



VB.Net
Programming



DR ZURIANI
MUSTAFA

AL - FATIHAH

CH 6 | STATE MANAGEMENT

COURSE OUTCOMES

Differentiate between Client Side State Management and Server Side State Management

- On the **internet** there is **one problem**: the **web server does not know who you are and what you do** because the HTTP address doesn't maintain state.
- The HTTP protocol that browsers use to communicate with Web servers over the Internet or over an intranet is a **“stateless”** protocol.
- The word **“state”** is used in this context to refer to **data**.
- When you type a word into an empty text box on a Web form, **that word becomes part of the state** of the form, along with all the other pieces of data that make up the contents and formatting of that Web page.
- But if you call the same page again, the text box will be **empty**, unless the Web application does something to record the data you entered and make it part of the **“state”** of the page that is sent back to your browser.

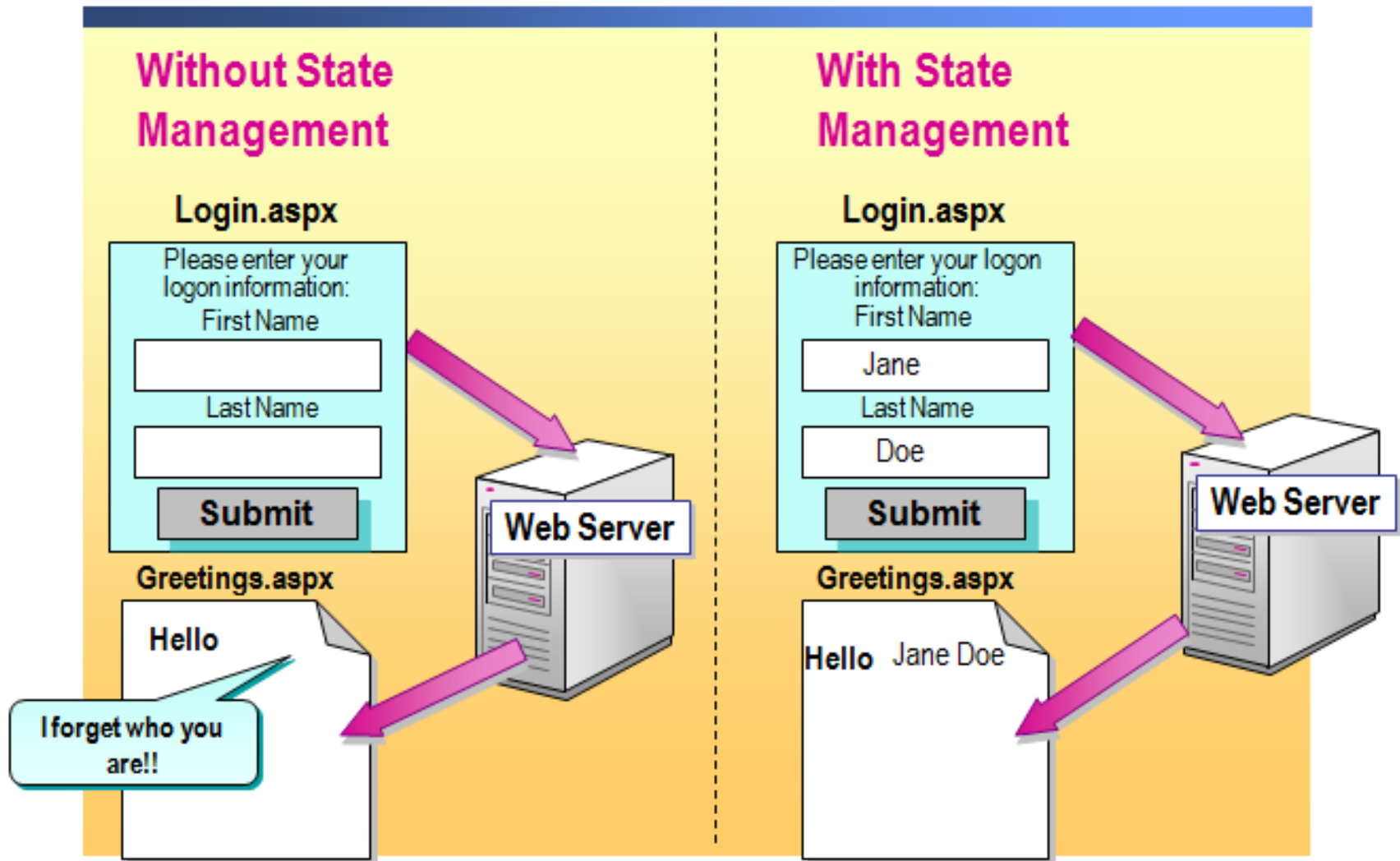
Web is meant to be Stateless?

- HTTP (the protocol over which Internet or the Web works) is a stateless protocol and it works on **Request** from clients to server and **Responses** from server back to client.
- Beyond a Request/Response cycle, the server doesn't know/care about the state of the client

Recall:

- Hyper Text Transfer Protocol (HTTP) is a communication protocol which is implemented in the "World Wide Web(WWW)".
- It is a request/response style protocol. Clients (browsers, spider, etc) will request to a server (web server) and the server responds to these requests.
- HTTP uses TCP protocol for communication. It connects to a specific port (default is 80) to the server and communicates via that port. Once the response is received completely, client programs will be disconnected from the server. For each request, client programs have to acquire a connection with servers and do all the request cycles again.

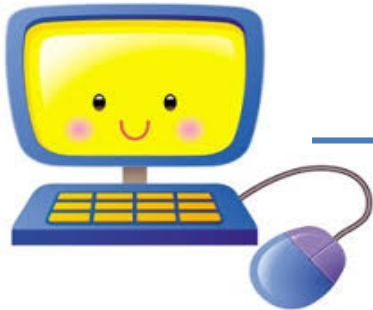
What is state management?



- There are mainly two types of state management that ASP.NET provides:
 1. Client side state management
 2. Server side state management

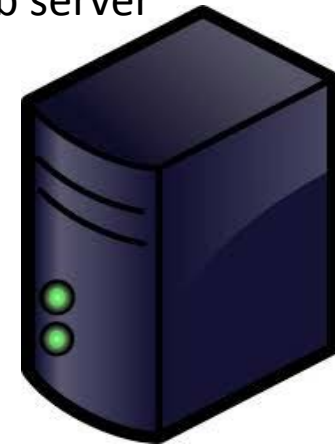
Client Side State Management

Client



I am Jane.
I am shopping for bags.
I have three titles in my cart totaling RM500

Web server



When we use client side state management:

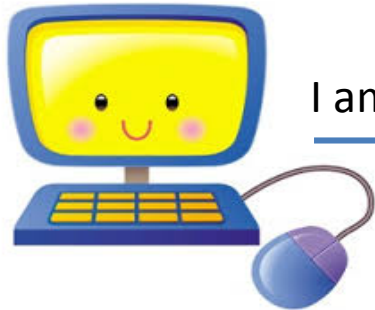
- The state related information will be stored on client side.
- This information will travel back and forth with every request and response.

Client Side State Management

- Advantage:
 - Relieve the server from the burden of keeping the state related information
- Disadvantage:
 - Makes the information travel back and forth
 - Information can be intercepted by anyone in between
 - Sensitive information (credit card number, password)?

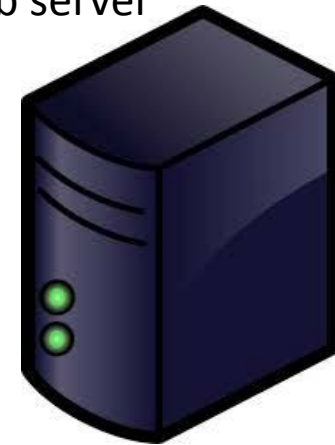
Server Side State Management

Client



I am session 237

Web server



Session 237 is Jane.

Jane is shopping for bags.

I have three titles in Jane's cart totaling RM500

When we use server side state management:

- The state related information will be stored on server memory.

Server Side State Management

Advantage:

- Sensitive information is secure

Disadvantage:

- More memory usage on server

Mix of client side and server side state management depending on the type and size of information.

- Different techniques to manage the state on client and server:

Client Side State Management

- **View State**
- Query String
- Hidden Fields
- Cookies

Server Side State Management

- Application State
- **Session State**

Client Side State Management

- View State
 - For saving state on a page between post backs(for a single page)
 - Insecure
- Query String
 - Are used to pass data between pages (in the url)
 - It's temporary storage because when user close the browser or paste another URL data will be lost.
- Hidden fields
 - Similar to textbox
- Cookies
 - Stored at the client. Can expire.

```
<form name="_ctl0" method="post" action="page.aspx" id="_ctl0">  
<input type="hidden" name="__VIEWSTATE"  
value="dDwtNTI0ODU5MDE1Ozs+ZBCF2ryjMpeVgUrY2eTj79HN14Q=" />
```

```
.....some code
```

```
</form>
```


Server Side State Management

Session

- For saving data specific to the user
- It is very secure because data never transmitted to client, stores at server side, storing large amount of data may slow down the performance, lifetime is default 20 min but can be altered.
- Sessions work by creating a unique id (UID) for each visitor and store variables based on this UID.
- Session variables hold information about one single user, and are available to all pages in one application.

Server Side State Management using Session State

- Session objects are stored on the server side.
- It can be used to store any type of data in the memory.
- They are primarily used for storing user specific information like shopping cart, preferences etc and is never transferred to the client.
- You can declare session level variables in the following manner -

```
Session("AccountNo") = 1998773887324556;
```

- To destroy session:

```
Session.Remove("AccountNo")
```

Session

- A session is defined as **the period of time** that a unique user interacts with a web application.
- As long as the user maintains their state, you can use the session object as much as you want.
- ASP developers who wish to retain data for unique user session can use a feature known as session state.

ASP.NET Session State

- The session state is used to maintain the session of each user throughout the application.
- Session allows information to be stored in one page and access in another page and support any type of object.

Example:

- In many websites we will see the functionality like once if we login into website they will show username in all the pages for that they will store username in session and they will access that session username in all the pages.
- Whenever user enters into website new session ID will be generated for that user
- This session ID will be deleted when he leave from that application. If he enters again he will get new session ID.

STATE MANAGEMENT

FASHIONVALET

Wanderlust + Co

ADD EDGE TO ANY ENSEI

WHAT'S NEW | WOMEN | SHOES | ACCESSORIES | SAL

SIGN IN / REGISTER

SIGN IN

Already registered? Please Login Below:

Email Address*

zuriani.m@gmail.com

Password*

☒ Remember Me What's this?

* Required Fields

[Forgot Your Password?](#)

LOGIN



FASHIONVALET

Wanderlust + Co

ADD EDGE TO ANY ENSEI

WHAT'S NEW | WOMEN | SHOES | ACCESSORIES | SAL

MY ACCOUNT

Account Dashboard

Account Information

Address Book

My Orders

My Wishlist

My Dashboard

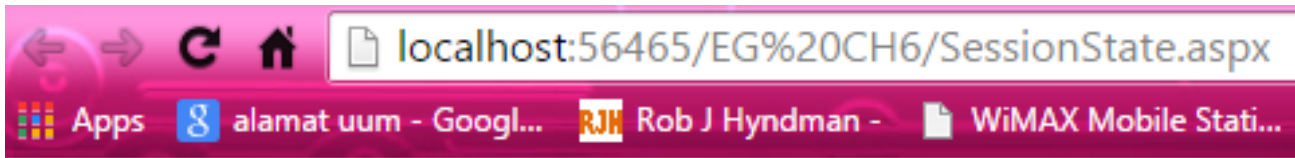
Hello, **ZURIANI MUSTAFFA!**

From your My Account Dashboard you have tl
account information. Select a link below to vie

Recent Orders

ORDER #	DATE	SHIP TO
---------	------	---------

STATE MANAGEMENT



First Name

Last Name

```
Session("FirstName") = TextFirstName.Text  
Session("LastName") = TextLastName.Text  
Response.Redirect("SessionState2.aspx")
```



Welcome Siti Fatimah Muhammad

First Name Siti Fatimah

Last Name Muhammad

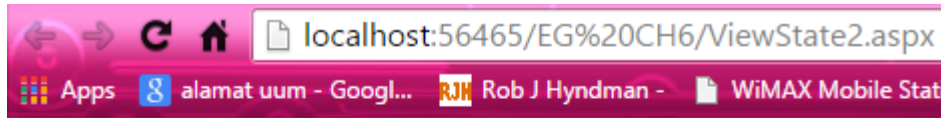
```
LabelString.Text = Session("FirstName") & Session("LastName")  
LabelFirstName.Text = Session("FirstName")  
LabelLastName.Text = Session("LastName")
```

Client Side State Management using ViewState

- ASP.NET uses this mechanism to track the values of the controls on the web page between page request for **same page**.
 - Client-side management techniques don't utilize server resources.
 - They are less secure but fast performing.
 - Let's explore the ViewState technique to save state
- ViewState objects are used to store page related data/State values which are preserved between the round trips from client to server and server to client/across page PostBacks.

STATE MANAGEMENT

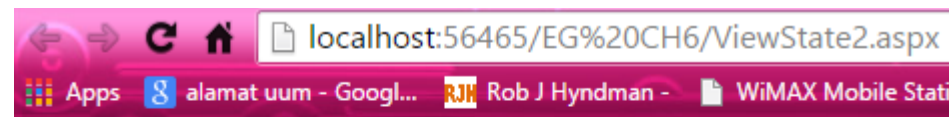
1



Button Click:

```
ViewState("NameOfUser") = TextBoxName.Text  
LabelName.Text = ViewState("NameOfUser")
```

2

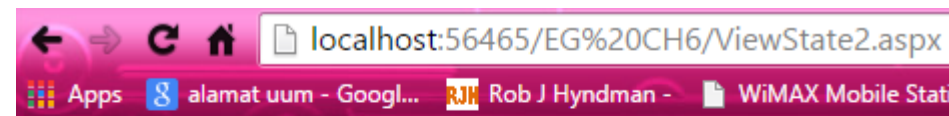


Page Load:

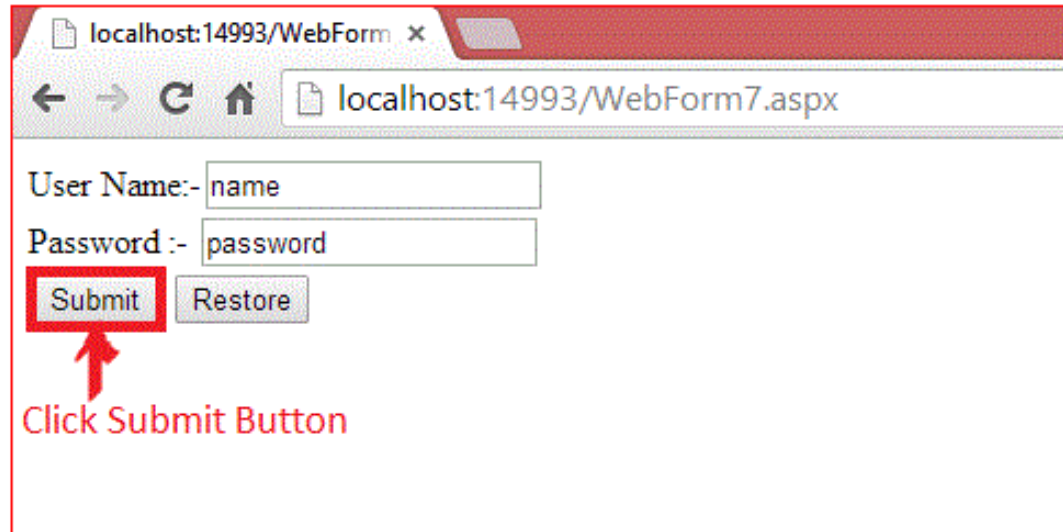
```
If ViewState("NameOfUser") IsNot Nothing Then  
    LabelName.Text = ViewState("NameOfUser").ToString()  
Else  
    LabelName.Text = "Not set yet..."  
End If
```

Hit this button

3



STATE MANAGEMENT| Example ViewState



localhost:14993/WebForm x

localhost:14993/WebForm7.aspx

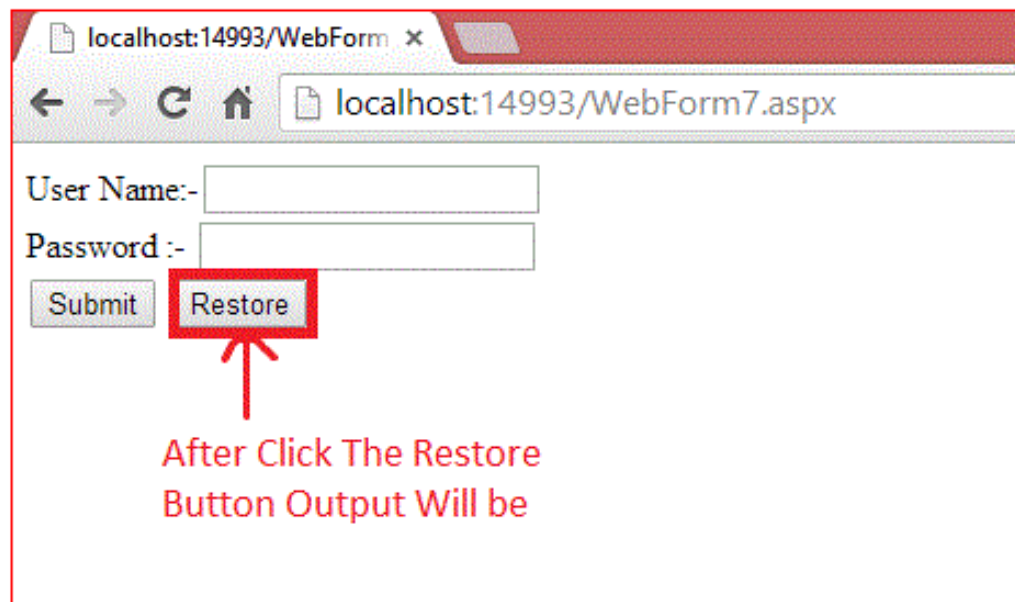
User Name:-

Password :-

Submit Restore

Click Submit Button

Without ViewState



localhost:14993/WebForm x

localhost:14993/WebForm7.aspx

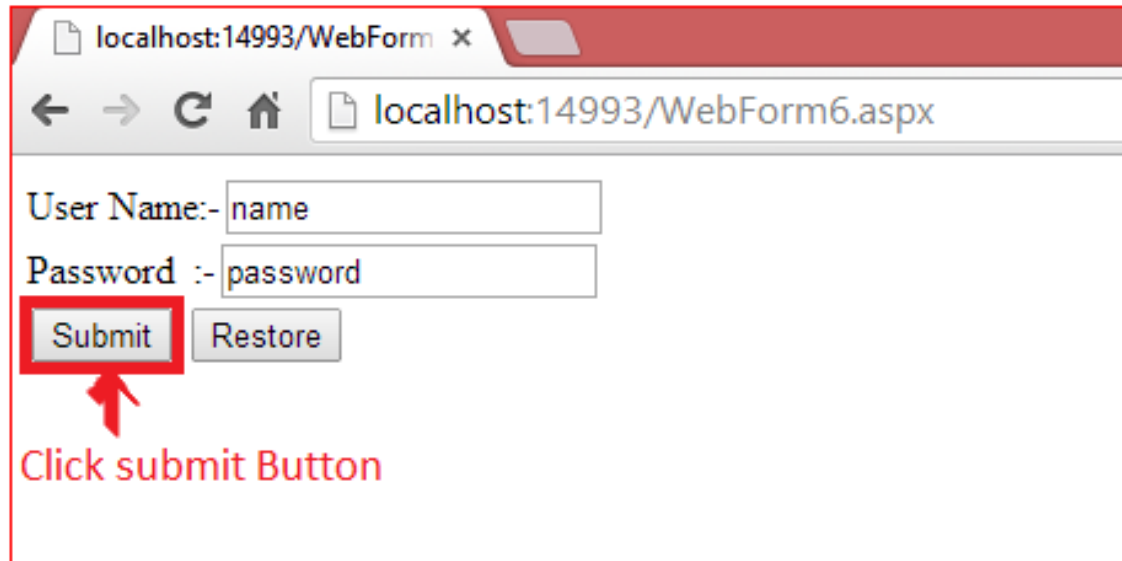
User Name:-

Password :-

Submit **Restore**

After Click The Restore Button Output Will be

STATE MANAGEMENT| Example ViewState



localhost:14993/WebForm x

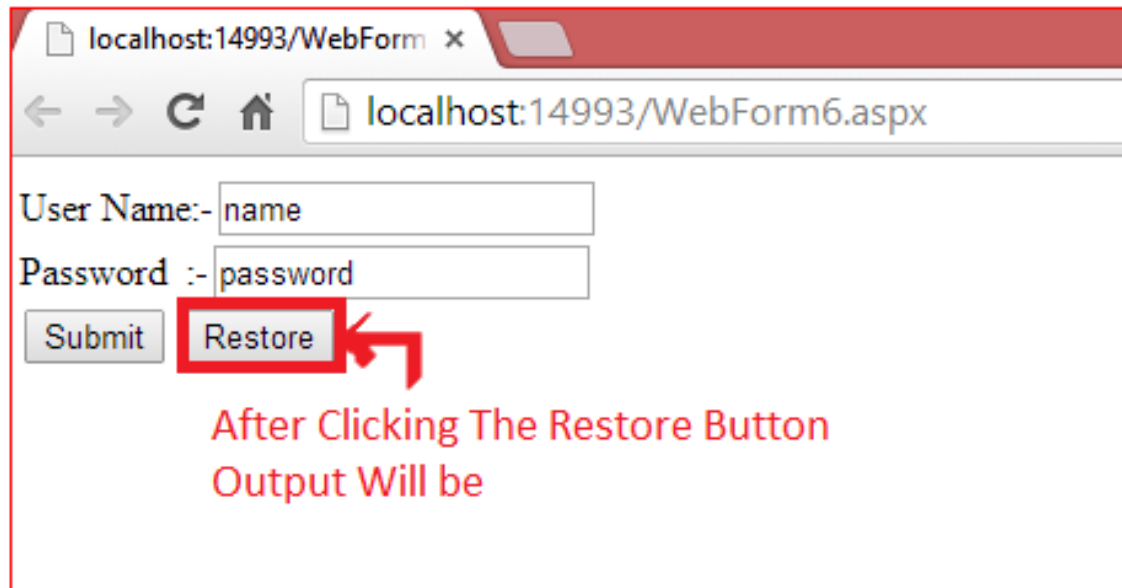
localhost:14993/WebForm6.aspx

User Name:-

Password :-

Click submit Button

With ViewState



localhost:14993/WebForm x

localhost:14993/WebForm6.aspx

User Name:-

Password :-

After Clicking The Restore Button
Output Will be

Session State	View State
Session state is maintained in session level	View state is maintained in page level only
Session state value is available in all pages within a user session	View state of one page is not visible in another page
Session state information stored in server	View state information stored in client only

Usage

- If you want to access the information on different web pages, you can use SessionState
- If you want to access from the same page, then you can use Viewstate

Security

- Session state provides more security compared with view state as the data value is stored in server side

TO BE
CONTINUED