

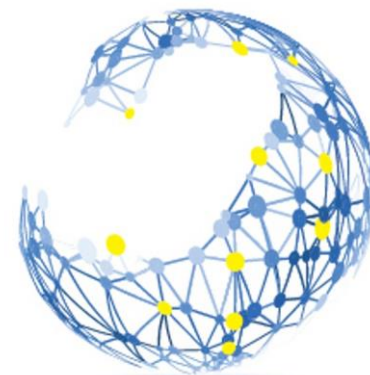
תכנות באינטרנט

תכנות צד שרת

Stateless Programming

גלעד מרקמן

קריית החינוך פארק המדע, נס
ציונה



קריית החינוך
השש שנתית
פארק המדע
עמיד של חדשנות



- אובייקט Response ואובייקט Request
- כתיבת קוד צד שרת בדף asp (HTML) - `<% %>`.
- כתובת קוד בקובץ נפרד – Code Behind
- שינוי דף ה-HTML באמצעות קוד בצד השרת.

Stateless Programming

- תכנות "חסר מצב" – Stateless Programming
- אובייקט Session, אובייקט Application.
- Global.asax – ניהול אירועים ברמת היישום (Application).

- השרת אינו שומר נתונים על ההתקשרות עם הלקוח.
- לאחר שהלקוח שלח Request וקיבל Response הקשר עם המשתמש (הדפדפן) "מנותק" והשרת אינו שומר נתונים על הדף שנשלח.
- שיטה זו מאפשר לשרת לטפל במקביל במספר רב של משתמשים.

- לחיצה על הכפתור מעלה מ-0 ל-1, אבל לא ל-2.

Lesson 2

Counter 0

Add

```
<body>
  <h1 style="text-align:center">Lesson 2</h1>
  <form id="form1" runat="server">
    <div>
      Counter
      <input type="text" name="counter" id="counter" runat="server"/>
      <br />
      <input type="submit" name="Add" id="Add" value="Add" />
    </div>
  </form>
</body>
```

2 references

```
public partial class Pages_Lesson2_0 : System.Web.UI.Page
{
    0 references
    protected void Page_Load(object sender, EventArgs e)
    {
        int val=0;
        if (!IsPostBack)
        {
            val = 0;
            counter.Value = val.ToString();
        }
        else
        {
            val++;
            counter.Value = val.ToString();
        }
    }
}
```

Lesson 2

Counter 5

Add

```
0 references
protected void btnClick(object sender, EventArgs e)
{
    if (counter.Value == "")
    {
        counter.Value = "1";
    }
    else
    {
        int val = int.Parse(counter.Value);
        val++;
        counter.Value = val.ToString();
    }
}
```

```
<body>
  <h1 style="text-align:center">Lesson 2</h1>
  <form id="form1" runat="server">
    <div>
      Counter
      <input type="text" name="counter" id="counter" runat="server"/>
      <br />
      <input type="button" name="Add" id="Add" value="Add" runat="server" onclick="btnClick"/>
    </div>
  </form>
</body>
```

```
2 references
public partial class Pages_Lesson2_0 : System.Web.UI.Page
{
    0 references
    protected void Page_Load(object sender, EventArgs e)
    {
        if (Session["counter"]==null)
            Session["counter"] = 0;
        counter.Value = Session["counter"].ToString();
    }

    0 references
    protected void btnClick(object sender, EventArgs e)
    {
        Session["counter"] = (int)Session["counter"] + 1;
        counter.Value = Session["counter"].ToString();
    }
}
```

- אובייקט Session מאפשר לשמור נתונים המשותפים ל Session ("שיחה") עם משתמש.
- המשתנה בתוך Session הוא אובייקט, ולכן יש לבצע המרה Casting כדי לעשות בו שימוש.
- האובייקט נשמר גם אם הדף נסגר.
- האובייקט משותף לכל הדפים באותו .session

- באובייקט Session ניתן להגדיר מספר משתנים (מאפיינים).
- הפניה למשתנה נעשית על ידי המילה Session ובסוגריים שם המשתנה:
Session ["name"]
- אין צורך להגדיר משתנה Session. ברגע שמציבים ערך בתוך משתנה, המשתנה מוגדר אוטומטית.
- האובייקט נמחק בסיום ה-Session. ה-Session מסתיים באחד משניים:
 - העדר פעילות למשך זמן שנקבע במאפיין Session.Timeout. ברירת המחדל 20 דק'.
 - יציאה באמצעות קוד Session.Abandon().
- ה-Session אינו מסתיים כאשר המשתמש סוגר את הדפדפן.


```
2 references
public partial class Pages_Lesson2_0 : System.Web.UI.Page
{
    0 references
    protected void Page_Load(object sender, EventArgs e)
    {
        if (Session["counter"]==null)
            Session["counter"] = 0;
        if (Application["counter"]==null)
            Application["counter"] = 0;

        ApplicationCounter.Value = Application["counter"].ToString();
        SessionCounter.Value = Session["counter"].ToString();
    }
    0 references
    protected void btnClick(object sender, EventArgs e)
    {
        Application["counter"] = (int)Application["counter"] + 1;
        ApplicationCounter.Value = Application["counter"].ToString();
        Session["counter"] = (int)Session["counter"] + 1;
        SessionCounter.Value = Session["counter"].ToString();
    }
}
```

- האובייקט Application משותף לכל הפרויקט שלנו ולכל ה Sessions.
- הפניה למשתנים בתוך ה- Application נעשית באמצעות שם המשתנה: Application["Name"].
- המשתנה הוא אובייקט ושימוש בו מחייב המרה Casting.
- האובייקט נוצר בהפעלת האתר ונמחק כאשר האתר מפסיק "לרוץ".

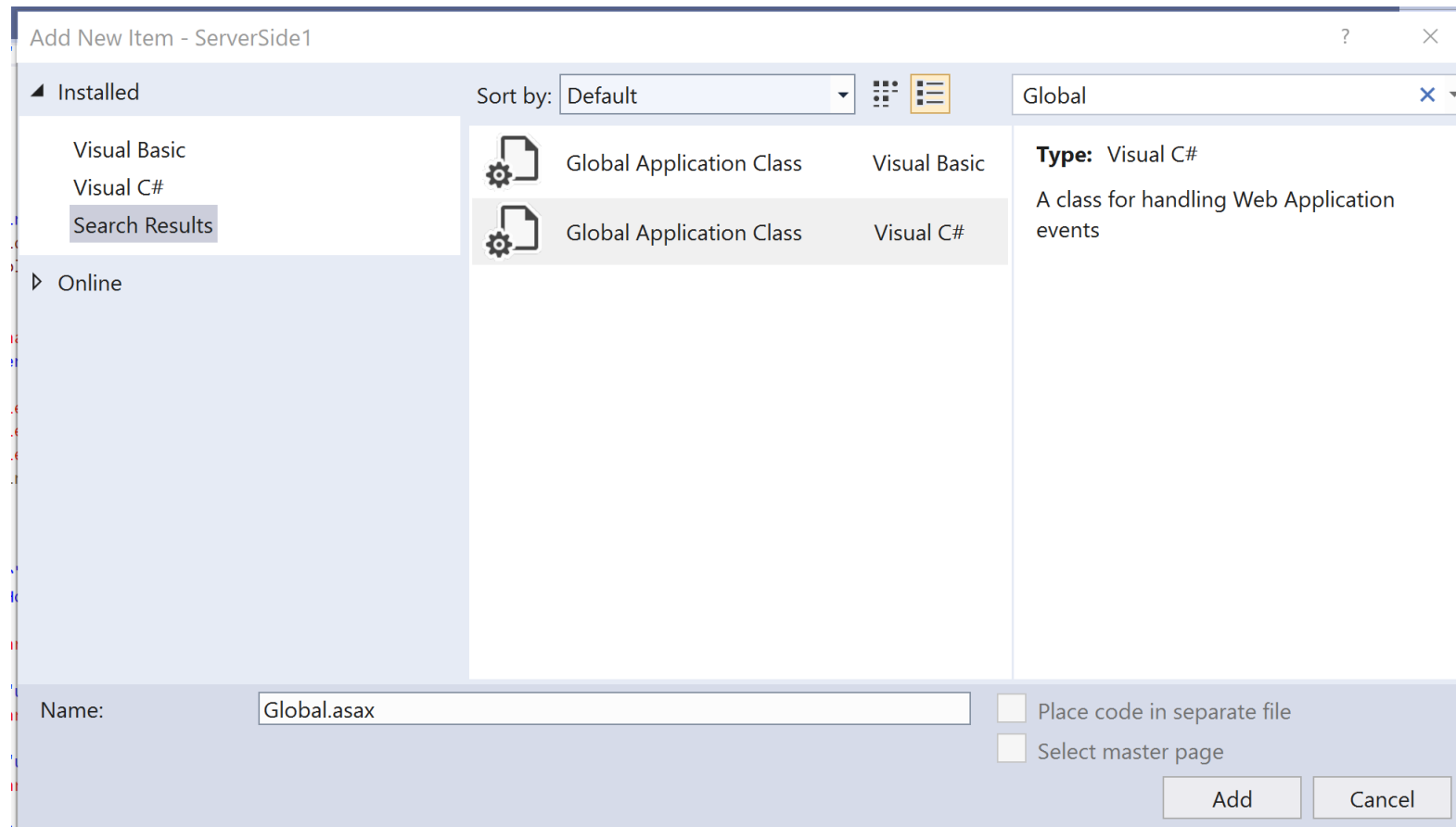
• דף הממוקם בספריית השורש ובו קיימים "אירועים" ברמת היישום:

- Application_Start
- Application_End
- Application_Error
- Session_Start
- Session_End

• הדף מאפשר לבצע פעולות, כגון איתחול משתנים, בעת קרות אחד האירועים הללו.

```
Global.aspx
1 <%@ Application Language="C#" %>
2
3 <script runat="server">
4
5     void Application_Start(object sender, EventArgs e)
6     {
7         // Code that runs on application startup
8
9     }
10
11    void Application_End(object sender, EventArgs e)
12    {
13        // Code that runs on application shutdown
14
15    }
16
17    void Application_Error(object sender, EventArgs e)
18    {
19        // Code that runs when an unhandled error occurs
20
21    }
22
23    void Session_Start(object sender, EventArgs e)
24    {
25        // Code that runs when a new session is started
26        Session["Login"] = false;
27    }
28
29    void Session_End(object sender, EventArgs e)
30    {
31        // Code that runs when a session ends.
32        // Note: The Session_End event is raised only when the sessionstate mode
33        // is set to InProc in the Web.config file. If session mode is set to StateServer
34        // or SQLServer, the event is not raised.
35
36    }
37
```

- ניתן ליצור רק דף גלובלי אחד בפרויקט, באמצעות Add New Item.



- תכנות "חסר מצב" – Stateless Programming
- אובייקט Session, אובייקט Application.
- הגדרת משתנים באובייקטים: Application["name"], Session["name"]
- .Session.Timeout = 30, Session.Abandon()
- Global.asax – ניהול אירועים ברמת היישום (Application).