.NET Core Questions

# References

* <https://docs.microsoft.com/en-gb/>
* <https://www.nuget.org/>

# Questions 1

1. Read input value from screen and use string string turpulation (New feature in C#7) to combine into new string and return to the page as a printout.
   1. Worked
2. Run Web Application from command line.
   1. Need to go to folder with .csproj and run project with command “dotnet run”
3. Run application using watch.
   1. Watch needs to be installed by adding this snippet to the .csproj.

<ItemGroup>

<DotNetCliToolReference Include="Microsoft.DotNet.Watcher.Tools" Version="1.0"/>

</ItemGroup>

* 1. After you need to run “dotnet restore”
  2. After you can run “dotnet watch run” and code will be auto compiled and deployed after changing csharp files.

1. Checkout created code.
   1. This was done.

# Questions 2

1. Change the URL used by tour dev server. Comment it out after and change it back to random.
   1. Did activity
2. Edit the “.csproj” from within visual studio.
   1. Added Entity Framework but it dodnt work.
3. Use file server middleware in the startup project file. Allow directory browsing.
   1. Did Activity

# Question 3

1. Add a class library which contains a model for person.
   1. Did Activity
2. Use the Person class from class library and create a entity.
   1. This was done note after adding the new object to the db context you need to run “dotnet ef migrations add NameOfMigration” to add reference to new table changes. Then you need to run “dotnet ef databse update”

# Question 4

1. Create a new page called calculate ROI: ROI = (Gain on Investment - Cost of Investment)/Cost of Investment
2. The page should take two arguments : Cost of Investment and Gain from investment.
3. Use short cuts to add properties and constructors to the model.

# Question 5

1. Create a class to calculate the Factorial (!5 = 5 x 4 x 3 x 2 x 1)
   1. Calculate it using recursion.
   2. Calculate it using loop.
2. Call both methods one in multiple threads.
3. Call both methods using await.
4. Create an objects which has the answer, time taken, type of call thread or async.
5. Return list of results to web page.