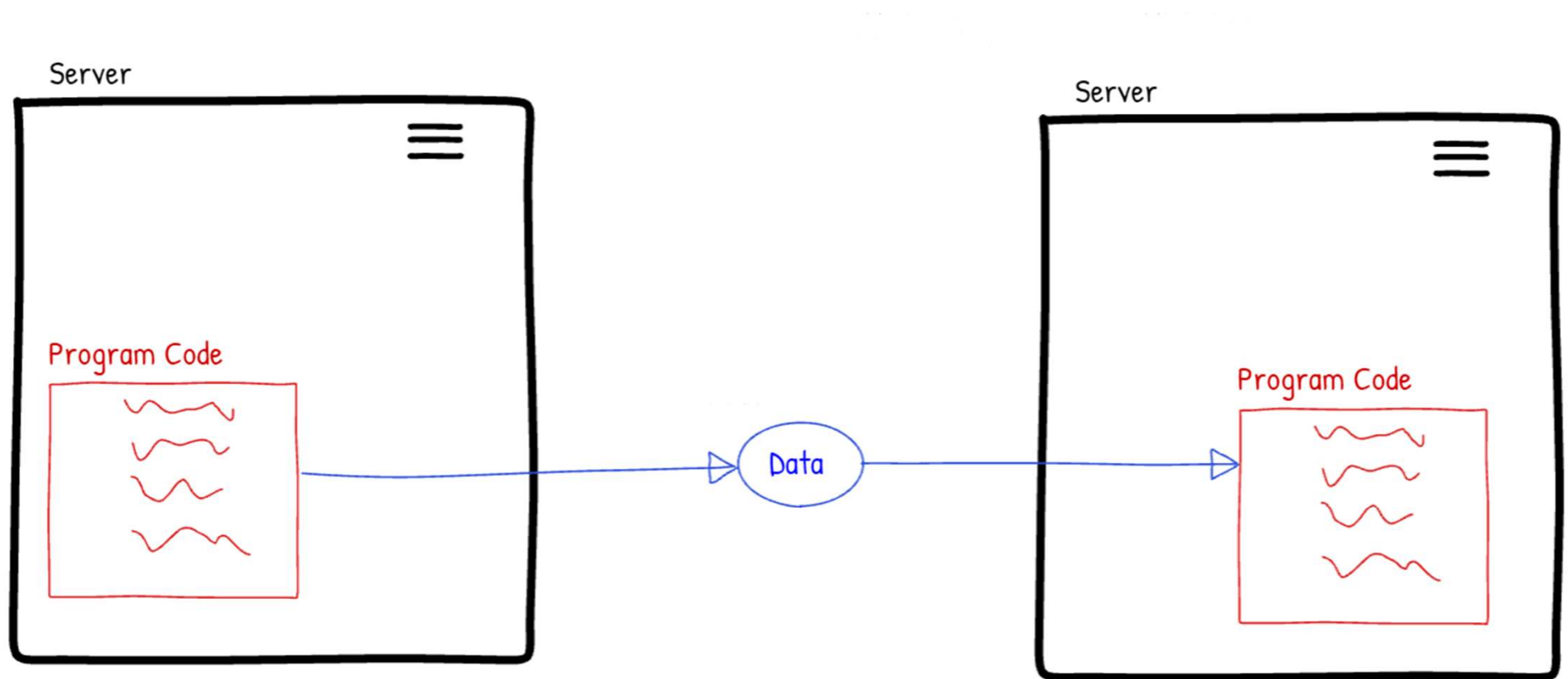


REST

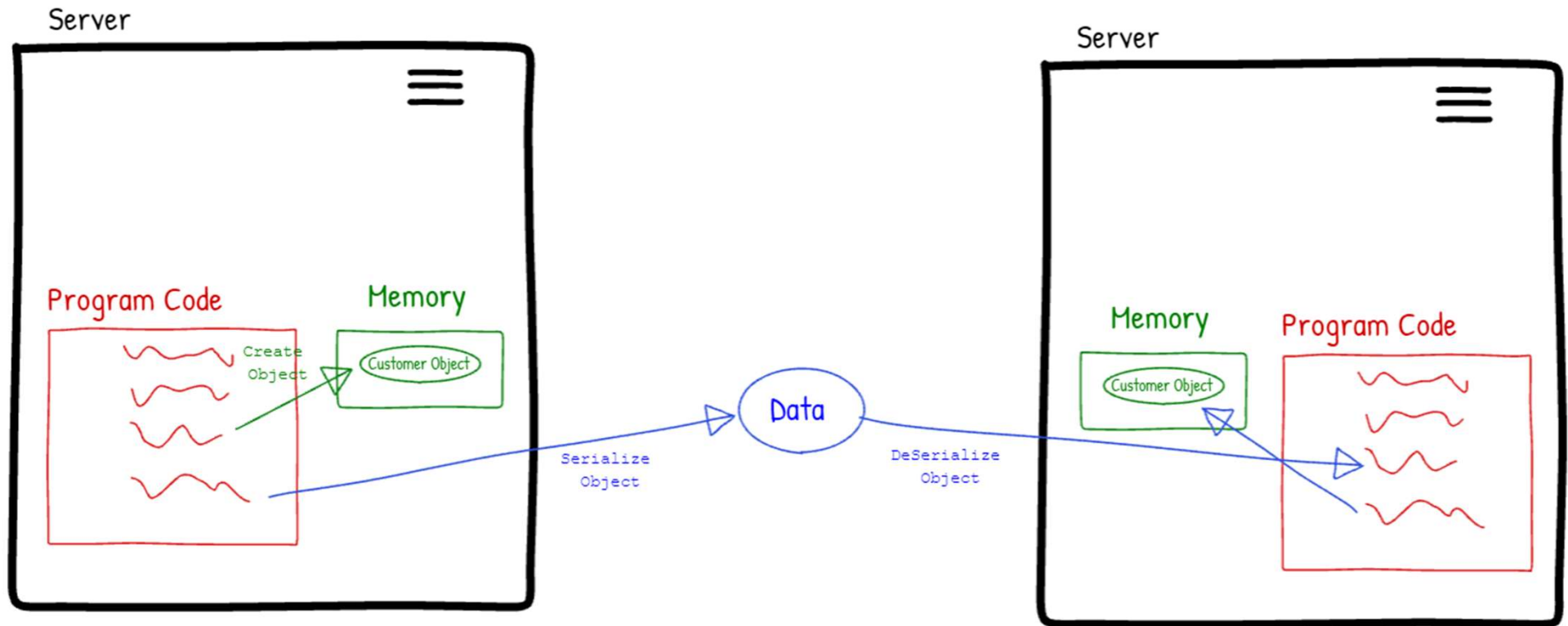
Representational State Transfer

by Mark Walsh (AIICT)

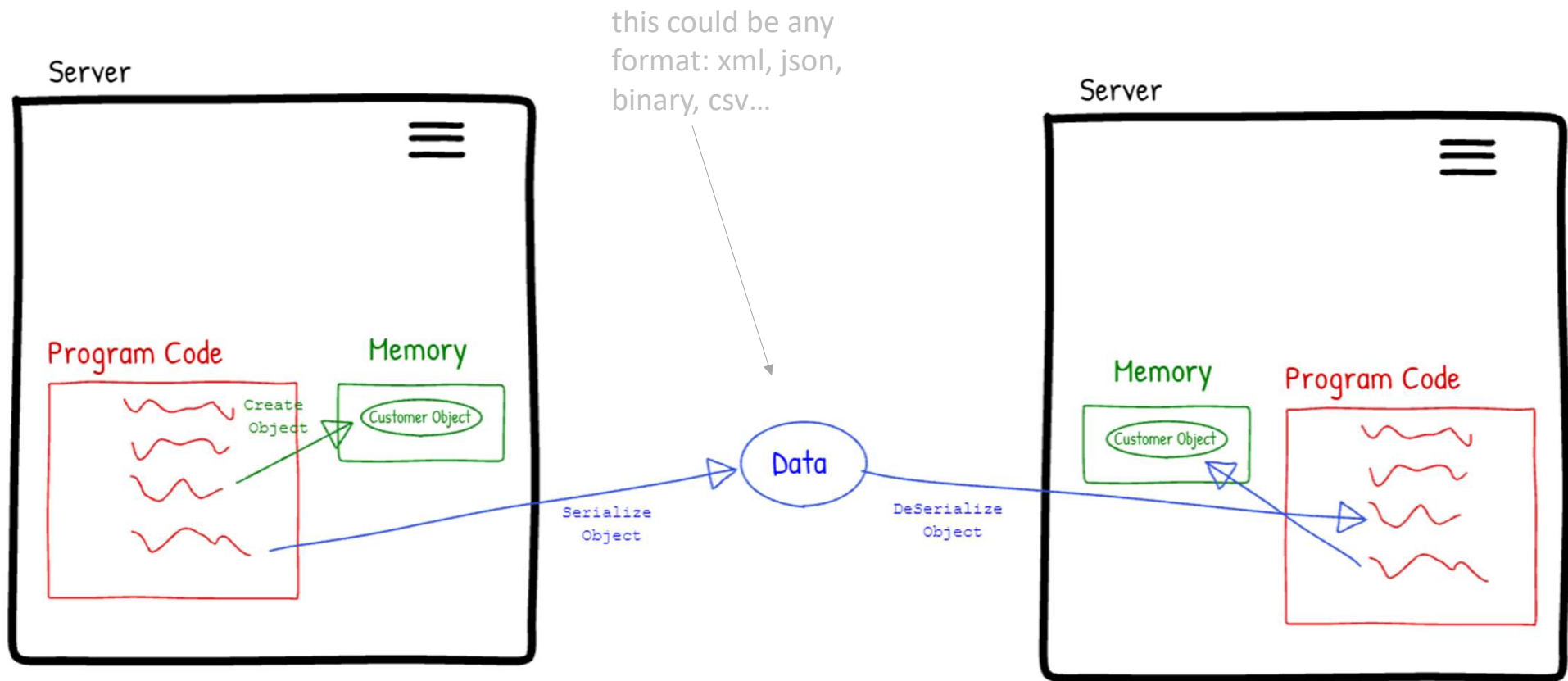
How does a program send an object to another program?



How does a program send an object to another program?



How does a program send an object to another program?



What is XML?

XML = eXtensible Markup Language

XML is not a language

Languages (e.g. Chemical Markup Language) use an XML syntax

XML is typically used to hold structured data

XML has a few key concepts:

- Elements
- Attributes
- Content

What is JSON?

JSON = JavaScript Object Notation

JSON is typically used to hold structured data

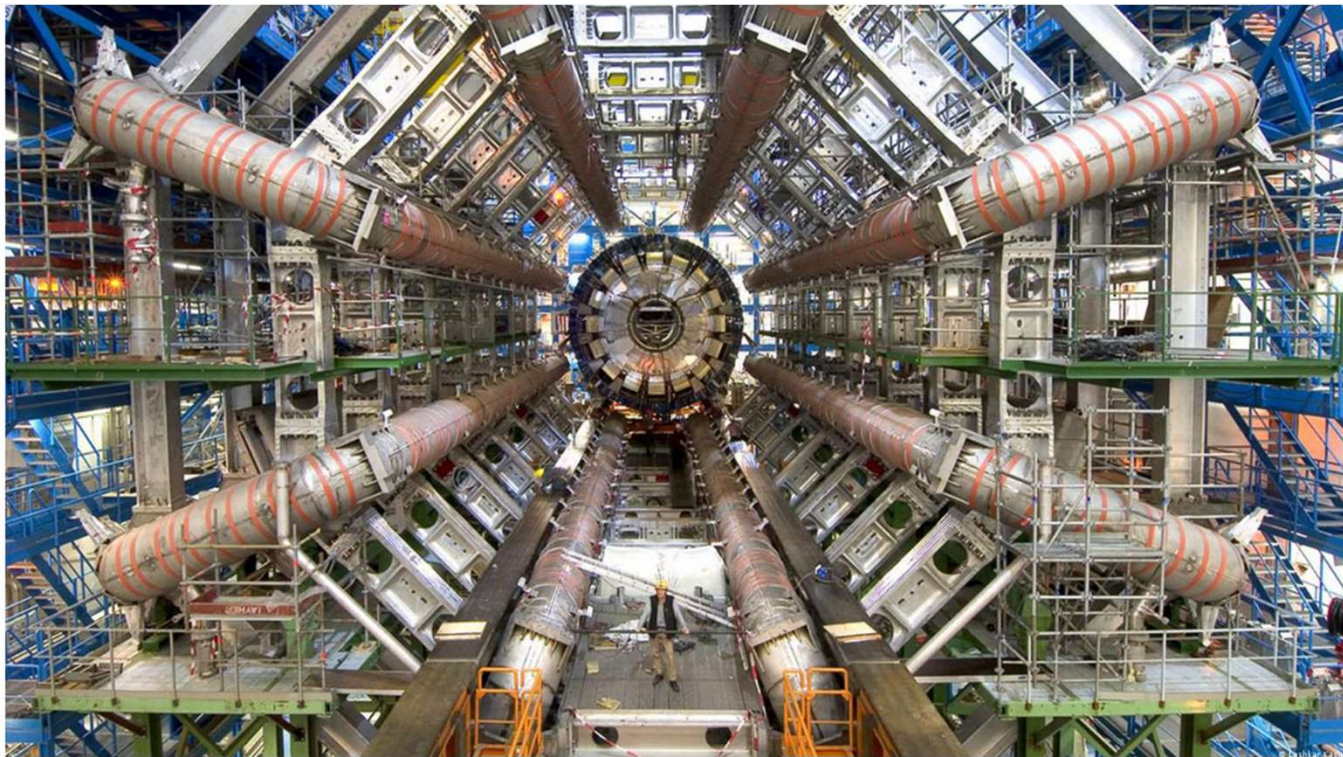
JSON has a few key concepts:

- Key-Value pairs
- DataTypes
 - strings
 - numbers
 - objects
 - arrays
 - Booleans (true or false)
 - null
- Nested Objects
- Nested Arrays

What is JSON?

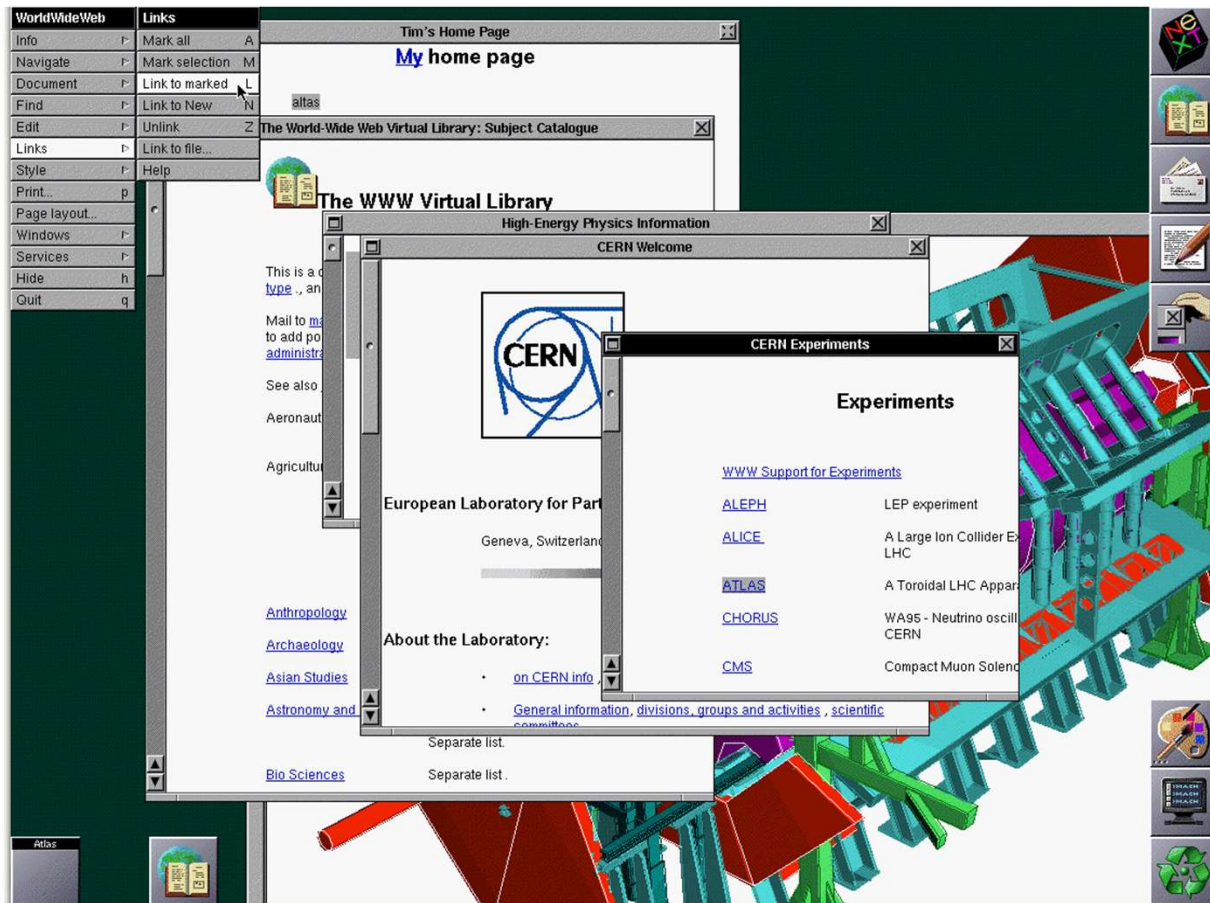
```
{
  "first_name" : "Hunstman",
  "last_name" : "Spider",
  "location" : "In Your Car",
  "websites" : [
    {
      "description" : "work",
      "URL" : "https://aiict.edu.au/"
    },
    {
      "description" : "tutorials",
      "URL" : "https://www.aiict.edu.au/community/tutorials"
    }
  ],
  "social_media" : [
    {
      "description" : "twitter",
      "link" : "https://twitter.com/aiict"
    },
    {
      "description" : "facebook",
      "link" : "https://www.facebook.com/aiictCloudHosting"
    },
    {
      "description" : "github",
      "link" : "https://github.com/aiict"
    }
  ]
}
```

What is 'the Internet'? What is 'the Web'?



The large Hadron Collider,
CERN

What is 'the Internet'? What is 'the Web'?

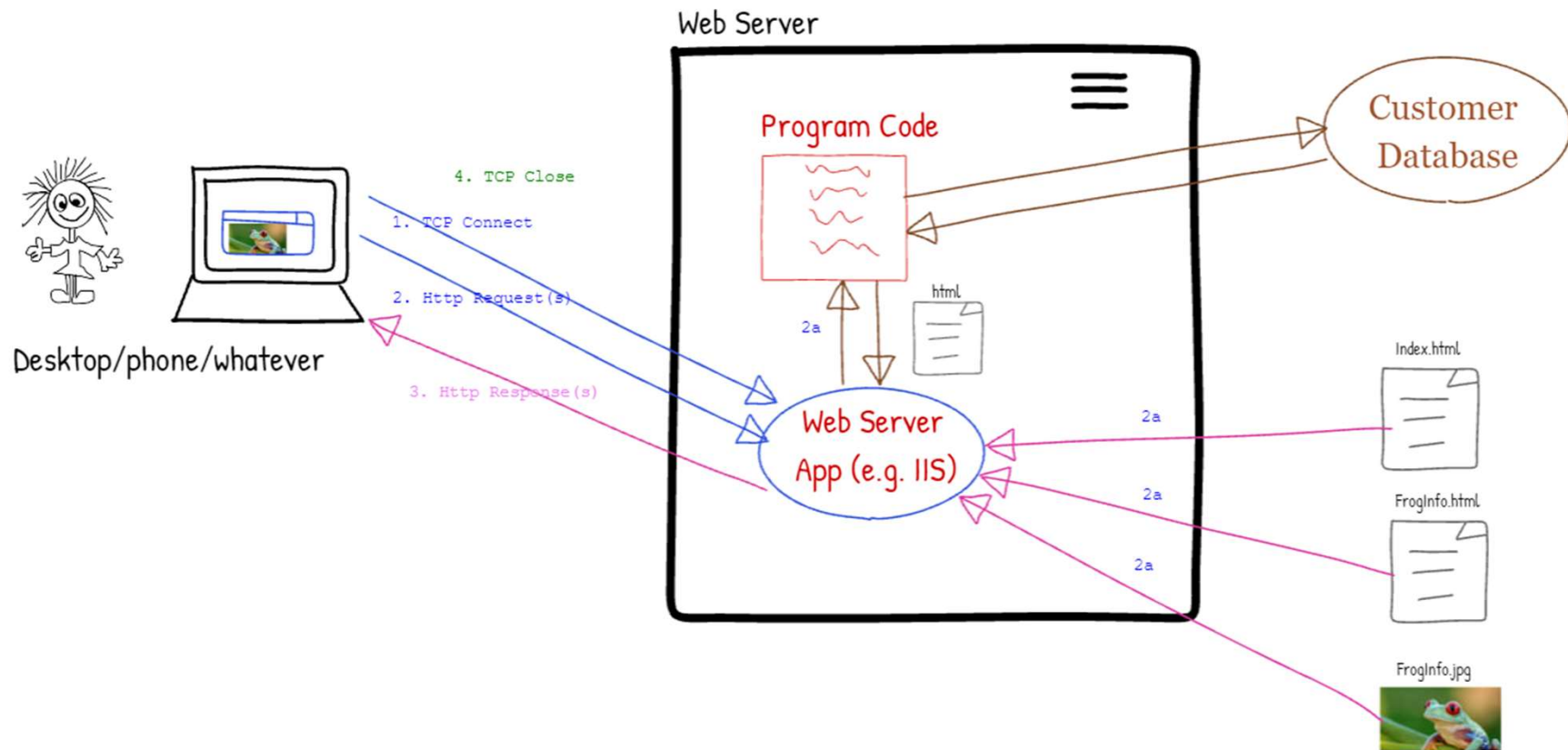


1990:

A "hypertext project" called "WorldWideWeb" in which a "web" of "hypertext documents" could be viewed by "browsers"

← The first 'browser'

Key elements of the 'the Web'



What is an API?

API = Application Programmer Interface

Not just seen in Web Applications

Is for *Application Programmers* rather than *Users*

What is REST?

REST is not a protocol

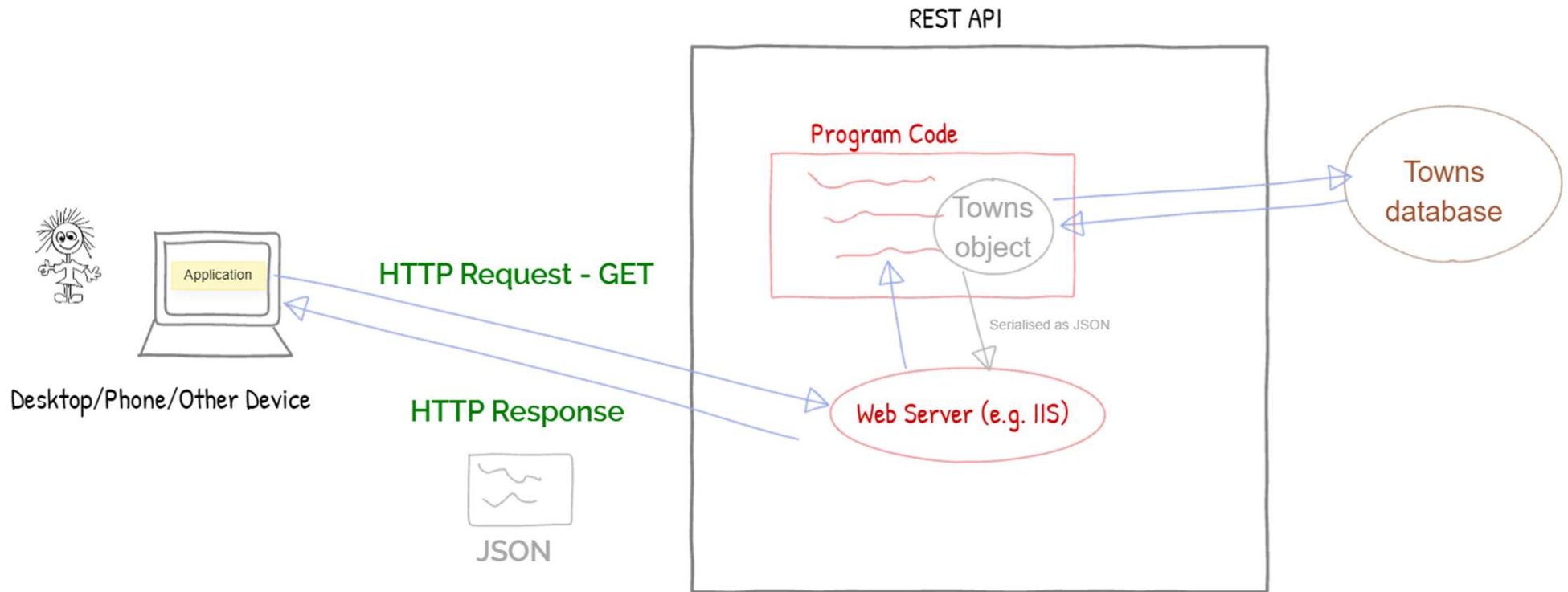
REST is not a standard

REST is not a new idea

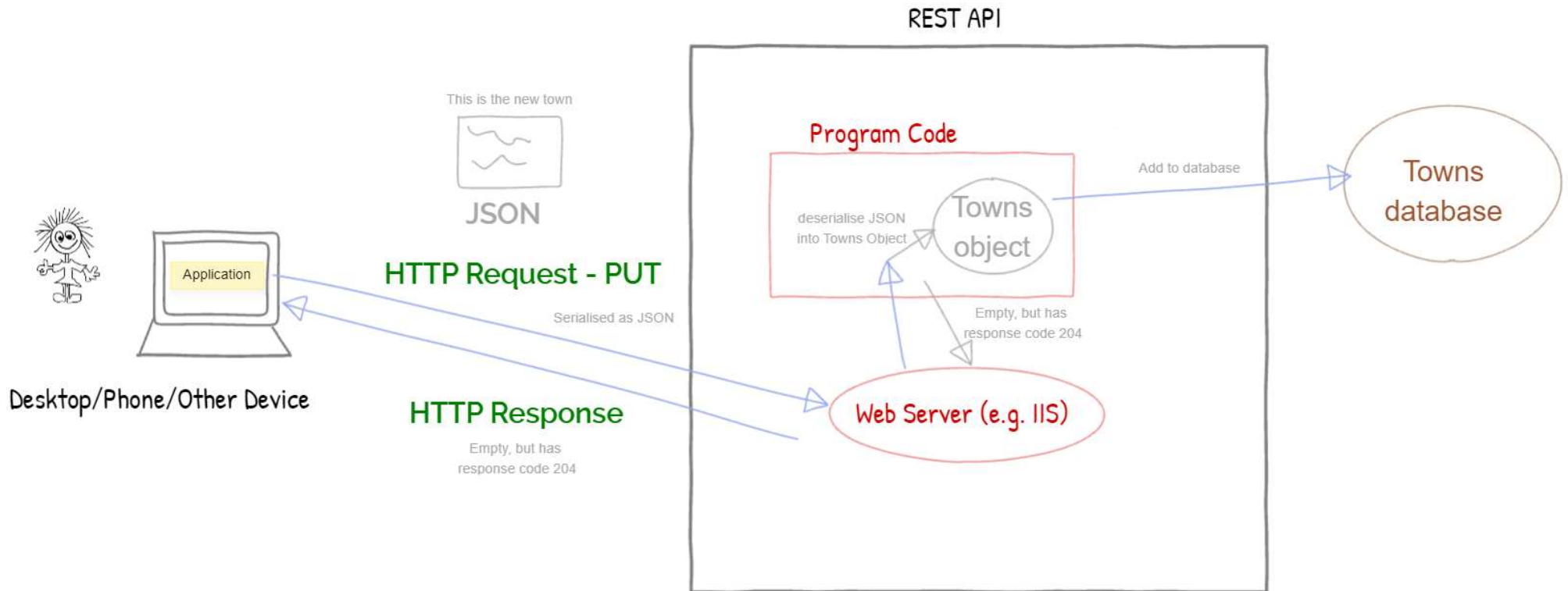
REST is not a formula

REST is a set of semantics, REST is a style

What is REST?



REST PUT Example



What is REST?

[What is REST? | Codecademy](#)

What is Swagger?

Run API, Show swagger

[API Documentation & Design Tools for Teams | Swagger](#)

Using Telerik Fiddler to Capture HTTP traffic

The screenshot displays the Telerik Fiddler interface with an intercepted HTTP request. The request details are as follows:

Host	URL
localhost:50532	/api/Values

Tunnel to: v10.vortex-win.data.m...

PUT <http://localhost:50532/api/Values> HTTP/1.1

Host: localhost:50532
Connection: keep-alive
Content-Length: 49
sec-ch-ua: ".Not/A)Brand";v="99", "Google Chrome";v="103", "Chromium";v="103"
sec-ch-ua-mobile: ?0
User-Agent: Mozilla/5.0 (Windows NT 10.0; win64; x64) AppleWebKit/537.36 (KHTML, like
sec-ch-ua-platform: "windows"
Content-Type: application/json
Accept: */*
Origin: <http://localhost:50532>
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: cors
Sec-Fetch-Dest: empty
Referer: <http://localhost:50532/swagger/ui/index>
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9

{
 "PostCode": 3000,
 "TownName": "Melbourne"
}

Annotations in the image:

- http verb**: Points to the **PUT** method.
- http headers**: Points to the header section.
- URI**: Points to the <http://localhost:50532/api/Values> URL.
- blank line**: Points to the empty line separating the headers from the body.
- payload**: Points to the JSON body content.

How do I call a REST API?

The older way

```
HttpWebRequest req = (HttpWebRequest)WebRequest.Create("uri goes here");  
Stream responseStream;  
  
req.Method = "GET";  
  
HttpWebResponse res = (HttpWebResponse)req.GetResponse();  
responseStream = res.GetResponseStream();  
  
StreamReader sr = new StreamReader(responseStream, Encoding.ASCII);  
  
string s = sr.ReadToEnd();
```

How do I call a REST API?

"casting" (i.e. converting), the result type of the create method to the `HttpRequest` data type

What is that? `DataType` (probably a class)

The older way

variable

Method (static)

```
HttpRequest req = (HttpRequest)WebRequest.Create("uri goes here");  
Stream responseStream;
```

Data Type

```
req.Method = "GET";
```

HTTP Verb

data type (probably a class)

argument (string literal)

```
HttpWebResponse res = (HttpWebResponse)req.GetResponse();  
responseStream = res.GetResponseStream();
```

— makes the call to the web server

— This is the response that came back

```
StreamReader sr = new StreamReader(responseStream, Encoding.ASCII);
```

— This is used to read the response

```
string s = sr.ReadToEnd();
```

contains response as a string

variable that will contain the stream of data coming from the web server

Let's Try it

Call Book API (the old way)

[https://www.googleapis.com/books/v1/volumes?q=isbn:*isbn-goes-here*](https://www.googleapis.com/books/v1/volumes?q=isbn:<i>isbn-goes-here</i>)

How do I call a REST API?

The newer way

```
static void Main(string[] args)
{
    Action task = new Action(RestCalls);
    Task.Run(task).Wait();
}

static async void RestCalls()
{
    HttpClient client = new HttpClient();

    Console.WriteLine("Retrieving....\n\n");
    var response = await client.GetAsync("uri goes here");
    var s = await response.Content.ReadAsStringAsync();

    Console.WriteLine($"s={s}");
}
}
```

Let's Try it

Call Book API (the new way)

[https://www.googleapis.com/books/v1/volumes?q=isbn:*isbn-goes-here*](https://www.googleapis.com/books/v1/volumes?q=isbn:<i>isbn-goes-here</i>)

Let's Try it

Draw picture of Towns & Postcodes app

Let's Try it, HTTP Get

Run API, Write Code, get one town

Let's Try it, HTTP Get

Run API, Write Code, get all towns

Let's Try it, HTTP delete

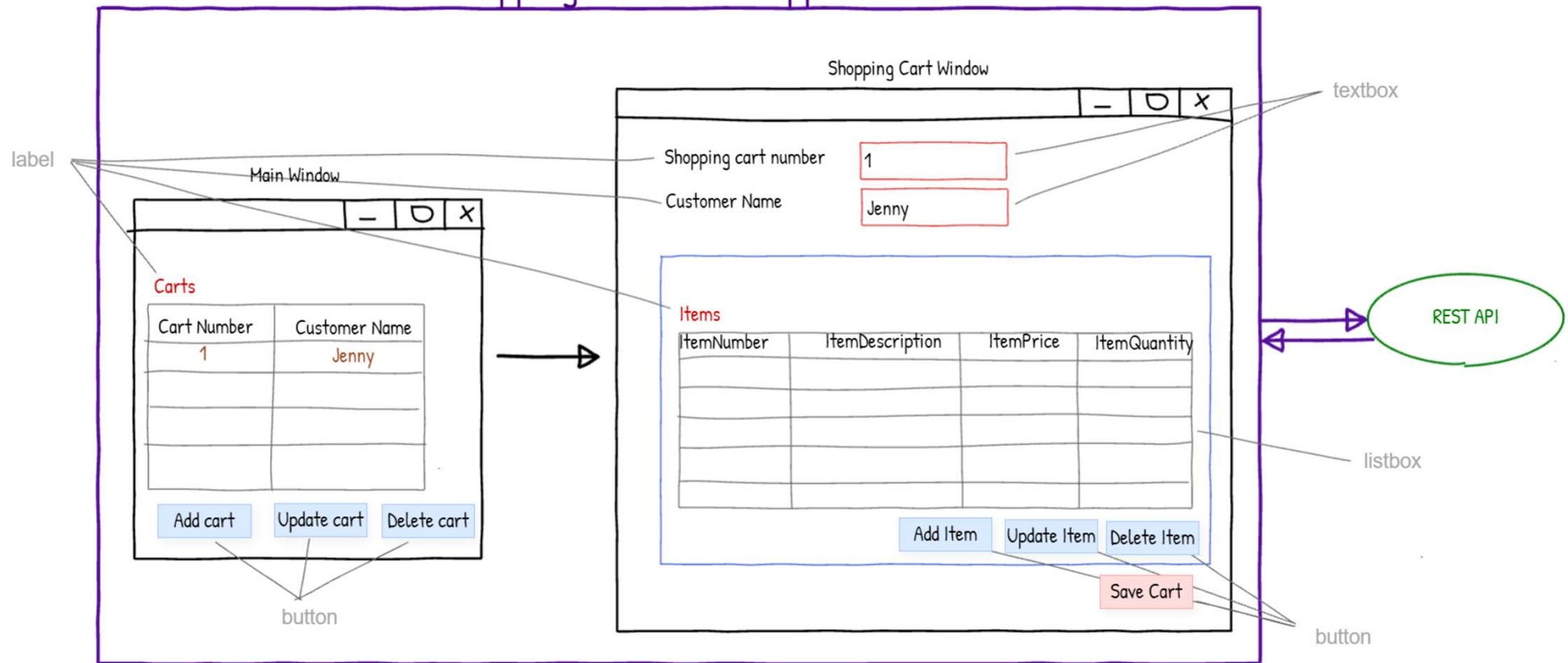
Run API, Write Code, Remove a Town

Let's Try it, HTTP Patch

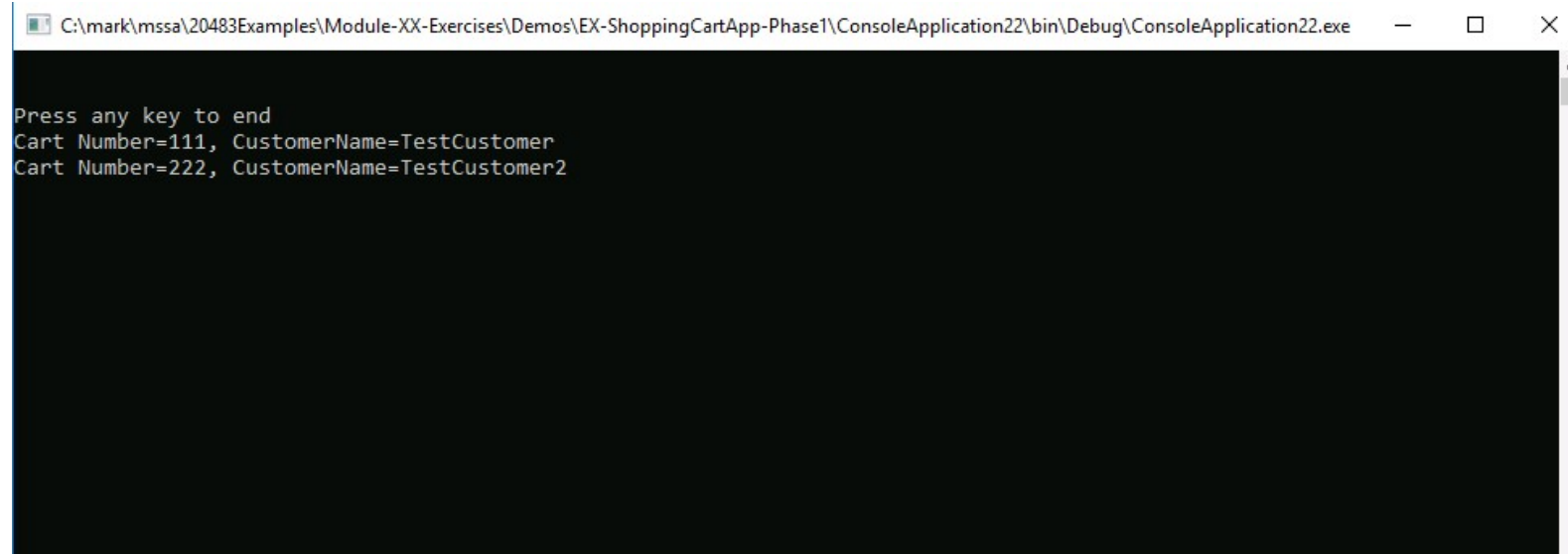
Run API, Write Code, update a Town

Exercise

Shopping.exe (A WPF Application)



Exercise Phase - 1



A screenshot of a Windows console window. The title bar shows the file path: `C:\mark\mssa\20483Examples\Module-XX-Exercises\Demos\EX-ShoppingCartApp-Phase1\ConsoleApplication22\bin\Debug\ConsoleApplication22.exe`. The console output is as follows:

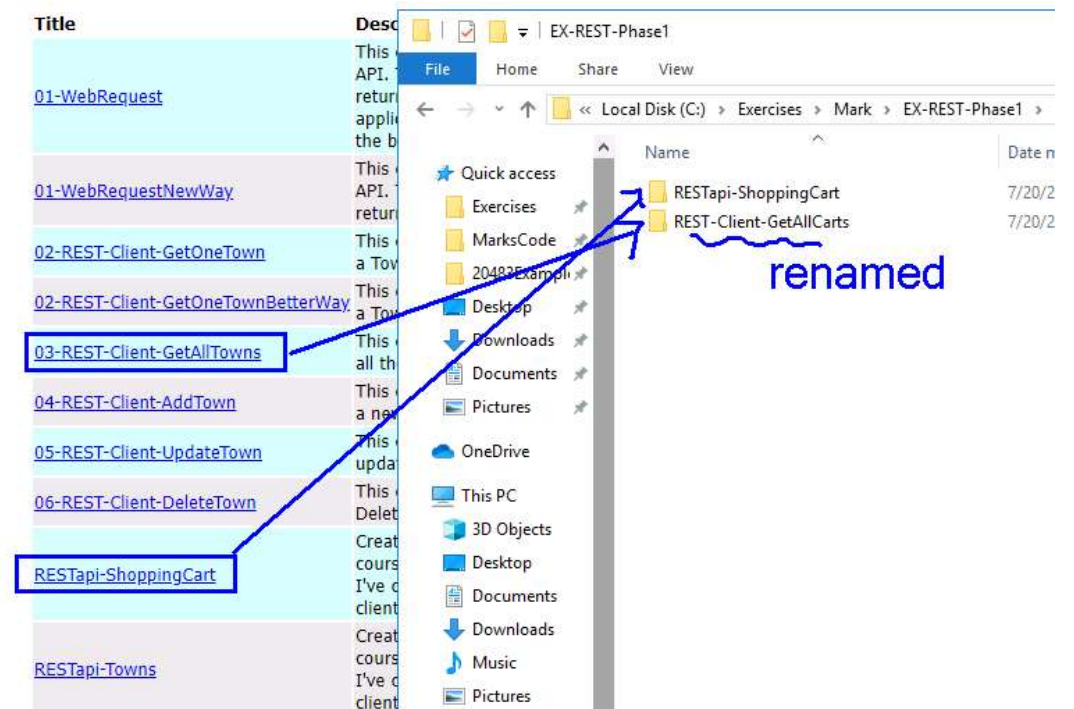
```
Press any key to end  
Cart Number=111, CustomerName=TestCustomer  
Cart Number=222, CustomerName=TestCustomer2
```

Exercise Phase - 1

1. Create new git branch for new work

```
Windows PowerShell
PS C:\Exercises> git branch
* Mark-GetBook-NewWay
  main
PS C:\Exercises> git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
PS C:\Exercises> git branch
  Mark-GetBook-NewWay
* main
PS C:\Exercises> git branch Mark-REST-Phase1
PS C:\Exercises> git checkout Mark-REST-Phase1
Switched to branch 'Mark-REST-Phase1'
PS C:\Exercises>
```

2. Copy ShoppingCart RestAPI and “getAllTowns” demo 3. Rename Folder



4. Open the RESTapi-ShoppingCart application in Visual Studio and run it.

5. Learn about it using Swagger.

6. Change the code in the REST api client, start with the *Model.cs* file.

Exercise Phase - n

Clean-up your Remote Branch

1. Merge into the main branch on GitHub with a PR

Clean-up your local branches

1. `git checkout main`
2. `git remote prune origin`
3. `git branch`
4. *for each **local** branch where you have finished work*
`git branch --delete branchname`

The End