

Practical Assignment 4

Github Link - <https://github.com/gamble4846/WebApi-PracticalAssignment4.git>

1	<p>Create a Web API to perform following tasks:</p> <ul style="list-style-type: none"> A. Convert user given input to uppercase. B. Convert user given input to lowercase. C. Convert user given input to reverse case. <ul style="list-style-type: none"> a. (For Example: AbC => aBc) D. Make first letter capital in user given input. <ul style="list-style-type: none"> a. (For Example: hello world =>Hello World)
Ans	<p>Controllers - DefaultController.cs</p> <pre> 1. using System; 2. using System.Collections.Generic; 3. using System.Linq; 4. using System.Net; 5. using System.Net.Http; 6. using System.Web.Http; 7. 8. namespace WebApplication2.Controllers 9. { 10. public class DefaultController : ApiController 11. { 12. [Route("api/Uppercase")] 13. [HttpGet] 14. public String Uppercase(String str) 15. { 16. return str.ToUpper(); 17. } 18. 19. [Route("api/Lowercase")] 20. [HttpGet] 21. public String Lowercase(String str) 22. { 23. return str.ToLower(); 24. } 25. 26. 27. [Route("api/Reversecase")] 28. [HttpGet] 29. public String Reversecase(String str) 30. { 31. String ReversedString = new string(str.Select(c => 32. char.IsLetter(c) ? (char.IsUpper(c) ? char.ToLower(c) : 33. char.ToUpper(c)) : c).ToArray()); 34. return ReversedString; 35. } 36. } 37. }</pre>

```
34.
35.     [Route("api/Capitalize")]
36.     [HttpGet]
37.     public String capitalize(String str)
38.     {
39.         bool flag = false;
40.         String CapitalizedString = "";
41.
42.         for(int i=0;i<str.Length;i++)
43.         {
44.             if (str[i].Equals(' ') || i == 0)
45.             {
46.                 flag = true;
47.                 if(str[i].Equals(' '))
48.                 {
49.                     CapitalizedString += ' ';
50.                     i++;
51.                 }
52.             }
53.             else
54.             {
55.                 flag = false;
56.             }
57.             CapitalizedString += (flag) ?
str[i].ToString().ToUpper() : str[i].ToString();
58.         }
59.
60.         return CapitalizedString;
61.     }
62. }
63. }
```

2	<p>Create a Web API to manage data for Movie List:</p> <ul style="list-style-type: none"> A. Create a class Movie (movie_id, movie_name, movie_cast) B. Create a list to store movie data C. Create web API actions to perform add, update, delete, view data operation into the movie list.
Ans	<p>Models - Movie.cs</p> <pre> 1. using System; 2. using System.Collections.Generic; 3. using System.Linq; 4. using System.Web; 5. 6. namespace WebApplication1.Models 7. { 8. public class Movie 9. { 10. // (movie_id, movie_name, movie_cast) 11. public int movie_id { get; set; } 12. public String movie_name { get; set; } 13. public String movie_cast { get; set; } 14. } 15. } </pre> <p>Controllers - DefaultController.cs</p> <pre> 1. using System; 2. using System.Collections.Generic; 3. using System.Linq; 4. using System.Net; 5. using System.Net.Http; 6. using System.Web.Http; 7. using WebApplication1.Models; 8. 9. namespace WebApplication1.Controllers 10. { 11. public class DefaultController : ApiController 12. { 13. static List<Movie> movies = new List<Movie>(); 14. 15. [Route("api/addMovie")] 16. [HttpPost] 17. public String addMovie(String movie_name, String movie_cast) 18. { 19. Movie obj = new Movie(); 20. obj.movie_name = movie_name; 21. obj.movie_cast = movie_cast; </pre>

```

22.         obj.movie_id = movies.Count == 0 ? 0 :
(movies[movies.Count - 1].movie_id + 1);
23.         movies.Add(obj);
24.         return "Movie Added";
25.     }
26.
27.     [Route("api/viewMovies")]
28.     [HttpGet]
29.     public List<Movie> viewMovies()
30.     {
31.         return movies;
32.     }
33.
34.     [Route("api/deleteMovie")]
35.     [HttpPost]
36.     public String deleteMovie(int movie_id)
37.     {
38.         String Movie_Name = "";
39.         String returnstring;
40.         bool flag = false;
41.
42.         for (int i = 0; i < movies.Count; i++)
43.         {
44.             if (movies[i].movie_id == movie_id)
45.             {
46.                 Movie_Name = movies[i].movie_name;
47.                 movies.RemoveAt(i);
48.                 flag = true;
49.             }
50.         }
51.
52.         returnstring = (flag) ? Movie_Name + " Is Deleted" :
"Movie Not Found";
53.
54.         return returnstring;
55.     }
56.
57.     [Route("api/updateMovie")]
58.     [HttpPost]
59.     public String updateMovie(int movie_id, String movie_name,
String movie_cast)
60.     {
61.         String Movie_Name = "";
62.         String returnstring;
63.         bool flag = false;
64.
65.         for (int i = 0; i < movies.Count; i++)
66.         {
67.             if (movies[i].movie_id == movie_id)
68.             {
69.                 Movie_Name = movies[i].movie_name = movie_name;
70.                 movies[i].movie_cast = movie_cast;
71.                 flag = true;

```

```
72.         }  
73.     }  
74.  
75.     returnstring = (flag) ? Movie_Name + " Is Updated" :  
    "Movie Not Found";  
76.     return returnstring;  
77. }  
78. }  
79. }
```

3	<p>Create a Web API to manage data for Product List:</p> <ol style="list-style-type: none"> Create a class Movie (Product_id, Product_name, Product_description) Create a list to store Product data Create web API actions to perform add, update, delete, view data operation into the movie list. Use concept of Route Prefix and Route Parameter For view specific data (based on id), use the concept of [FromBody].
Ans	<p>Models - Product.cs</p> <pre> 1. using System; 2. using System.Collections.Generic; 3. using System.Linq; 4. using System.Web; 5. 6. namespace WebApplication1.Models 7. { 8. public class Product 9. { 10. //(Product_id, Product_name, Product_description) 11. public int Product_id { get; set; } 12. public String Product_name { get; set; } 13. public String Product_description { get; set; } 14. } 15. } </pre> <p>Controllers - DefaultController.cs</p> <pre> 1. using System; 2. using System.Collections.Generic; 3. using System.Linq; 4. using System.Net; 5. using System.Net.Http; 6. using System.Web.Http; 7. using WebApplication1.Models; 8. 9. namespace WebApplication1.Controllers 10. { 11. [RoutePrefix("api/products")] 12. public class DefaultController : ApiController 13. { 14. static List<Product> Products = new List<Product>(); 15. 16. [Route("Add")] 17. [HttpPost] 18. public String addProduct(String Product_name, String Product_description) </pre>

```

19.         {
20.             Product obj = new Product();
21.             obj.Product_name = Product_name;
22.             obj.Product_description = Product_description;
23.             obj.Product_id = Products.Count == 0 ? 0 :
(Product[Products.Count - 1].Product_id + 1);
24.             Products.Add(obj);
25.             return "Product Added";
26.         }
27.
28.         [Route("View")]
29.         [HttpGet]
30.         public List<Product> viewProducts()
31.         {
32.             return Products;
33.         }
34.
35.         [Route("Delete/{Product_ID}")]
36.         [HttpPost]
37.         public String deleteProduct(int Product_id)
38.         {
39.             String Product_Name = "";
40.             String returnstring;
41.             bool flag = false;
42.
43.             for (int i = 0; i < Products.Count; i++)
44.             {
45.                 if (Products[i].Product_id == Product_id)
46.                 {
47.                     Product_Name = Products[i].Product_name;
48.                     Products.RemoveAt(i);
49.                     flag = true;
50.                 }
51.             }
52.
53.             returnstring = (flag) ? Product_Name + " Is Deleted" :
"Product Not Found";
54.
55.             return returnstring;
56.         }
57.
58.         [Route("Update")]
59.         [HttpPost]
60.         public String updateMovie(int Product_id, String Product_name,
String Product_description)
61.         {
62.             String Product_Name = "";
63.             String returnstring;
64.             bool flag = false;
65.
66.             for (int i = 0; i < Products.Count; i++)
67.             {
68.                 if (Products[i].Product_id == Product_id)

```

```
69.         {
70.             Product_Name = Products[i].Product_name =
Product_name;
71.             Products[i].Product_description =
Product_description;
72.             flag = true;
73.         }
74.     }
75.
76.     returnstring = (flag) ? Product_Name + " Is Updated" :
"Movie Not Found";
77.     return returnstring;
78. }
79. }
80. }
```