

Assignment 05 Docker

step 1:

Created a console core app and added players class with property like id, name , team, role, jersey number and age.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Assignment05Docker
{
    public class Players
    {
        public int PId { get; set; }
        public string Name { get; set; }
        public string Team { get; set; }
        public string Role { get; set; }
        public int jersey_number { get; set; }
        public int Age { get; set; }
    }
}
```

Step 2:

Add new class playermanagement.cs to manage players data in a list and developed functions to access list data by id or get all players data.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Assignment05Docker
{
    public class PlayerManagement
    {
        static List<Players> players = new List<Players>() {
            new Players() { PId=1, Name="Lionel Messi", Team="Paris Saint-
Germain", Role="Forward", jersey_number=30, Age=34 },
        }
    }
}
```

```

        new Players() { PId=2, Name="Cristiano Ronaldo", Team="Manchester
United", Role="Forward", jersey_number=7, Age=37 },
        new Players() { PId=3, Name="Neymar Jr.", Team="Paris Saint-
Germain", Role="Forward", jersey_number=10, Age=30 },
        new Players() { PId=4, Name="Kevin De Bruyne", Team="Manchester
City", Role="Midfielder", jersey_number=17, Age=30 },
        new Players() { PId=5, Name="Robert Lewandowski", Team="Bayern
Munich", Role="Forward", jersey_number=9, Age=33 },
        new Players() { PId=6, Name="Virgil van
Dijk", Team="Liverpool", Role="Defender", jersey_number=4, Age=31 },
    };

    public static void GetAllPlayers()
    {
        Console.WriteLine($"{"P_ID",-5} {"Name",-20} {"Team",-20} {"Role",-15}
{"Jersey No",-10} {"Age",-5}");
        foreach (Players player in players)
        {

            Console.WriteLine($" {player.PId,-5} {player.Name,-20} {player.Team,-20}
{player.Role,-15} {player.jersey_number,-10} {player.Age,-5}");
        }
    }

    public static void GetPlayerById(int id)
    {
        Console.WriteLine($"{"P_ID",-5} {"Name",-20} {"Team",-20} {"Role",-15}
{"Jersey No",-10} {"Age",-5}");
        List<Players> p = players.FindAll(x=>x.PId == id);
        foreach(Players player in p)
        {

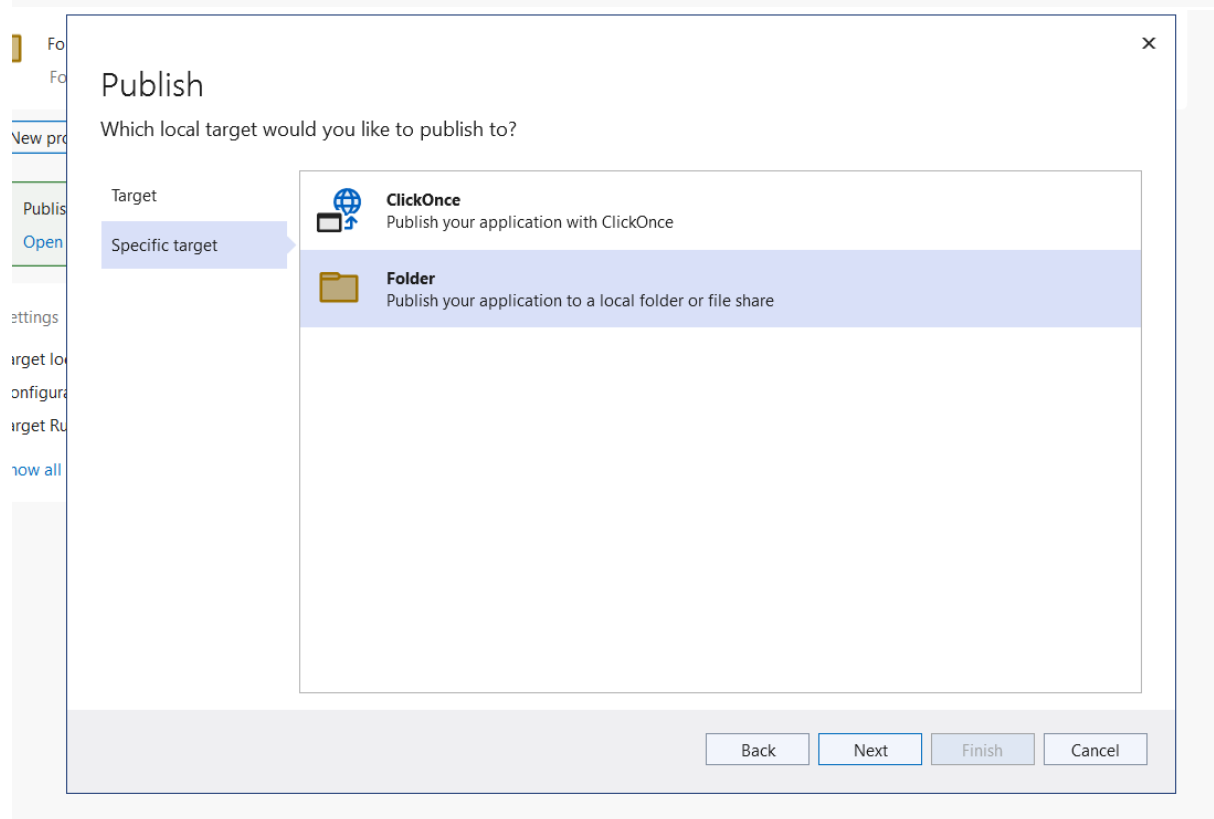
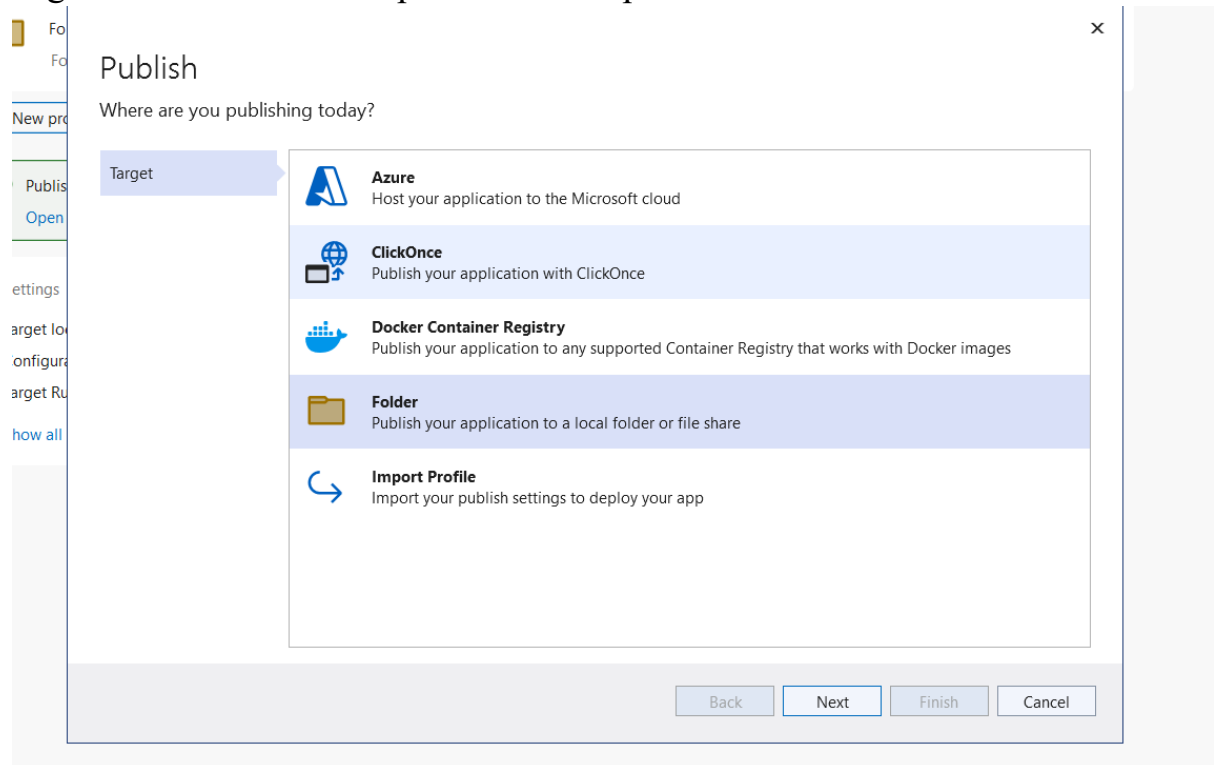
            Console.WriteLine($" {player.PId,-5} {player.Name,-20} {player.Team,-20}
{player.Role,-15} {player.jersey_number,-10} {player.Age,-5}");

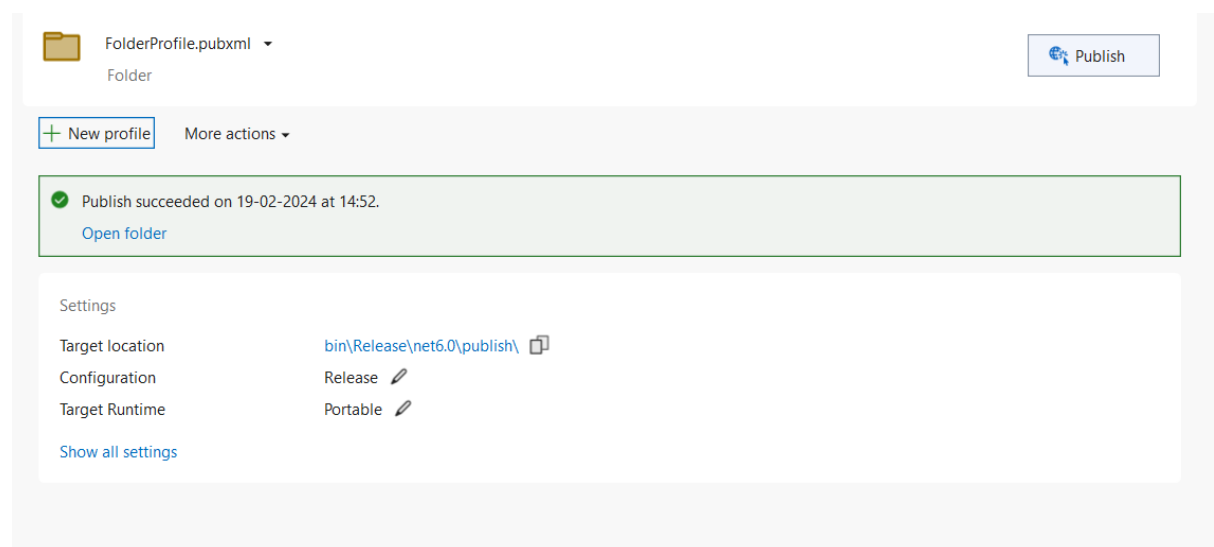
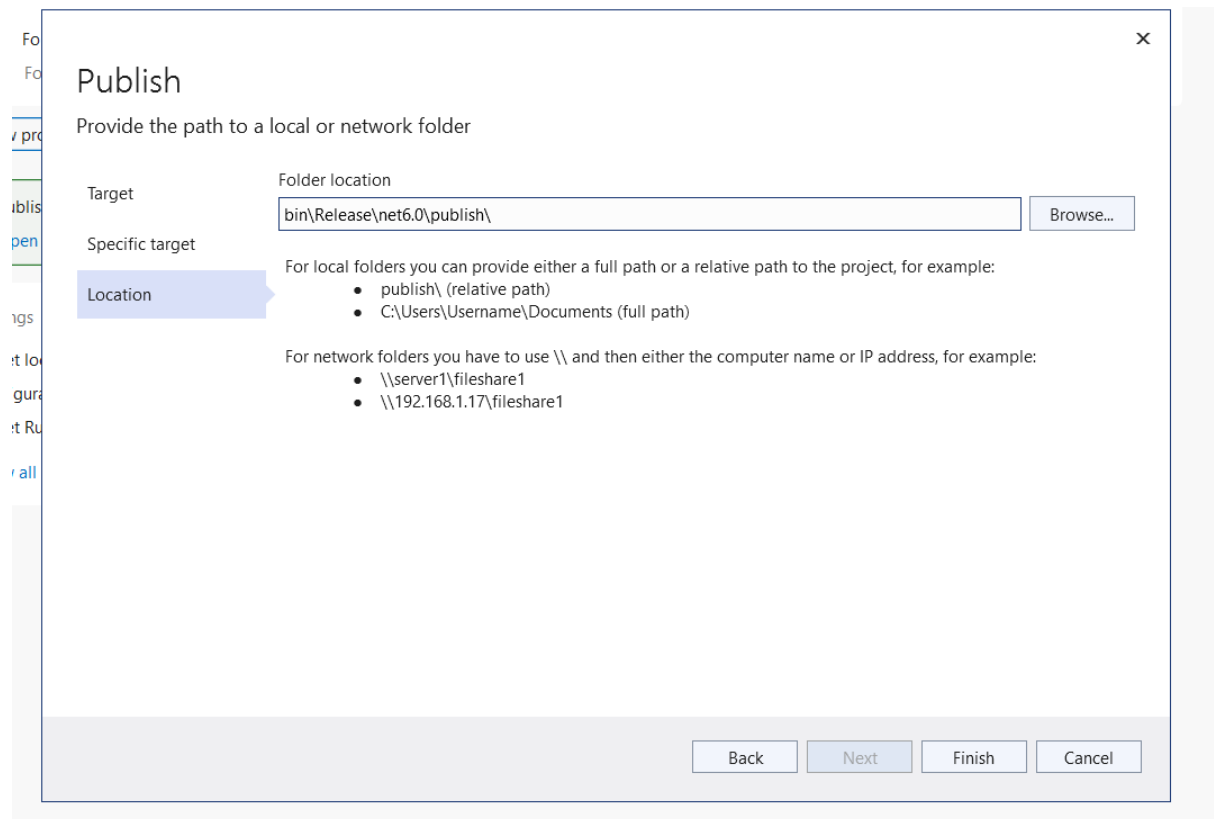
        }
    }
}

```

Step 3:

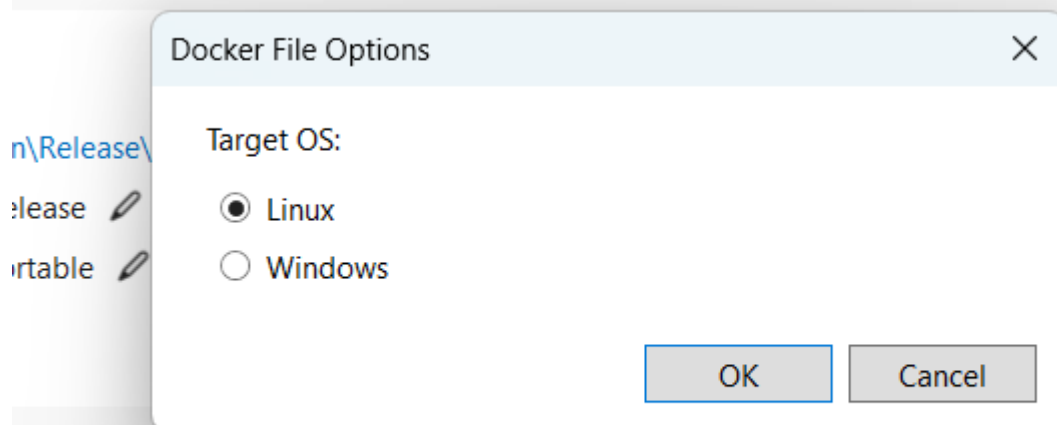
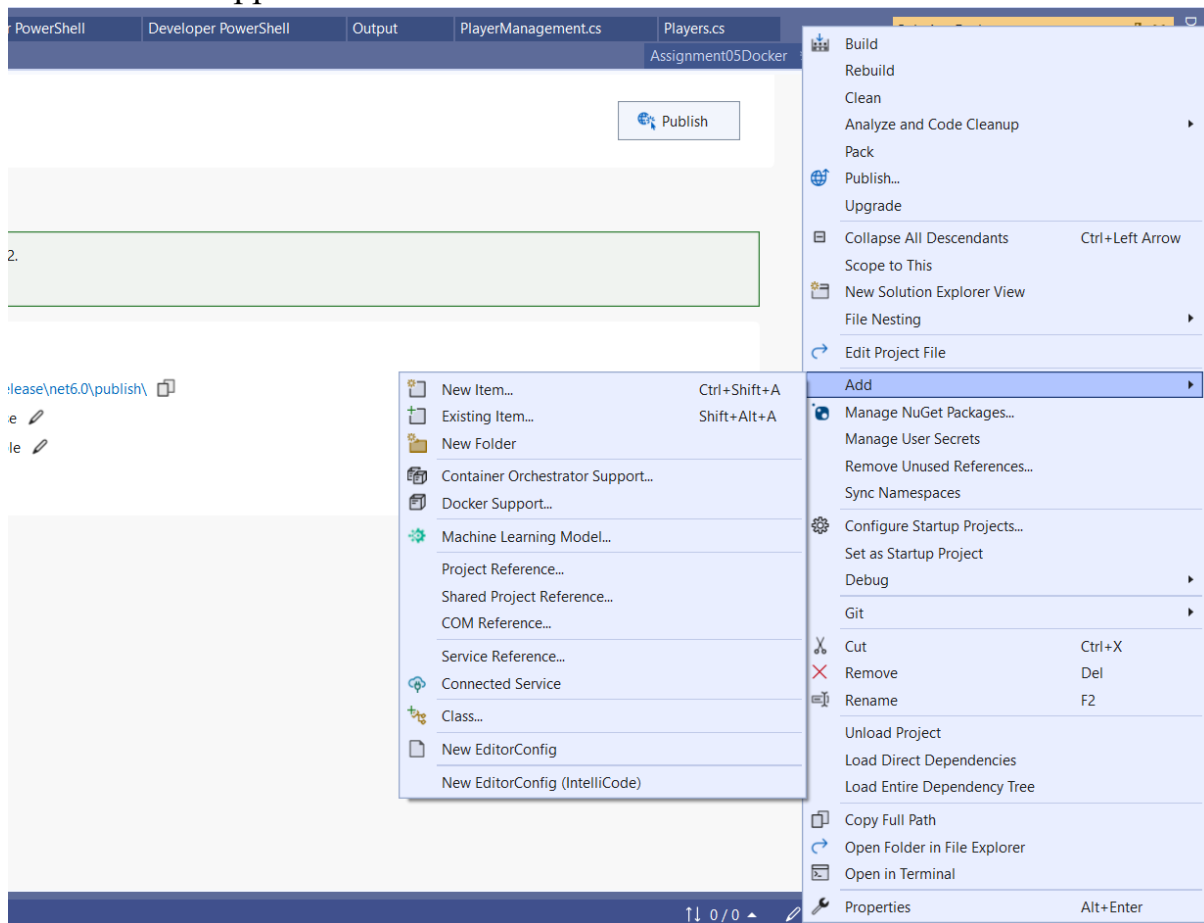
Right click on solution explorer and add publish





Step 4:

Add Docker support



Step 5

Goto terminal and run the commands

```
$ docker login
```

```
$ docker build -t {name of image} -f {path\Dockfile} .
```

```
$ docker images
```

```
$ docker run -it --rm {image name}
```

Step 6:

Push to Docker hub

```
$ docker tag {image name} {username}/{new repository name}
```

```
$ docker push {username}/{new repository name}
```

Step 7 :

Goto <https://labs.play-with-docker.com/>

create new instance

```
docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
[node3] (local) root@192.168.0.11 ~
$ docker pull naveen2312/players_manag_sys
Using default tag: latest
latest: Pulling from naveen2312/players_manag_sys
5d0aeceef7ee: Pull complete
7c2bfda75264: Pull complete
950196e58fe3: Pull complete
ecf3c05ee2f6: Pull complete
75d658cd13e4: Pull complete
4f4fb700ef54: Pull complete
d50b6dcf16aa: Pull complete
Digest: sha256:ed840213f8172ce99d68039c16ed346505a6d41ed380bfdcac1e5b9f64057d0b
Status: Downloaded newer image for naveen2312/players_manag_sys:latest
docker.io/naveen2312/players_manag_sys:latest
[node3] (local) root@192.168.0.11 ~
$ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
------------	-----	----------	---------	------

```
[node3] (local) root@192.168.0.11 ~
$ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
naveen2312/players_manag_sys	latest	fed2a88d084e	5 minutes ago	188MB

```
[node3] (local) root@192.168.0.11 ~
$ docker run -it --rm naveen2312/players_manag_sys
Welcome to Players Management System

Enter your choice :
1 GetAll Players
2 GetById
3 Exit

1
P_ID Name Team Role Jersey No Age
1 Lionel Messi Paris Saint-Germain Forward 30 34
2 Cristiano Ronaldo Manchester United Forward 7 37
3 Neymar Jr. Paris Saint-Germain Forward 10 30
4 Kevin De Bruyne Manchester City Midfielder 17 30
5 Robert Lewandowski Bayern Munich Forward 9 33
6 Virgil van Dijk Liverpool Defender 4 31
```

```

2 GetById
3 Exit

2

Enter Player Id :
5
P_ID Name Team Role Jersey No Age
5 Robert Lewandowski Bayern Munich Forward 9 33

Enter your choice :
1 GetAll Players
2 GetById
3 Exit

3
[node3] (local) root@192.168.0.11 ~

```

Step 8:

Github <https://github.com/naveen232000/Assignment05Docker>

The screenshot displays the GitHub interface for the repository 'Assignment05Docker'. The left sidebar shows the file structure, including a 'Properties' folder and various source files. The main content area shows the commit history for the 'Assignment05Docker' directory. The table below summarizes the commit data shown in the screenshot.

Name	Last commit message	Last commit date
..		
Properties	Add project files.	17 minutes ago
Assignment05Docker.csproj	Add project files.	17 minutes ago
Dockerfile	Add project files.	17 minutes ago
PlayerManagement.cs	Add project files.	17 minutes ago
Players.cs	Add project files.	17 minutes ago
Program.cs	Add project files.	17 minutes ago
dockerplaygd.bt	Add project files.	17 minutes ago