the api
- Work with Log Analytics with the sample logs available

Steps to create web api project:-

1.From the File menu,select New > Project.

2.Enter Web API in the search box.

3.Select the **ASP.NET Core Web API** template and select Next.

4.In the Configure your new project dialog, name the project "**MiniProject**" and select Next.

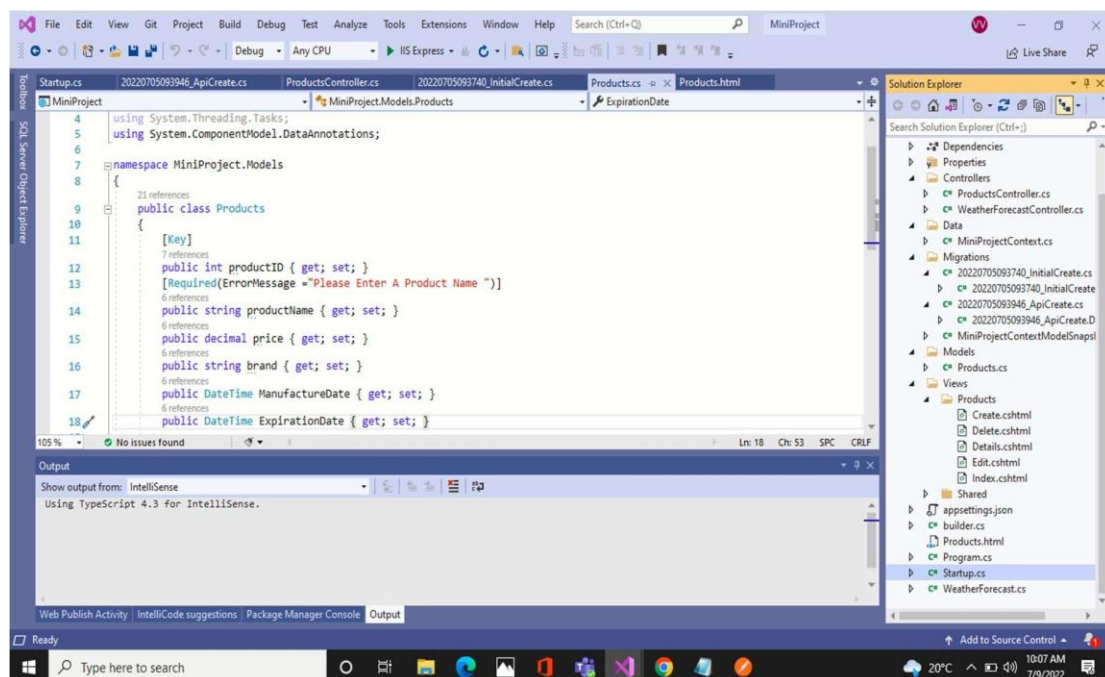
>>In the Additional information dialog: >>Confirm the Framework is .NET 6.0 (Long-term support). >>Confirm the checkbox for Use controllers(uncheck to use minimal APIs) is checked. >>Select **Create**.

5.Add a model class with properties:-

In Solution Explorer, right-click the project. Select "**Add >> New Folder**". Name the folder "**Models**". Right-click the Models folder and select "**Add >> Class**". Name the class "products" and select Add. >>product id >>product name. >>Price >>Brand >>Manufacture Date >>Expiration Date

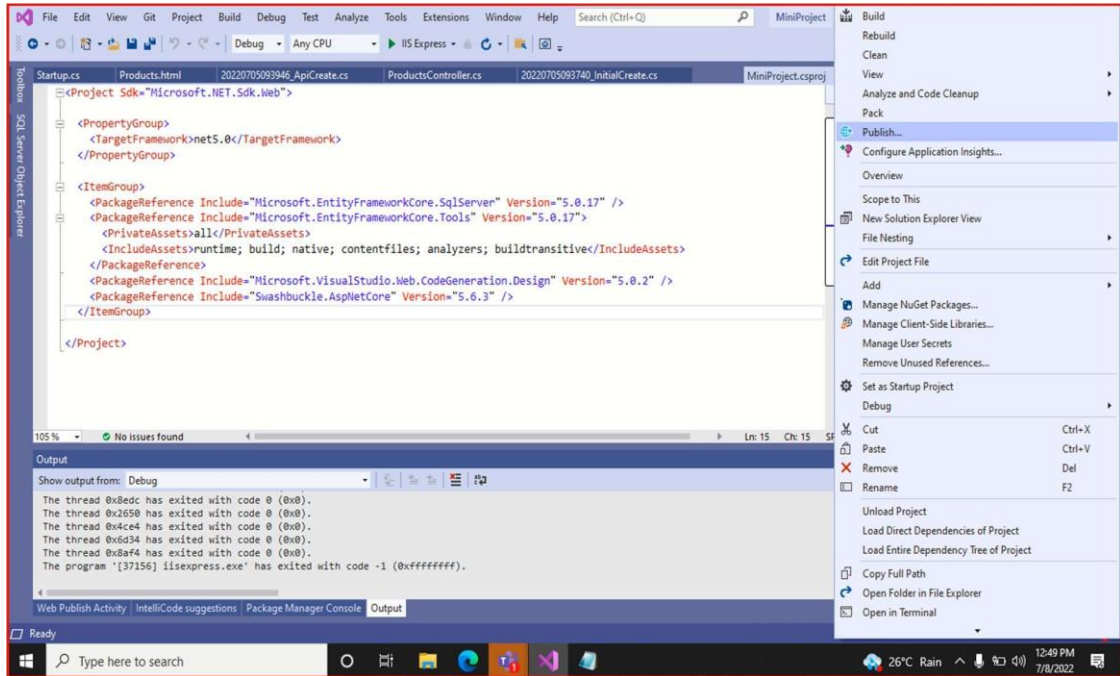
6.Add NuGet packages From the **Tools** menu, select **NuGet Package Manager >> Package Manager Console** for Solution.

7.Add Web-API Controller. In Solution Explorer, right-click on the controller **Create a web Api controller**.

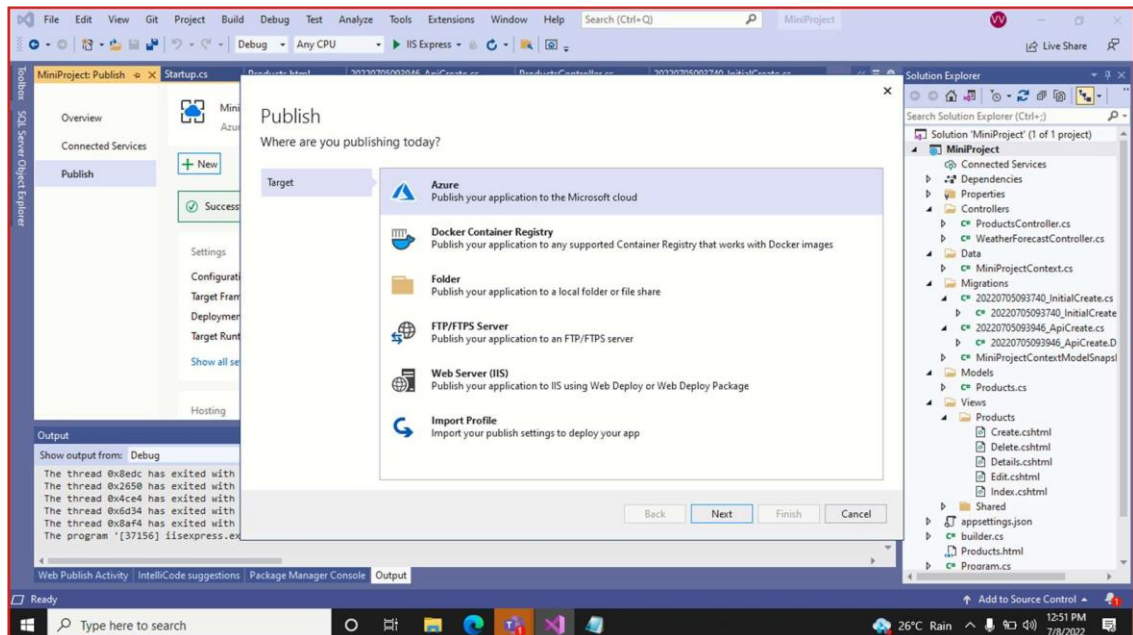


1. Host the web Api in azure and consume the same using jQuery Client.

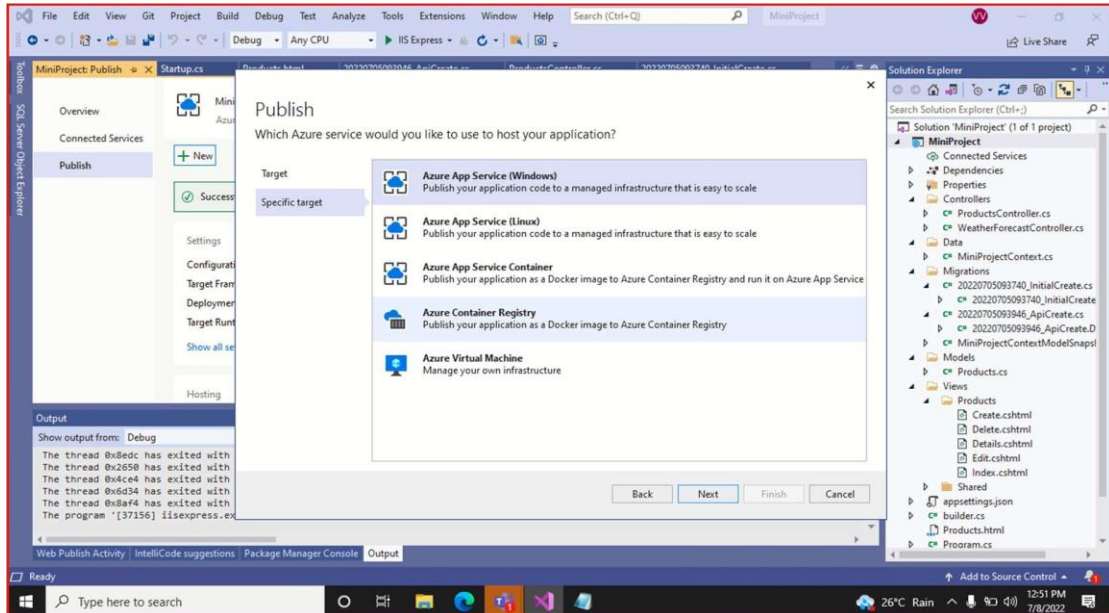
- In **Solution Explorer**, right-click the project and select **Publish**.



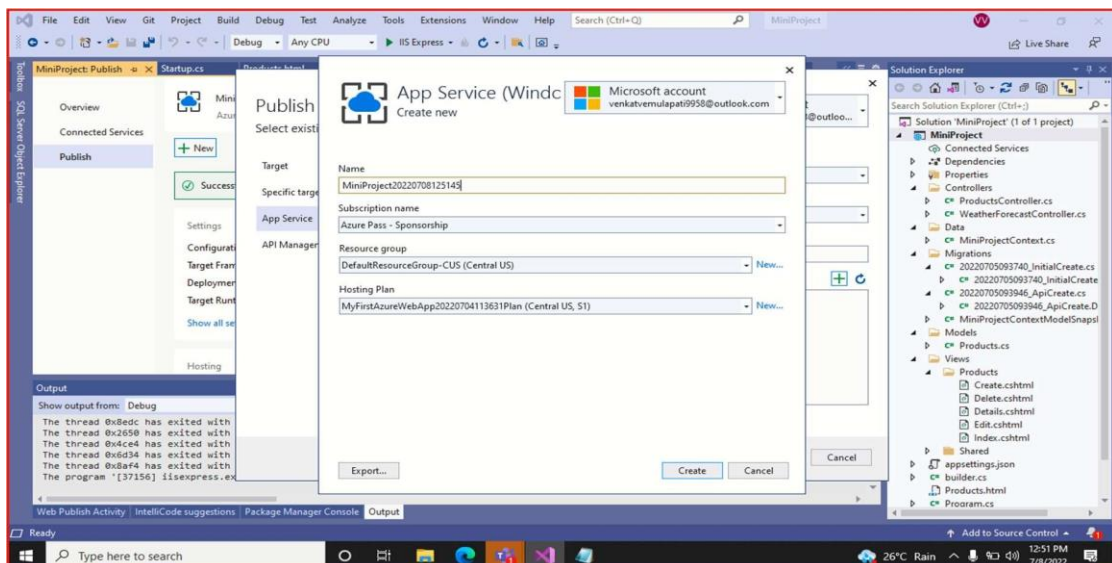
- In the **Publish** dialog, select **Azure** and select the **Next** button:



- Select **Azure App Service (Windows)** and select the **Next** button:

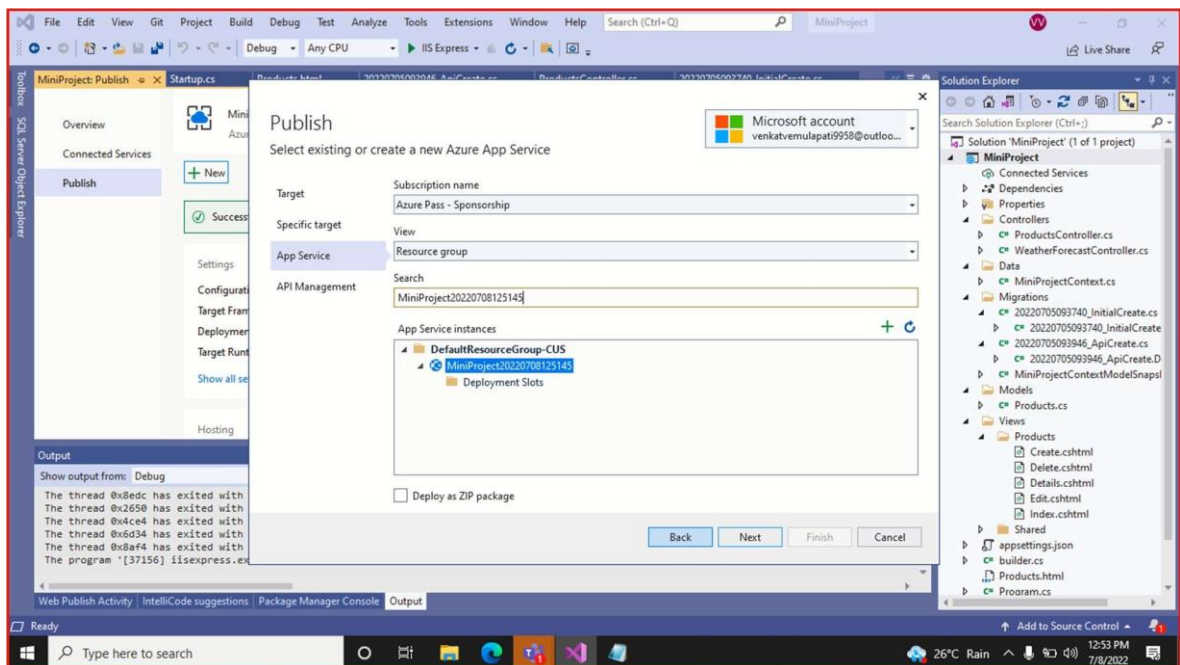


- Select **Create a new Azure App Service**.

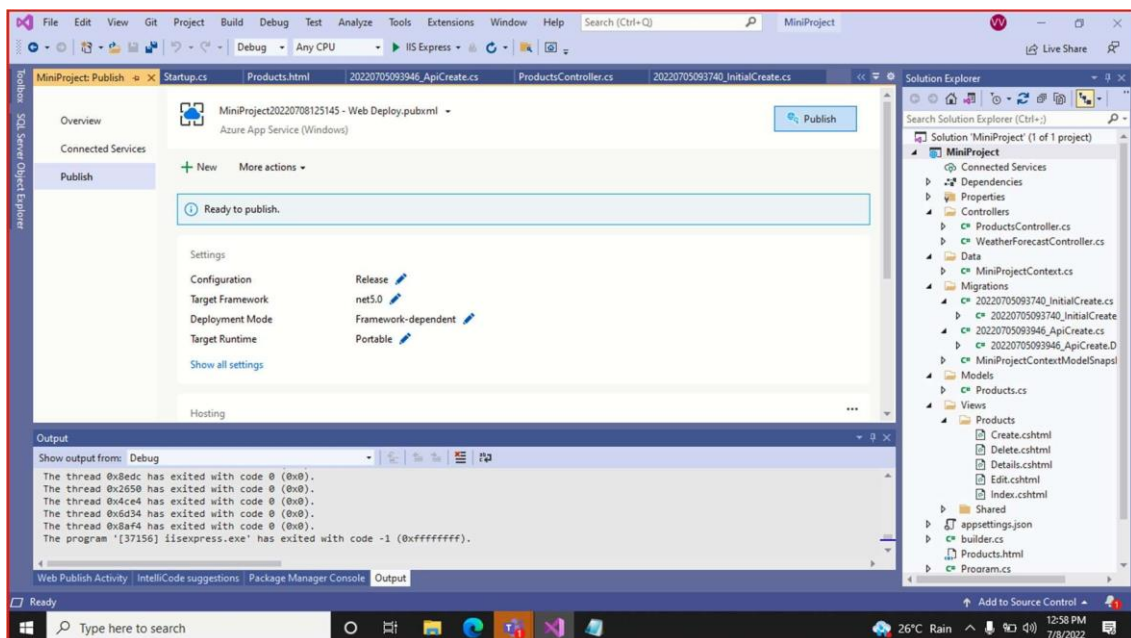


The **Create App Service** dialog appears. The **App Name**, **Resource Group**, and **App Service Plan** entry fields are populated. You can keep these names or change them.

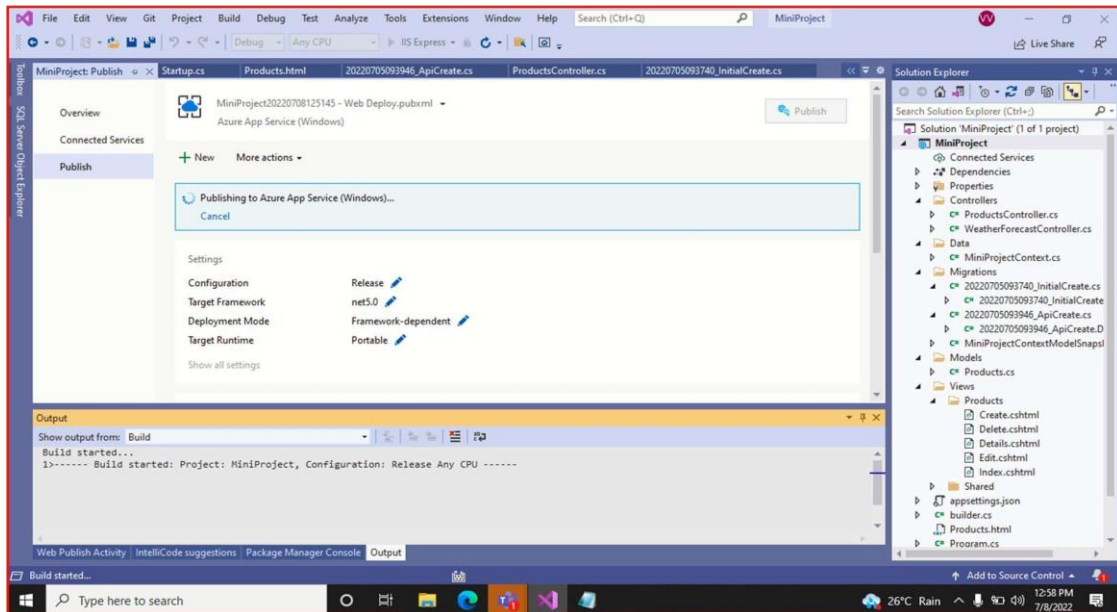
>> After creation is completed, the dialog is automatically closed and the **Publish** dialog gets focus again. The instance that was created is automatically selected.



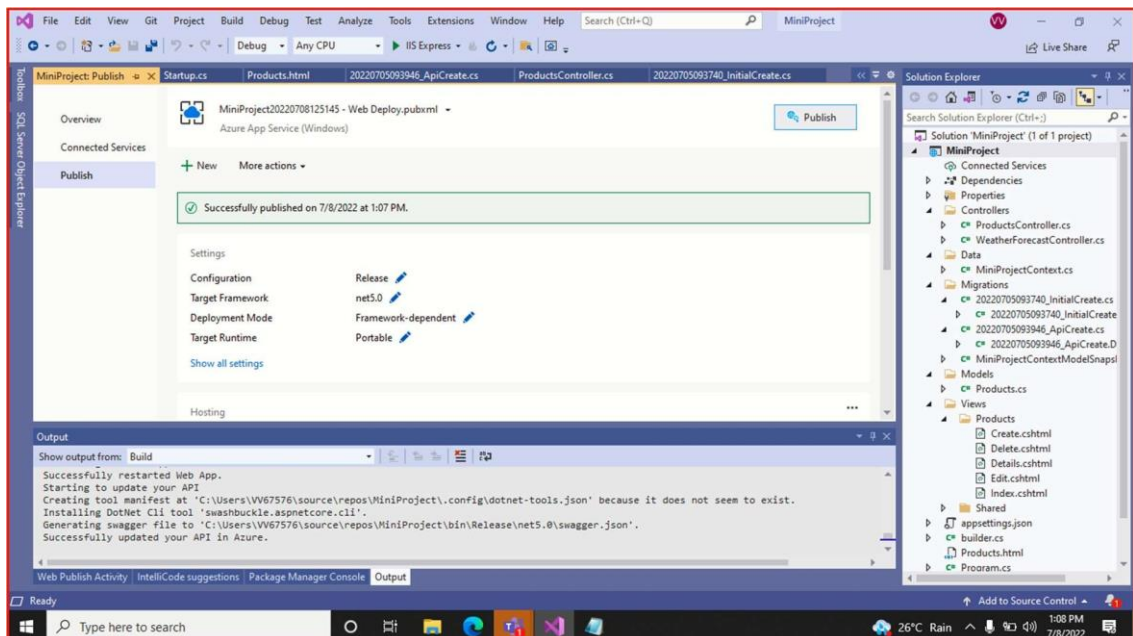
- After that project is **Ready to publish**



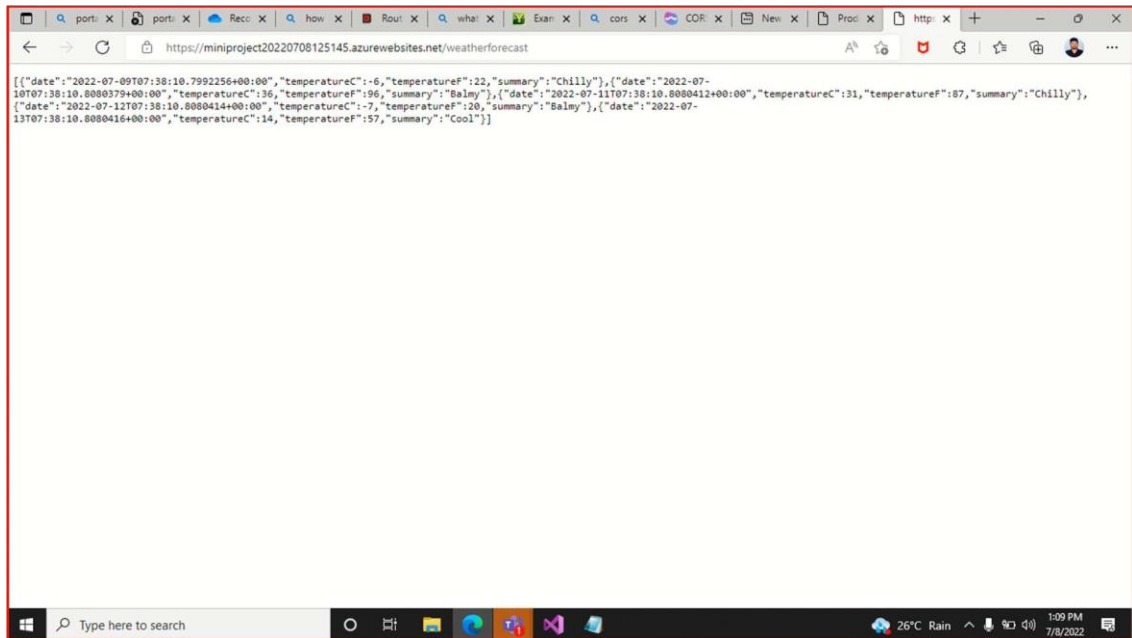
- Click on **publish**.



- After publishing successfully

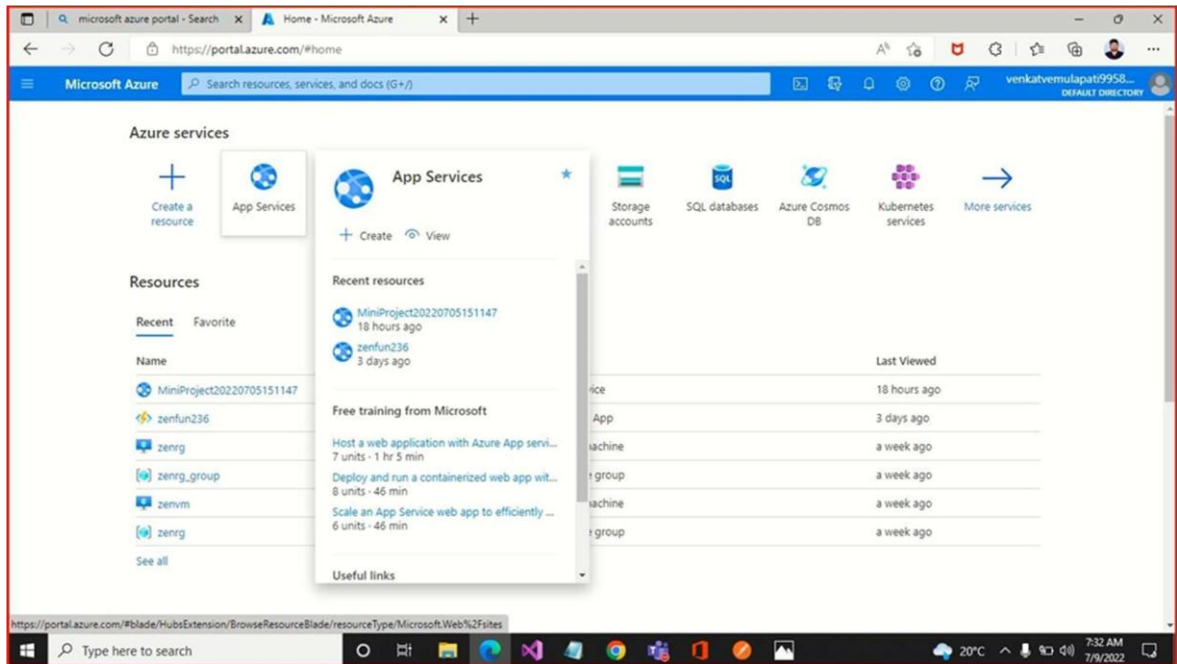


Run the project through IIS EXPRESS. Routing over the weatherforecast Controller it shows the JSON result



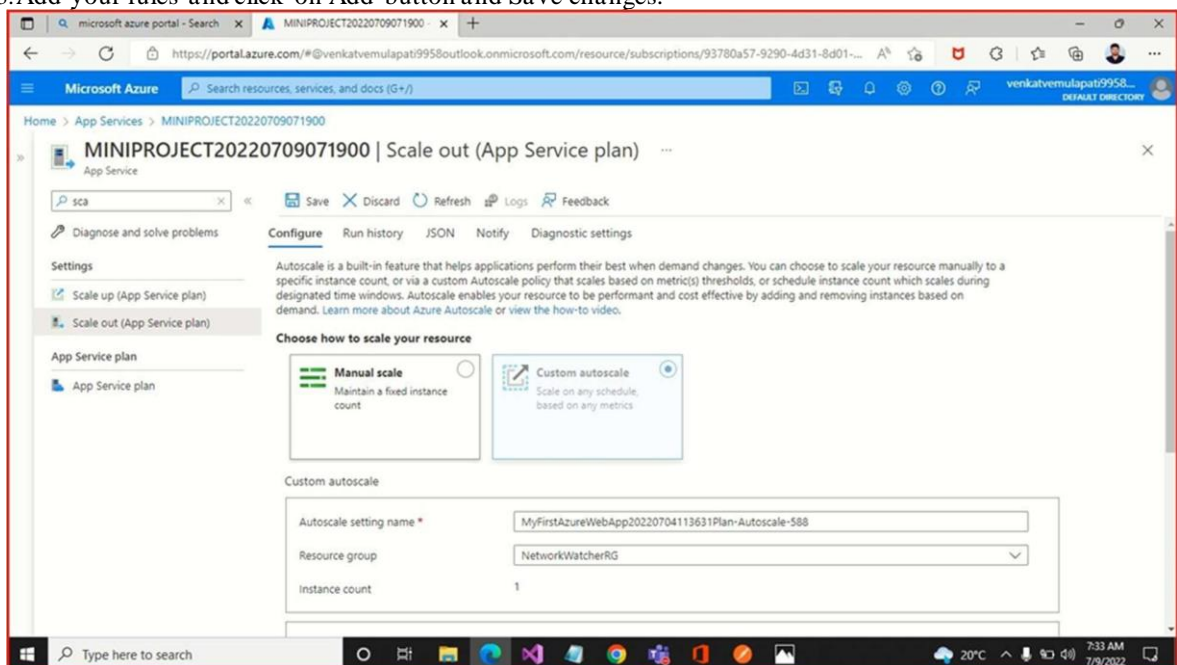
2. Configure Scale out by adding rules for custom scaling

A scale out operation is the equivalent of creating multiple copies of your web site and adding a load balancer to distribute the demand between them. When you scale out a web site in Windows Azure Web Sites there is no need to configure load balancing separately since this is already provided by the platform

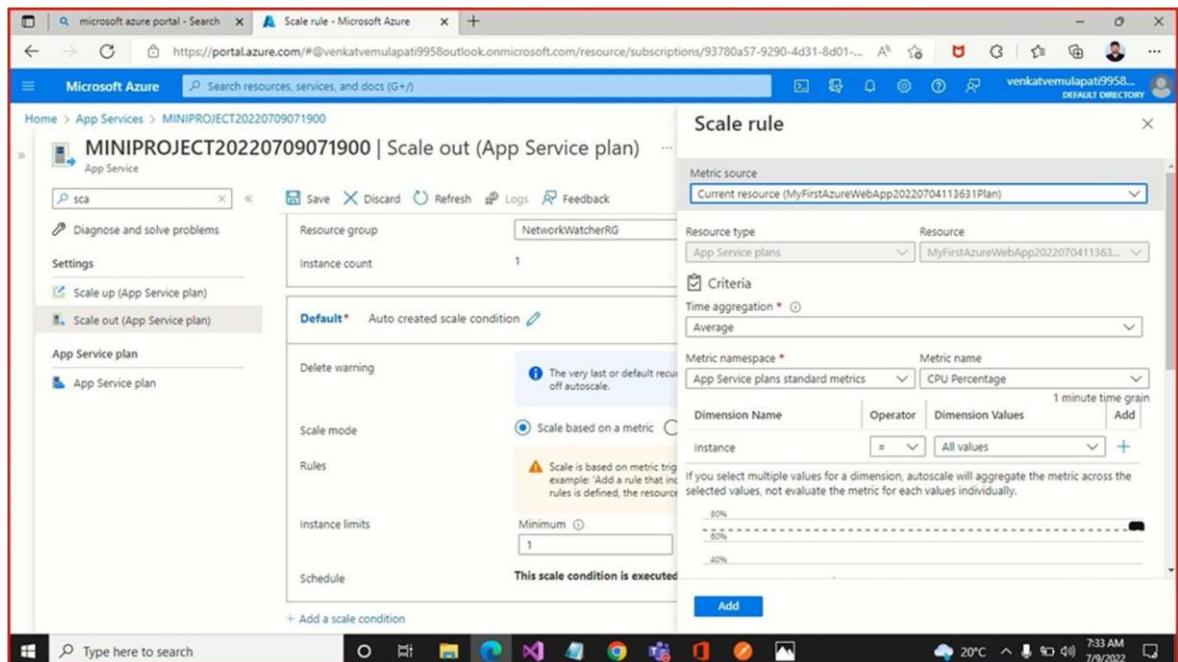


Follow the Below steps to add the scale out to our web site.

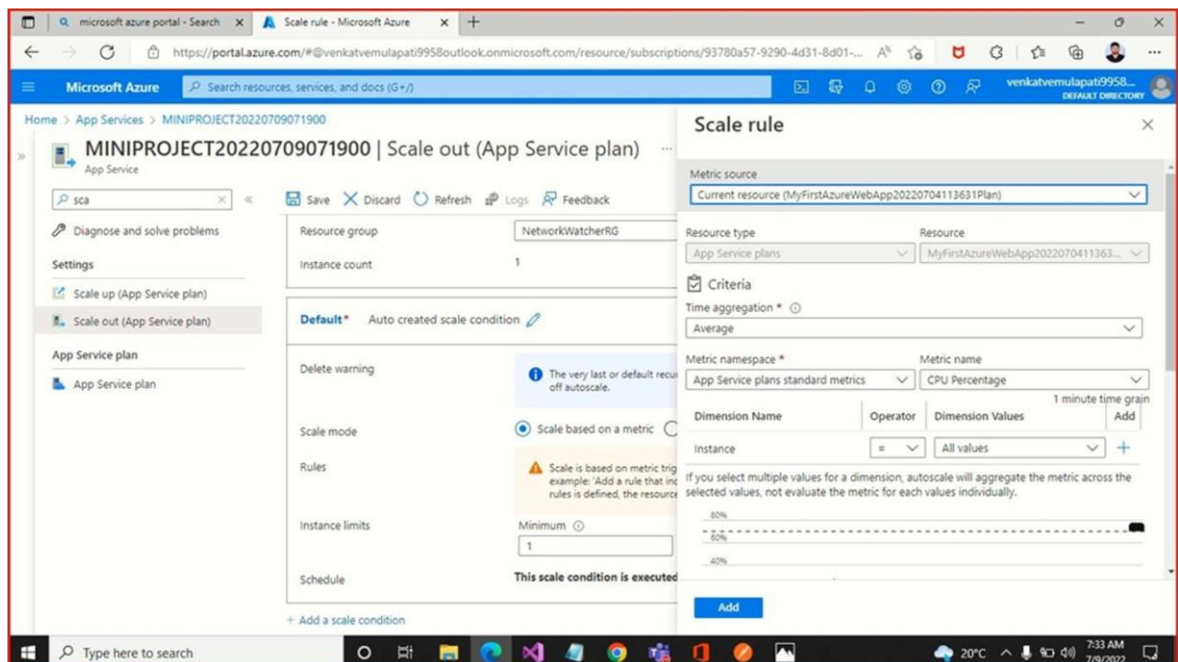
1. select Custom Auto Scale.
2. Click on Add a rule.
3. Add your rules and click on Add button and Save changes.



2. Click on Add a rule



3. Add your rules and click on Add button and Save changes.



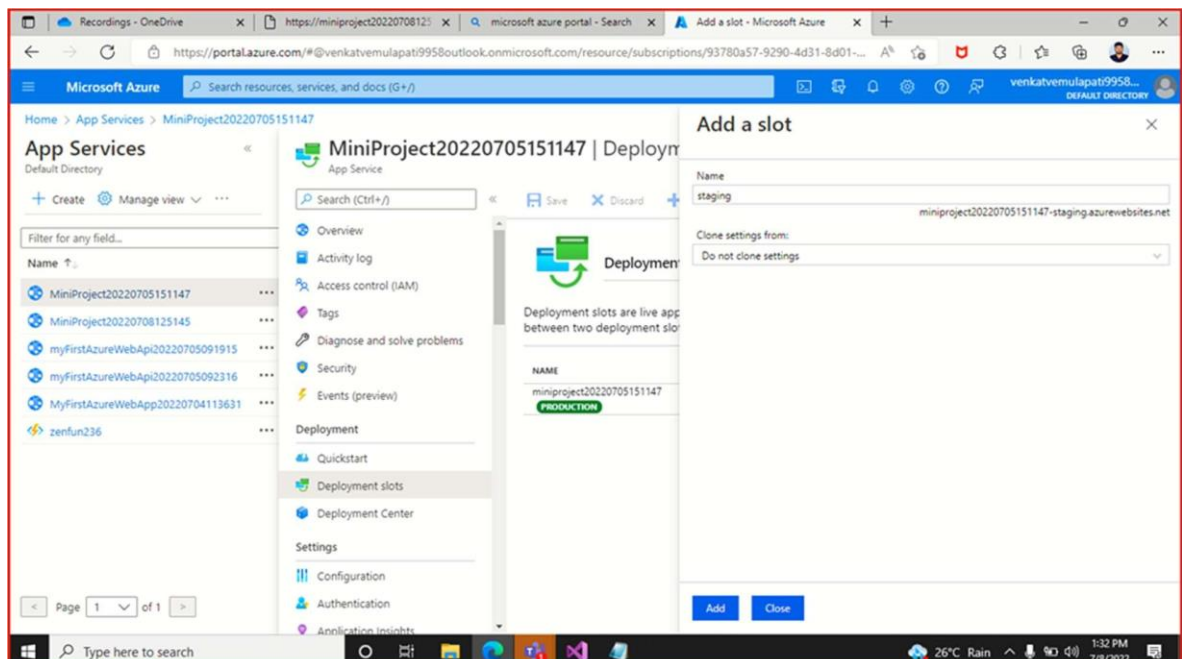
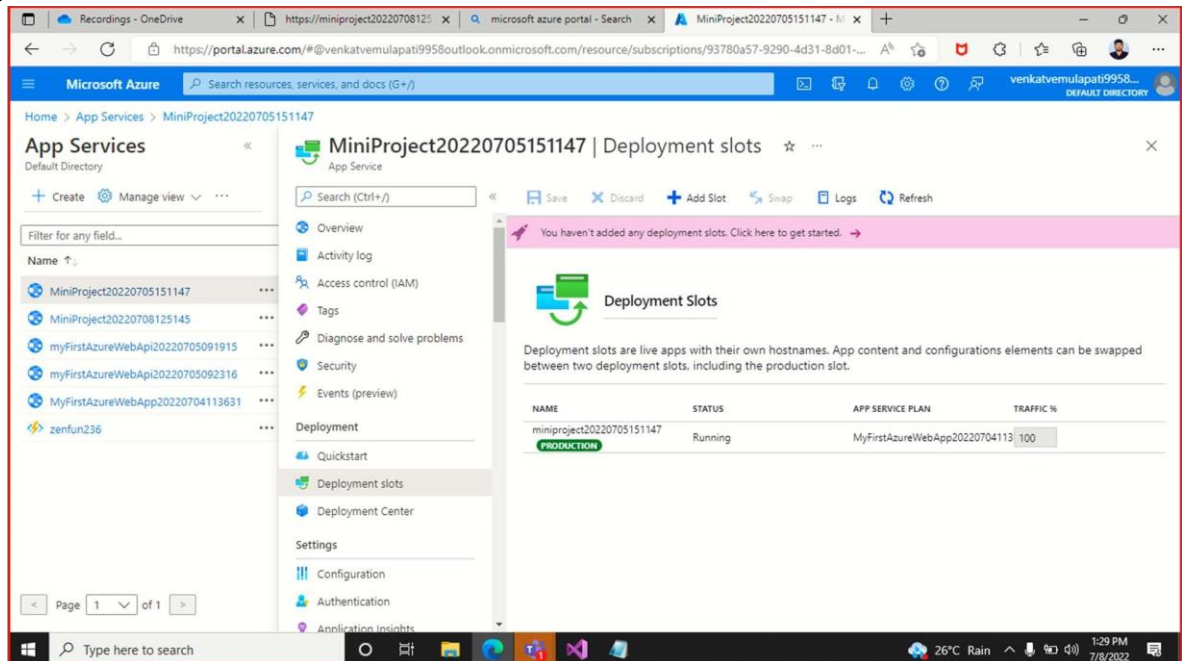
3.Configure Deployment slots for staging and production

Azure Functions deployment slots allow your function app to run different instances called "slots". Slots are different environments exposed via a publicly available endpoint. One app instance is always mapped to the production slot, and you can swap instances assigned to a slot on demand. Function apps running under the Apps Service plan may have multiple slots, while under the Consumption plan only one slot is allowed.

Follow the bellow images to add deployment slots.

1.Click on Add Slot.

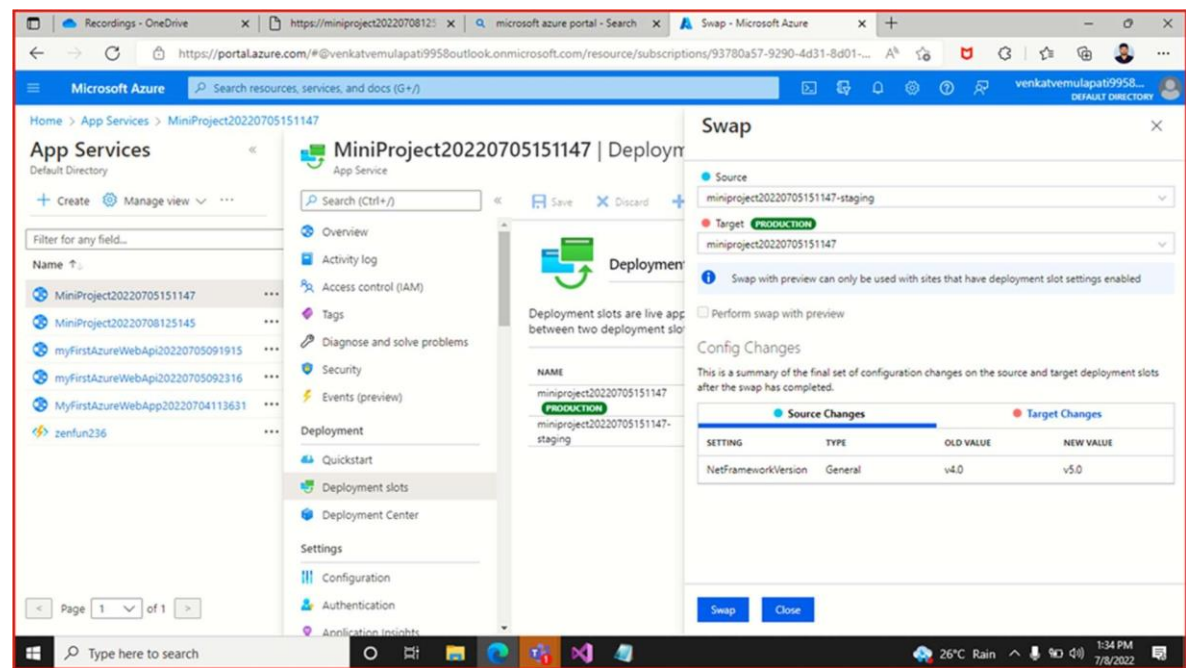
2.Enter The slot name and click on add button.



The following reflect how functions are affected by swapping slots:

Traffic redirection is seamless; no requests are dropped because of a swap. This seamless behavior is a result of the next function triggers being routed to the swapped slot.

Finally swapping the slots



4. Configure Application Insights for the project :-

Application Insights is a feature of [Azure Monitor](#) that provides extensible application performance management (APM) and monitoring for live web apps. Developers and DevOps professionals can use Application Insights to:

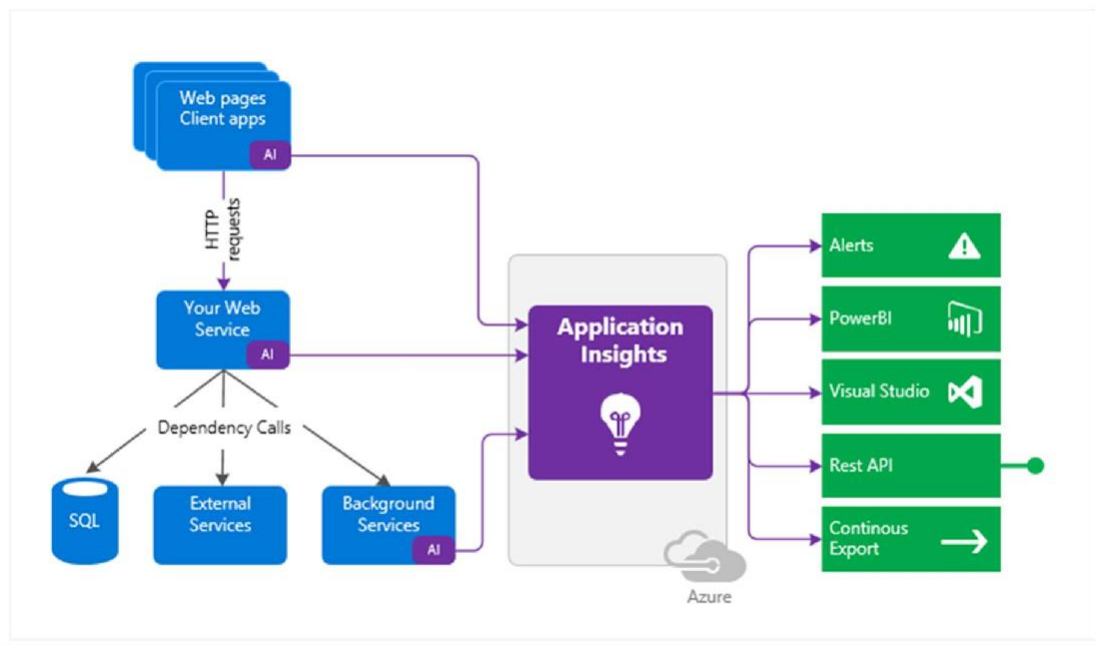
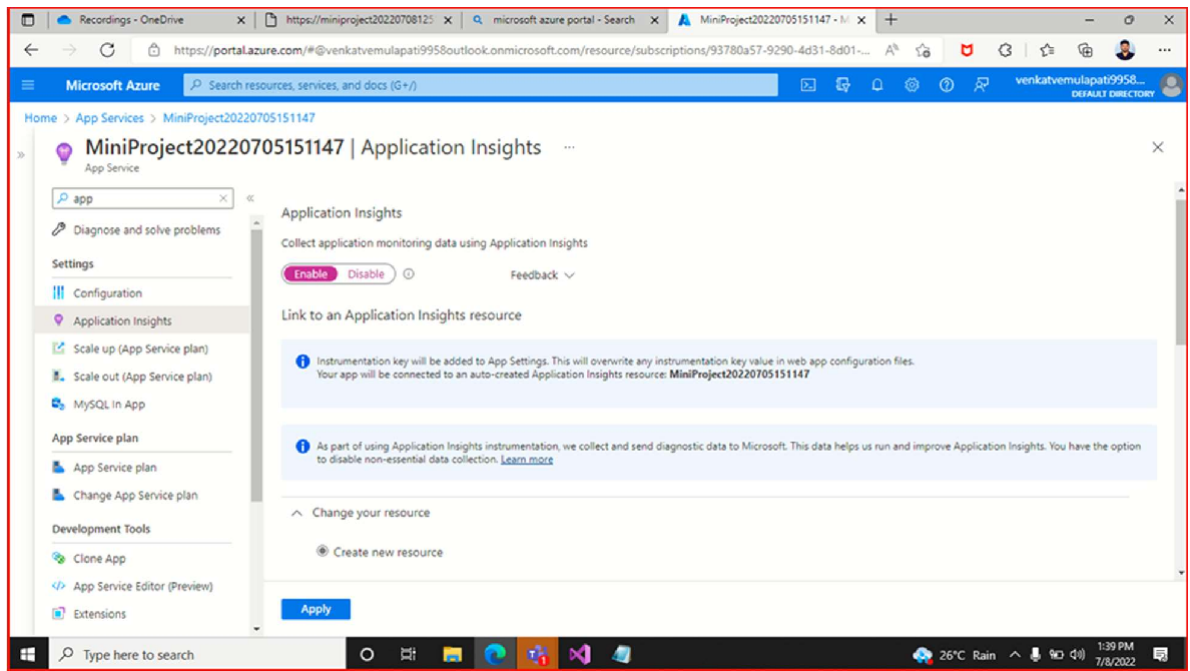
- Automatically detect performance anomalies.
- Help diagnose issues by using powerful analytics tools.
- See what users actually do with apps.
- Help continuously improve app performance and usability.

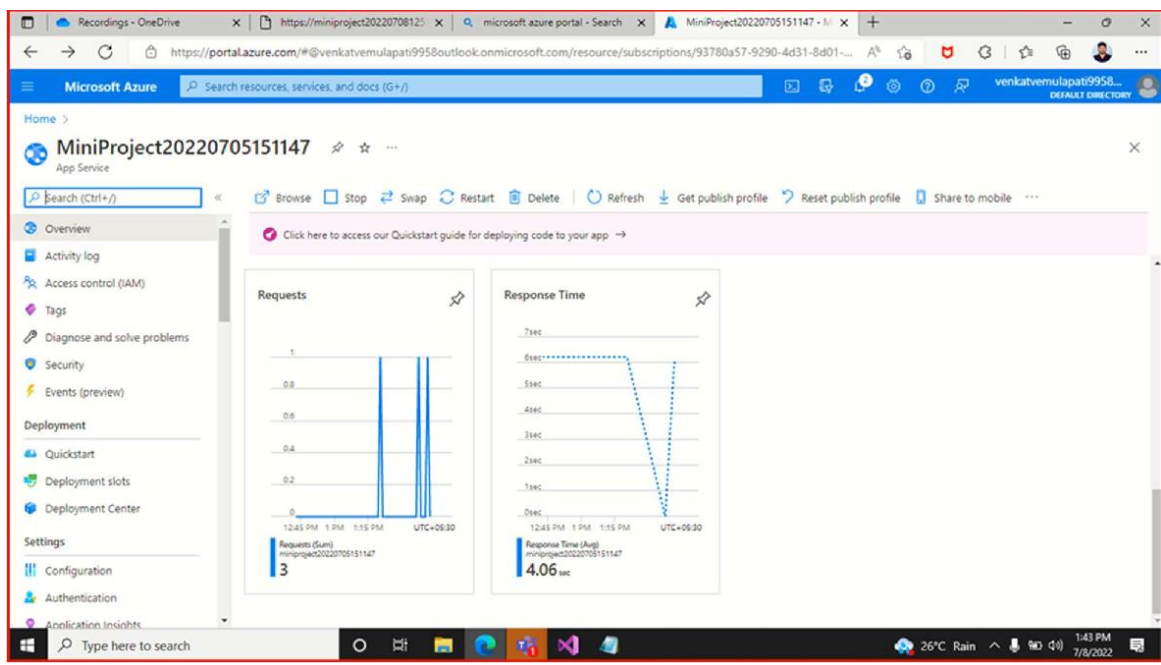
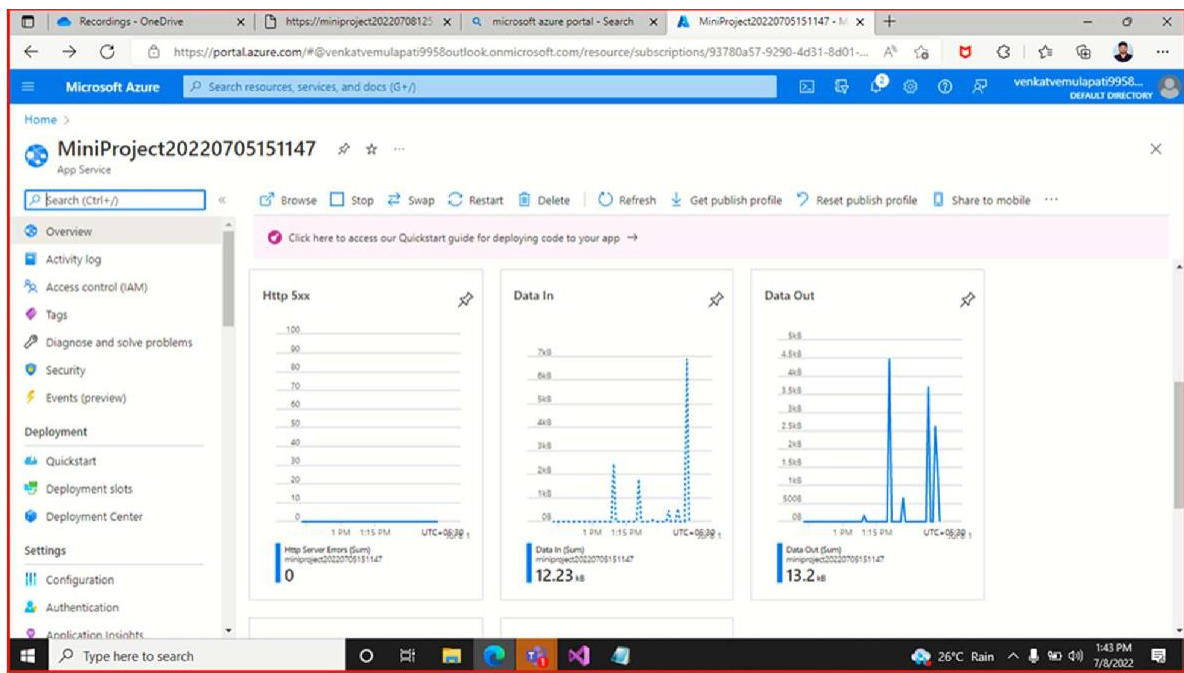
Application Insights:

- Supports a wide variety of platforms, including .NET, Node.js, Java, and Python.
- Works for apps hosted on-premises, hybrid, or on any public cloud.
- Integrates with DevOps processes.
- Has connection points to many development tools.
- Can monitor and analyze telemetry from mobile apps by integrating with Visual Studio [App Center](#).

Follow the bellow images to add Application insights

1. Trun on Application Insights.
2. Select Existing Resource.
3. Click on the Apply button.





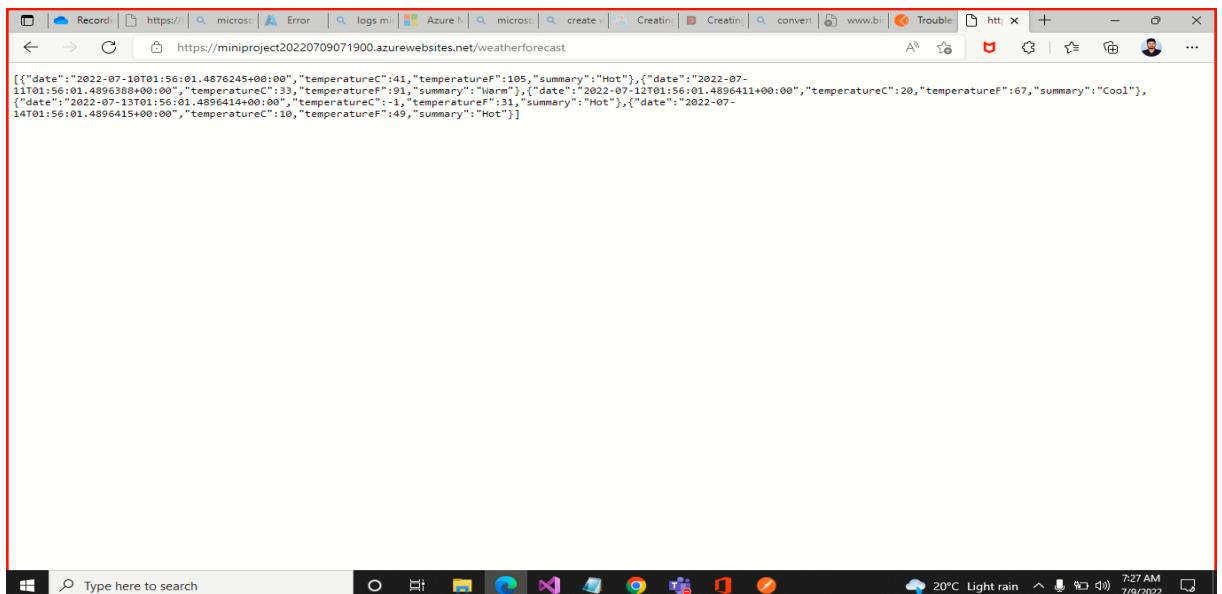
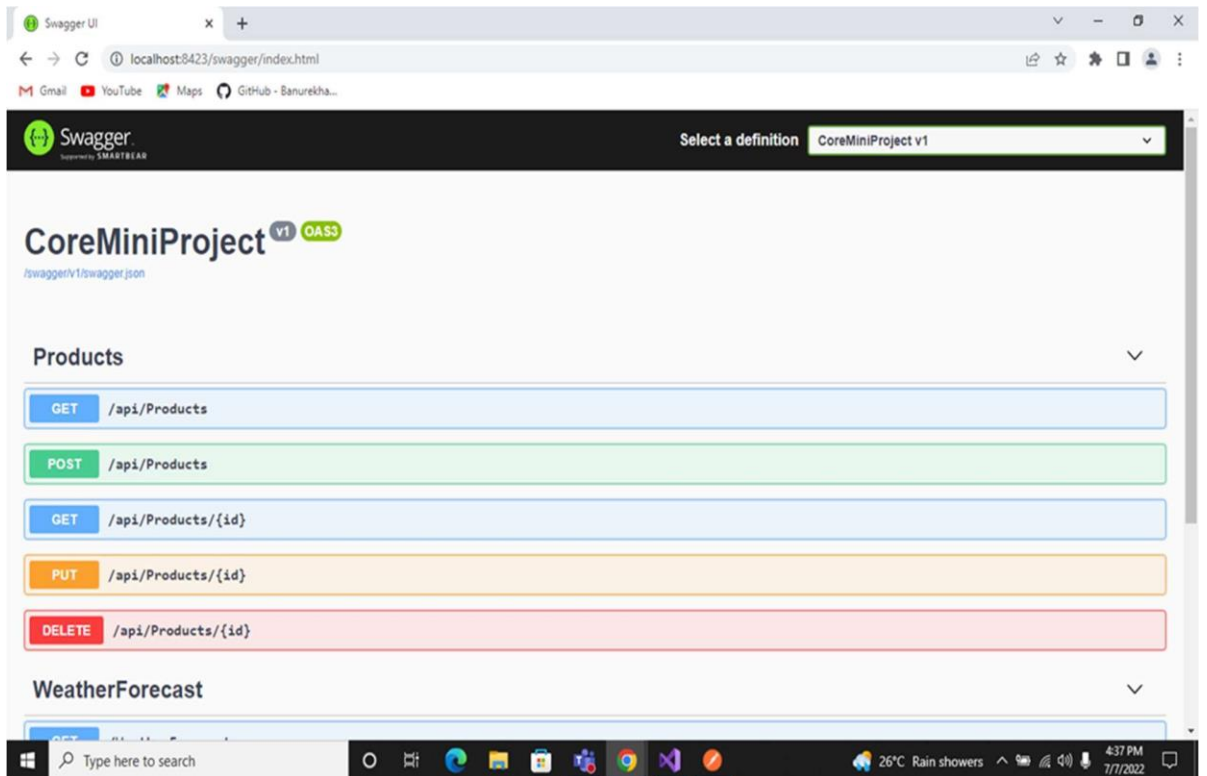
5. Configure Swagger for the Api :-

Swagger UI allows anyone — be it your development team or your end consumers — to visualize and interact with the API's resources without having any of the implementation logic in place. It's automatically generated from your OpenAPI (formerly known as Swagger) Specification, with the visual documentation making it easy for back end implementation and client side consumption.

Advantages of Swagger Api :-

1. Testing is always crucial; on both the development and QA levels. Swagger provides a UI integrated page where all the API methods are listed and enables the user to test any method that is required from the UI.
2. Swagger does the documentation in a conventional way (OpenAPI) which means it is in a machine-readable language. If a user starts the documentation first, Swagger will write the structure of the API automatically based on the written documentation. The API logic relies on the developer and business requirements but the structure will be written by Swagger itself.
3. The user does not need a separate applications to test APIs. Just configure Swagger once in the project and access it through a URL to test the APIs.

These are the major benefits users will see once they start using Swagger. Swagger provides immense support for a wide range of platforms, languages, and domains.



6. Work with Log Analytics with the sample logs available :-

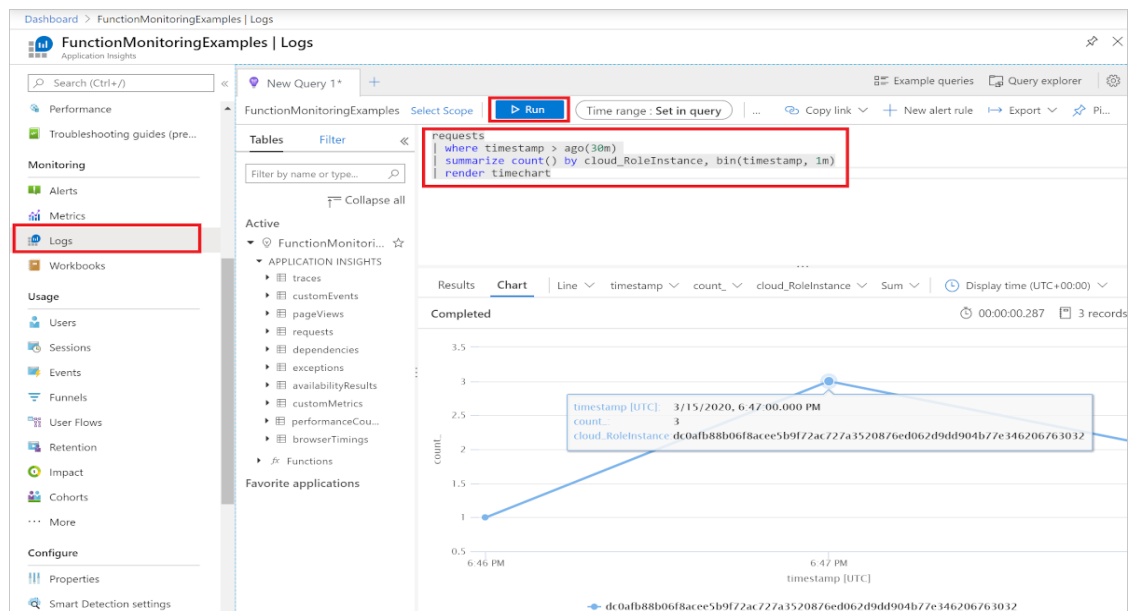
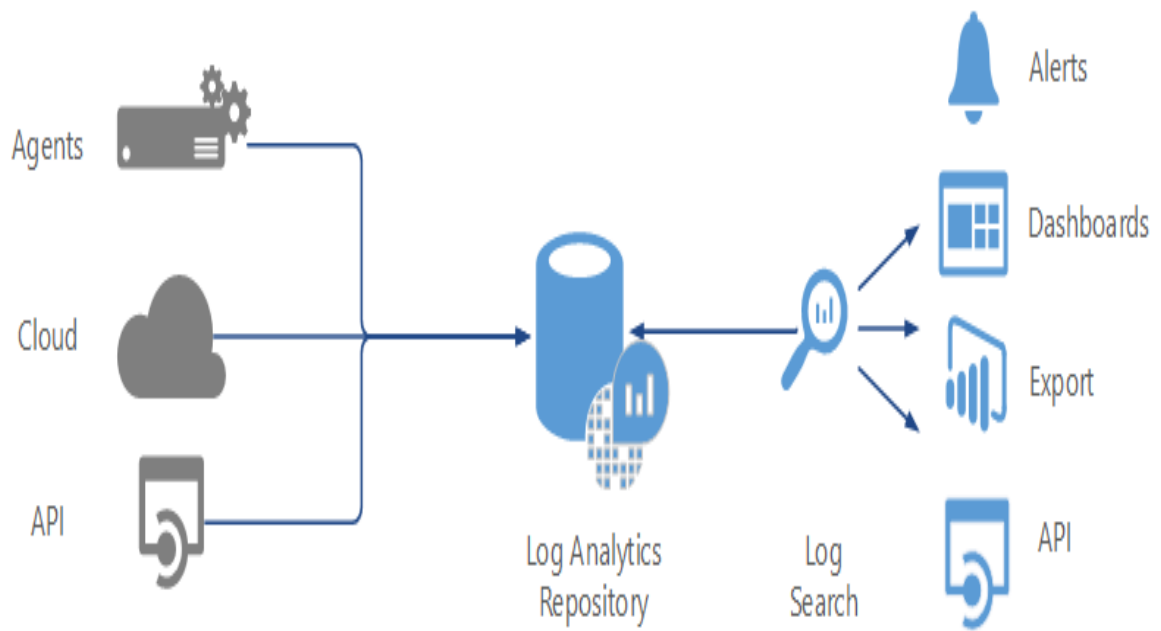
Log Analytics is a tool in the Azure portal to edit and run log queries from data collected by Azure Monitor logs and interactively analyze their results.

You can use Log Analytics queries to retrieve records that match particular criteria, identify trends, analyze patterns, and provide various insights into your data.

- You might write a simple query that returns a set of records and then use features of Log Analytics to sort, filter, and analyze them. Or you might write a more advanced query to perform statistical analysis and visualize the results in a chart to identify a particular trend.
- If you start Log Analytics from the **Azure Monitor** menu or the **Log Analytics workspaces** menu, you'll have access to all the records in a workspace. If you select **Logs** from another type of resource, your data will be limited to log data for that resource. For more information, see [Log query scope and time range in Azure Monitor Log Analytics](#).

How Logs work by using Query language as shown in bellow images.

1. Open Azure portal and click on Logs.
2. Select the Tables what you check.
3. Write a query logic and click on Run.



Home > Resource groups > demos > demosworkspace - Logs

demosworkspace - Logs

Log Analytics

Search (Ctrl+J)

Activity log
Access control (IAM)
Tags
Diagnose and solve problems

Settings
Locks
Automation script
Advanced settings

General
Quick Start
Workspace summary
View Designer

Logs
Logs (classic)
Solutions
Saved searches
Pricing tier
Usage and estimated costs
Properties
Service Map

Workspace Data Sources
Virtual machines

New Query 1*

demosworkspace

Schema Filter (preview)

Filter by name...

Showing top 10 values for each facet

- Computer (1)
 - (empty) (15000)
- Message (1)
 - lorem ipsum dolor sit amet (15000)
- SourceSystem (1)
 - RestAPI (15000)

search *
where Type == "demoLog_CL"

Time range: Last 7 days

Run Save Copy link

Completed. Showing partial results from the last 7 days.

TABLE CHART Columns

Drag a column header and drop it here to group by that column

	Stable	Criticality_s	Message	Category
>	demolog_CL	Exception	lorem ipsum dolor sit amet	
>	demolog_CL	Informational	lorem ipsum dolor sit amet	
>	demolog_CL	Warning	lorem ipsum dolor sit amet	
>	demolog_CL	Exception	lorem ipsum dolor sit amet	
>	demolog_CL	Warning	lorem ipsum dolor sit amet	
>	demolog_CL	Informational	lorem ipsum dolor sit amet	
>	demolog_CL	Exception	lorem ipsum dolor sit amet	
>	demolog_CL	Warning	lorem ipsum dolor sit amet	
>	demolog_CL	Exception	lorem ipsum dolor sit amet	
>	demolog_CL	Informational	lorem ipsum dolor sit amet	
>	demolog_CL	Warning	lorem ipsum dolor sit amet	

Apply & Run Clear

Page 1 of 200 50 items per page

Product.html

File C:/Assignments/ASP.NetCore/ASP.NETCoreProject/CoreMiniProject/CoreMiniProject/Product.html

Gmail YouTube Maps GitHub - Banurekha...

GetProducts WeatherForecast

Elements Console Sources Network Performance Memory Application

Filter

Product.html:22

```
{
  "date": "2022-07-09T12:17:18.201003+00:00",
  "summary": "Mild",
  "temperature": 28,
  "temperatureF": 82,
  "type": "Object"
}, {
  "date": "2022-07-10T12:17:18.201004+00:00",
  "summary": "Cool",
  "temperature": 24,
  "temperatureF": 75,
  "type": "Object"
}, {
  "date": "2022-07-11T12:17:18.201006+00:00",
  "summary": "Sweltering",
  "temperature": 39,
  "temperatureF": 102,
  "type": "Object"
}, {
  "date": "2022-07-12T12:17:18.201007+00:00",
  "summary": "Cool",
  "temperature": -14,
  "temperatureF": 7,
  "type": "Object"
}, {
  "date": "2022-07-13T12:17:18.201009+00:00",
  "type": "Object"
}
```

Console Issues What's New

Include third-party cookie issues

No issues detected so far

Type here to search

25°C Rain 5:48 PM 7/8/2022

Home > Resource groups > demos > demosworkspace - Logs

Search (Ctrl+J)

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Settings

Locks

Automation script

Advanced settings

General

Quick Start

Workspace summary

View Designer

Logs

Logs (classic)

Solutions

Saved searches

Pricing tier

Usage and estimated costs

Properties

Service Map

Workspace Data Sources

Virtual machines

New Query 1*

+

Run

Time range: Last 7 days

Save

Copy link

Schema

Filter (preview)

Search *
| where Type == "demolog_CL"

Filter by name...

Showing top 10 values for each facet

Computer (1)

(empty) (15000)

Message (1)

lorem ipsum dolor sit amet (15000)

SourceSystem (1)

RestAPI (15000)

Completed. Showing partial results from the last 7 days.

TABLE

CHART

Columns

Drag a column header and drop it here to group by that column

\$table	Criticality_s	Message	Category
> demolog_CL	Exception	lorem ipsum dolor sit amet	
> demolog_CL	Informational	lorem ipsum dolor sit amet	
> demolog_CL	Warning	lorem ipsum dolor sit amet	
> demolog_CL	Exception	lorem ipsum dolor sit amet	
> demolog_CL	Warning	lorem ipsum dolor sit amet	
> demolog_CL	Informational	lorem ipsum dolor sit amet	
> demolog_CL	Exception	lorem ipsum dolor sit amet	
> demolog_CL	Warning	lorem ipsum dolor sit amet	
> demolog_CL	Exception	lorem ipsum dolor sit amet	
> demolog_CL	Informational	lorem ipsum dolor sit amet	
> demolog_CL	Warning	lorem ipsum dolor sit amet	

Apply & Run

Clear

Page 1 of 200

50 items per page