

تقانات الانترنت

الأستاذ: علاء الخوام

الطالبة: نور الهدى شعبان

دورة حياة صفحات الويب الخاصة بمشاريع
asp.net

وعناصر التحكم Asp.net 2.0 دورة حياة صفحة الديناميكية:

إن فهم دورة حياة الصفحة مهم جدًا لتطوير تطبيقات asp.net.

واجه العديد من مبتدئي net. مشاكل مثل فقدان القيم والحالة بعد عمليات إعادة النشر عند التعامل مع عناصر التحكم المحملة ديناميكيًا. بروتوكول HTTP عديم الحالة.

هذه مشكلة طبيعية بين برامج الويب وبرامج Windows. إذا كنت تريد معرفة asp.net ، فستكون دورة حياة الصفحة واحدة من أهم الأسس الخاصة بك. ما هو ترتيب الأحداث، خاصة بعد إضافة صفحة اللوحة الأم في asp.net 2.0 ، مما يجعل الأمر أكثر تعقيدًا، والغرض من هذه المقالة هو شرح ترتيب كل حدث والغرض منه لجعلك تفهم الأحداث ماذا يحدث.

في تطبيق asp.net ، يطلب المستخدم دائمًا صفحة .aspx. الشيء الوحيد الذي يهمنا هو ما يفعله خادم الويب الذي ينتمي إليه التطبيق عندما يزور المستخدم صفحة .aspx. ماذا عن ذلك؟

سيساعدنا فهم تسلسل الأحداث على فعل ما نريده في الأحداث الصحيحة، ويمكنه أيضًا القضاء على بعض ارتباكنا، مثل إلقاء اللوم على بعض المشاكل في انعدام الجنسية لبرامج الويب.

دورة حياة الصفحة

من المهم جدًا فهم كل طلب في دورة حياة الصفحة. قد يرجع سبب مشكلة فقدان القيمة والحالة إلى عدم فهمك الكافي لدورة حياة الصفحة. بالطبع، إذا كنت تريد الاحتفاظ بالحالة في asp.net ، فيمكنك استخدام تطبيقات مثل Application أو Session أو Cookies أو Cache.

ملف الإعدادات web.config: هو أهم ملف في ملفات أي مشروع ASP.NET لأنه يحتوي على جميع إعدادات المشروع مثل تحديد اللغة وجملة الربط بقواعد البيانات والمجلدات التي تريد حمايتها الخ وبهذا يمكنك التعديل في خصائص المشروع ككل من خلال التعديل في هذا الملف.

وهذا الملف هو عبارة عن ملف نصي عادي على شكل XML يمكن التعديل عليه باستخدام أي محرر نصوص مثل. notepad

يمكن أن يحتوي كل مجلد من مجلدات المشروع على ملف web.config ولكن ليس ضرورياً لأن ملف web.config يؤثر على المجلدات الفرعية للمجلد الذي يوجد فيه. فمثلاً لو كان هناك ملف web.config في c:\inetpub\wwwroot فإن الإعدادات تطبق على جميع المجلدات الفرعية لهذا المجلد ما لم يوجد ملف web.config لمجلد داخلي يحتوي على إعدادات مختلفة.

إن ملف Global.asax هو ملف اختياري يحوي على التعليمات البرمجية للأحداث Events على مستوى التطبيق مثل بداية التطبيق Application Start ونهايته Application End والأحداث الخاصة بالأخطاء.

يتم تحليل ملف Global.asax وتجميعه في .Net Framework.

يمكن إضافة Global.asax عن طريق اختيار Add من القائمة ثم NewItem ثم Global Application Class ، بعد إضافتك للملف يمكنك إضافة التعليمات البرمجية التي تريدها في أي حدث من الأحداث التالية التي تنشأ بشكل افتراضي في الملف:

Application_Start

Application_End

Session_Start

Session_End

Application_BeginRequest

Application_AuthenticateRequest

Application_Error

runat = "server": يتيح ذلك إتاحة التحكم في asp.net على جانب الخادم. وبالتالي جعله عنصر تحكم الخادم. هذا يعني؛ على الزر الخاص بك انقر فوق الحدث في التعليمات البرمجية خلف الملف يمكنك قراءة قيمة مربع النص. أو قم بتعيين نص lable على صفحة الويب من التعليمات البرمجية الموجودة في الملف. ويتم تنفيذ هذا الرمز على خادم الويب.

خاصية:Page.IsPostBack

سيتم تشغيل روتين Page_Load في كل مرة يتم فيها تحميل الصفحة. إذا كنت تريد فقط تنفيذ التعليمات البرمجية في Page_Load في المرة الأولى التي يتم فيها تحميل هذه الصفحة ،

يمكنك استخدام خاصية Page.IsPostBack. إذا كانت خاصية Page.IsPostBack خاطئة، فسيتم تحميل الصفحة للمرة الأولى، إذا كان ذلك صحيحًا يتم إعادة نشر الصفحة الى الخادم

عند النقر فوق عنصر تحكم زر أو LinkButton أو ImageButton على صفحة ويب ASP.NET ، يتم إرسال النموذج إلى الخادم. إذا تم تعيين خاصية AutoPostBack لبعض عناصر التحكم إلى true ، فعندما يتم تغيير حالة عنصر التحكم، سيتم أيضًا إرسال النموذج الى الخادم

في تطبيق ASP.NET ، إذا كنت بحاجة إلى إجراء بعض عمليات
التهيئة عند عرض الصفحة لأول مرة، يجب تحديد خاصية
IsPostBack!

استخدم Page.IsPostBack في ASP.NET ، ثم يمكنك تجنب العمل
الإضافي في رحلة ذهابًا وإيابًا: إذا كنت تتعامل مع إعادة نشر التحكم
بالخادم

فعادة ما تحتاج إلى تنفيذ الرمز عند طلب الصفحة لأول مرة ، والذي
يختلف عن رحلة ذهابًا وإيابًا عند تشغيل الحدث كود. إذا قمت بفحص
خاصية Page.IsPostBack

يمكن تنفيذ التعليمات البرمجية بشكل مشروط ، اعتمادًا على ما إذا
كان هناك طلب أولي للصفحة أو استجابة لأحداث التحكم بالخادم.
يبدو هذا واضحًا ، ولكن يمكنك في الواقع تجاهل هذا الاختيار دون
تغيير سلوك الصفحة. يرتبط استخدام هذه السمة بشكل مباشر بما إذا
كان برنامجك يعمل وفقًا لرغباتك الأولية ، ولكن أيضًا بكفاءة
الصفحة بأكملها. لأنه ، إذا كنت ستقوم بربط البيانات بعنصر التحكم
في كل مرة ، سواء كنت تقوم بالوصول إليها لأول مرة أو بعد تقديم
البيانات ، فيمكن تصور كفاءة برنامج الصفحة هذا.