Bahria University

Karachi Campus

A picture containing text, emblem, symbol, crown

Description automatically generated

LAB EXPERIMENT NO.

08

LIST OF TASKS

|  |  |
| --- | --- |
| **TASK NO** | **OBJECTIVE** |
| 1 | You are working on a project where you get data from third party webapi/service in XML format. You are creating charts from that data for creating a chart you need data in JSON format in order to make them compatiable you need to convert xml data into json but limitation is you can’t modify third party webservice/api you have to make Adapter that returns data in JSON by using adapter pattern. |

Submitted On:

**1-06-2023**

(Date: DD/MM/YY)

Q: You are working on a project where you get data from third party webapi/service in XML format. You are creating charts from that data for creating a chart you need data in JSON format in order to make them compatiable you need to convert xml data into json but limitation is you can’t modify third party webservice/api you have to make Adapter that returns data in JSON by using adapter pattern.

**MAIN CLASS**

using LAB07;

XMLAdaptee adapte = new XMLAdaptee();

ITarget target = new JsonAdapter(adapte);

Console.WriteLine("Adaptee interface is incompatible with the client");

Console.WriteLine("So by using adapter client can call its method");

Console.WriteLine("\nAfter using Adapter Design Pattern, we get the requested data below");

Console.WriteLine(target.GetRequest());

Console.WriteLine("So we get the converted data..\n");

**INTERFACE ITargetS**

namespace LAB07{

internal interface ITarget{

string GetRequest();}}

**CLASS XMLAdaptee**

namespace LAB07{

internal class XMLAdaptee{

public string GetSpecificRequest(){

return "We have to convert the XML data into JSON";}}}

**CLASS JsonAdapter**

namespace LAB07{

internal class JsonAdapter : ITarget{

private readonly XMLAdaptee \_adaptee;

public JsonAdapter(XMLAdaptee \_adaptee){

this.\_adaptee = \_adaptee;}

public string GetRequest(){

return this.\_adaptee.GetSpecificRequest();}}}

**OUTPUT**

