Question 1- Encrypt and decrypt the user id and password in the connection string in web.config.

- We can use **ASP.NET IIS Registration Tool (aspnet_regiis.exe)** to encrypt and decrypt the connections strings.

Prerequisites:

- Microsoft Internet Information Services (IIS) installed and configured on the computer that will host the site.
- Create an empty project ASP.NET MVC5 or Web API.

Steps:

- Add a connection string in web.config.
- Encrypt the connection string.
- Decrypt the connection string.

Add Connection String in web.config

```
<connectionStrings>
  <add name="myConnection" connectionString="Data Source= (localdb)\MSSQLLocalDB;Initial Catalog=mydatabase User ID=userName;Password=password"
   providerName="System.Data.SqlClient"/>
</connectionStrings>
```

Encrypt connection string

- Open the command prompt with admin privilege.
- Change directory: **cd C:\Windows\Microsoft.NET\Framwork\v4.0.30319**
- Encrypting using **RsaProctectedConfigurationProvider.**
- Run command: **aspnet_regiis -pef "connectionStrings" "path of the project that created"** [right click on your project and click open folder in file explorer and then copy the location]
- Open web.config and check if the connection string is encrypted.

Encrypted Connection string in web.config

```
<connectionStrings configProtectionProvider="RsaProtectedConfigurationProvider">
  <EncryptedData Type="http://www.w3.org/2001/04/xmlenc#Element" xmlns="http://www.w3.org/2001/04/xmlenc#">
    <EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#tripledes-cbc" />
```

<KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">

<EncryptedKey xmlns="http://www.w3.org/2001/04/xmlenc#">

<EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#rsa-1_5" />

<KeyInfo xmlns="http://www.w3.org/2000/09/xmldsig#">

<KeyName>Rsa Key</KeyName>

</KeyInfo>

<CipherData>

<CipherValue>B4B3oZrbpQsYM7Eaq5smukqDj9XUYUCwygBYRG1iasN4ll5W4wAKVCIFCRfvOJGoIXzgqpyjAI30IKf5pnZ/xWqmo3p/wGfOKdMrzd041dt9llLGbxFpLJs0Nkm583PJ1FppXLAy7FOD0YoBVhG/PBtBgLjTQqcXRNbVcgufzuArlv/EH+7lzSNRclXSTMOPMtISF65hPI9ICj9qLx7RBGhVZ6uFZVFteyyuRd2i3D2r7wJfr6KflFkakdxp1OWE2JK4Ldb8kZSwAy3bNaI/qaV9EgIWt9wM6RZO/IrI3kI/bX8JuvirPw3j/+TLDB3MoIgKjSbLpR3GYTm9csPu8g==</CipherValue>

</CipherData>

</EncryptedKey>

</KeyInfo>

<CipherData>

<CipherValue>0n1Y6ScSNZDR4x1sXfK05w9h+pp2OrAEQFQsoAUP5Y/hPsfpJS/7jv21PbPlkYmdCzycM4PGGb0+fuffR3RuL1x0tn7rfyUdA9llTfkyRQKwS9xOmkMsVFXgQDr8P4aXGef1fZPE2gjhcjm/JQToLwsfQZK1gNr4d6cIPFNqKD6wt24F7fuySJPX3OgLb8wXfQMd7ij+JcZzNlnyNHbq/DljxSpPOnMrC52t06Jj8F8+MsSud9GcijcFB2UhvLVXQwyZ51nEj6Tf36Zbca8bgw==</CipherValue>

</CipherData>

</EncryptedData>

</connectionStrings>

Decrypt connection string

- Run command: **aspnet_regiis -pdf "connectionStrings" "path of the project that created"** [right click on your project and click open folder in file explorer and then copy the location]
- Open the web.config and check if the connection string is decrypted.

Question 2- Build a feature in Angular App that feature doesn't exist in .NET or Angular need to build it from scratch.

- Created a TodoList App with Authentication, AuthGuard , and using json-server.

GitHub Repo Link: [Arunkumar0610/TodoListApp (github.com)]

1. Clone the repo.
2. Run npm install
3. Run json-server --watch db.json --port 3000
4. Add new terminal and ng serve to run application.
5. A user can login, register, add, edit, view, and remove todo-task from todo-list.