

PHP Variablen

Die Variablen in PHP sind zu mindestens bei der Initialisierung Typen undefiniert. Es ist so, als würde man unter C# ausschließlich mit dem var arbeiten.

Weiter kann die Variable entweder local, global oder static sein. Ist eine Variable außerhalb von Scopes {} definiert, so ist diese global. Kann aber nicht in den Scopes verwendet werden.

Wird eine Variable in den Scopes definiert, so gilt sie als local und kann nur in den Scopes verwendet werden.

Befindet sich eine Variable in den Scopes und ist diese als static deklariert, so behält sie ihren Wert, auch wenn der Programmcode die Scopes bereits abgeschlossen hat.

```
[crayon-6741a1a723dad592184441/]
```

Das Resultat wäre so eben 1 2 3. Hätte man hier nicht static genutzt, wäre das Ergebnis immer 1, da die Variable, wie bereits angesprochen gelöscht wird, sobald sie die Scopes verlässt.

Interessant sind auch die anonymen Methoden unter PHP. Man hat die Möglichkeit die Funktion quasi in eine Variable zu stecken. Nimmt man das obere Beispiel, so kann man einfach

```
[crayon-6741a1a723db3361181398/]
```

dazu verwenden die Funktion zu anonymisieren.

Variablentypen

PHP hat weit weniger Typen als z.B. noch C#.

Während der boolean, integer, string und null gleich sind, haben folgende Typen so ihre Besonderheiten.

Eine numerische Zahl mit Nachkommastellen ist zwar ein **float**,

fragt man aber mit

[crayon-6741a1a723db5228840998/]

den Variablentypen ab, sagt PHP, dass es ein **double** ist.

Der **array** in PHP darf alle Möglichen Typen enthalten und erfüllt sogar die Funktion einer aus C# bekannten List oder Diktionary mit Key and Value Prinzip.

Betrachtet man das manuel von PHP.net, so findet man auch eigenartige Funktionen um sogar die im array enthaltenen Werte zu summieren, zu mischen usw.

Man kann in PHP Funktionen und sogar Klassen in eine Variable oder als Variable? speichern. Der Typ ist dann ein object.

[crayon-6741a1a723db6708767703/]

Wie auch in C# kann man jeden anderen Typ in ein object konvertieren, nennt sich hier aber nicht casten... Dies geschieht aber im Prinzip genau so:

[crayon-6741a1a723db8749609834/]

Als letzter Type ist die Resource, welche quasi einen Zeiger auf externe Ressourcen zeigt. Diese beinhaltet z.B. das handeln mit Datenbanken, Pdf Dateien usw. Die folgende Liste zeigt einen Auszug davon:

Ressource-List

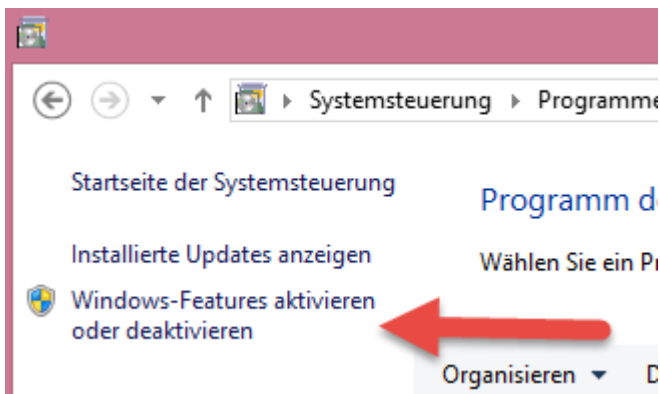
PHP 5.6.8 unter IIS und Windows 8

Mein nächstes Ziel ist die Webkomponente der Programmierung kennen zu lernen. Dazu gehört neben HTML, CSS, JavaScript auch die serverseitige Sprache PHP.

Erstes Ziel ist das Einrichten von PHP auf dem IIS (Man könnte natürlich den einfacheren Weg über XAMP gehen oder den alternativen Weg über den Webserver Apache). Ich wollte das einfach unter IIS machen, weil IIS eigentlich aus einfach von Haus aus auf dem Windows Rechner schlummert und ich so nicht wirklich was installieren brauche. Ok, los gehts.

1. Installieren von IIS

Systemsteuerung – Programme – Programme und Features auf



klicken und den Internetinformationsdienste nach folgenden Muster aktivieren.

Windows-Features aktivieren oder deaktivieren



Verwenden Sie die Kontrollkästchen, um die entsprechenden Features ein- oder auszuschalten. Ein ausgefülltes Kontrollkästchen bedeutet, dass ein Feature nur teilweise aktiviert ist.

- Internetinformationsdienste
 - FTP-Server
 - Webverwaltungstools
 - IIS-Verwaltungsdienst
 - IIS-Verwaltungskonsole
 - IIS-Verwaltungsskripts und -tools
 - Kompatibilität mit der IIS 6-Verwaltung
- WWW-Dienste
 - Allgemeine HTTP-Features
 - HTTP-Fehler
 - HTTP-Umleitung
 - Standarddokument
 - Statischer Inhalt
 - Verzeichnis durchsuchen
 - WebDAV-Veröffentlichung
 - Anwendungsentwicklungsfeatures
 - .NET-Erweiterbarkeit 3.5
 - .NET-Erweiterbarkeit 4.5
 - Anwendungsinitialisierung
 - ASP
 - ASP.NET 3.5
 - ASP.NET 4.5
 - CGI
 - ISAPI-Erweiterungen
 - ISAPI-Filter
 - Serverseitige Include-Dateien
 - WebSocket-Protokoll
 - Leistungsfeatures
 - Komprimieren dynamischer Inhalte
 - Komprimierung statischer Inhalte
 - Sicherheit
 - Anforderungsfilterung
 - Authentifizierung über Clientzertifikatzuordnung
 - Authentifizierung über IIS-Clientzertifikatzuordnung
 - Digestauthentifizierung
 - IP-Sicherheit
 - Standardauthentifizierung
 - Unterstützung zentraler SSL-Zertifikate
 - URL-Autorisierung
 - Windows-Authentifizierung
 - Systemzustand und Diagnose
 - Ablaufverfolgung
 - Anforderungsüberwachung
 - Benutzerdefinierte Protokollierung
 - HTTP-Protokollierung
 - ODBC
 - Protokollierungstools

OK

Abbrechen

Wenn IIS richtig installiert wurde, sind wir für uns unter `http://localhost/` erreichbar.

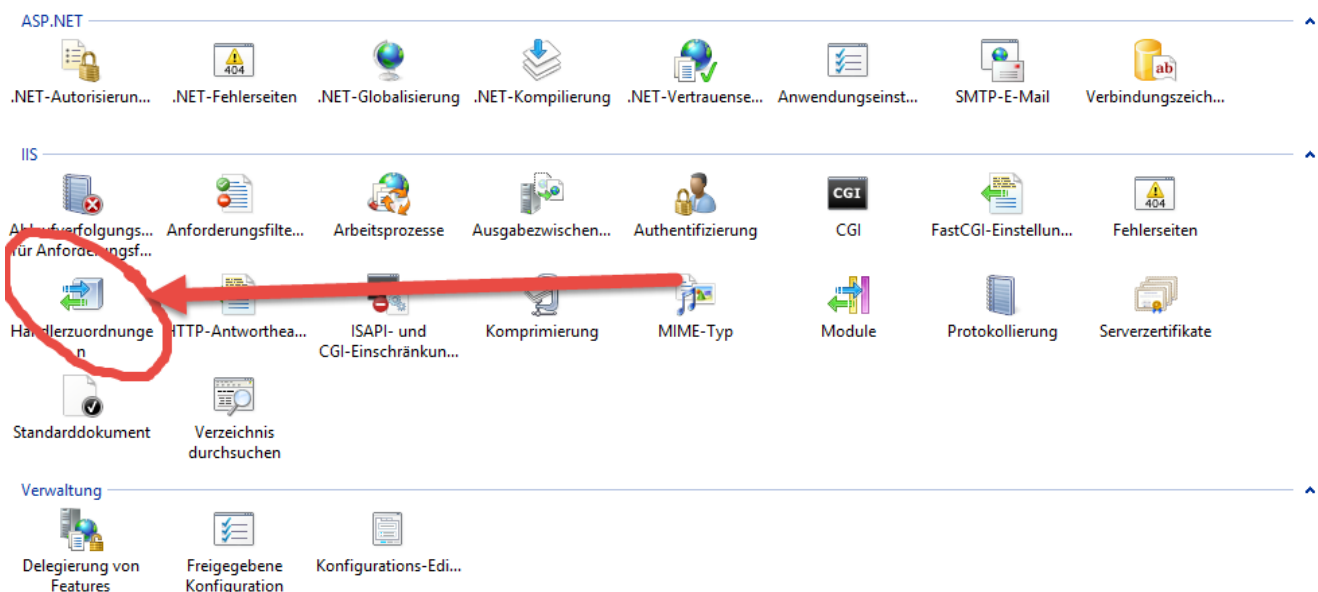
2. PHP installieren

Die Version **VC11 x86 Non Thread Safe** PHP für Windows auf `http://windows.php.net/download/` herunterladen, und in `C:\Program Files (x86)\PHP\php 5.6.8` entpacken

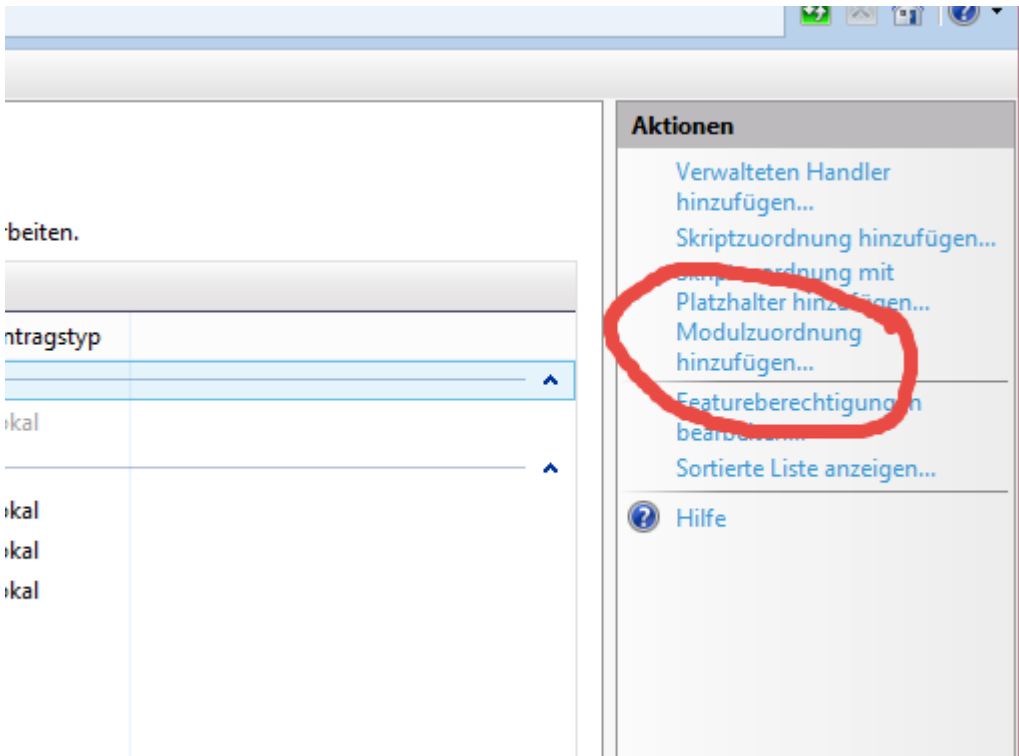
(Wegen den User Rechten muss man die Ordnerstruktur manuell anlegen)

3. IIS für PHP einrichten

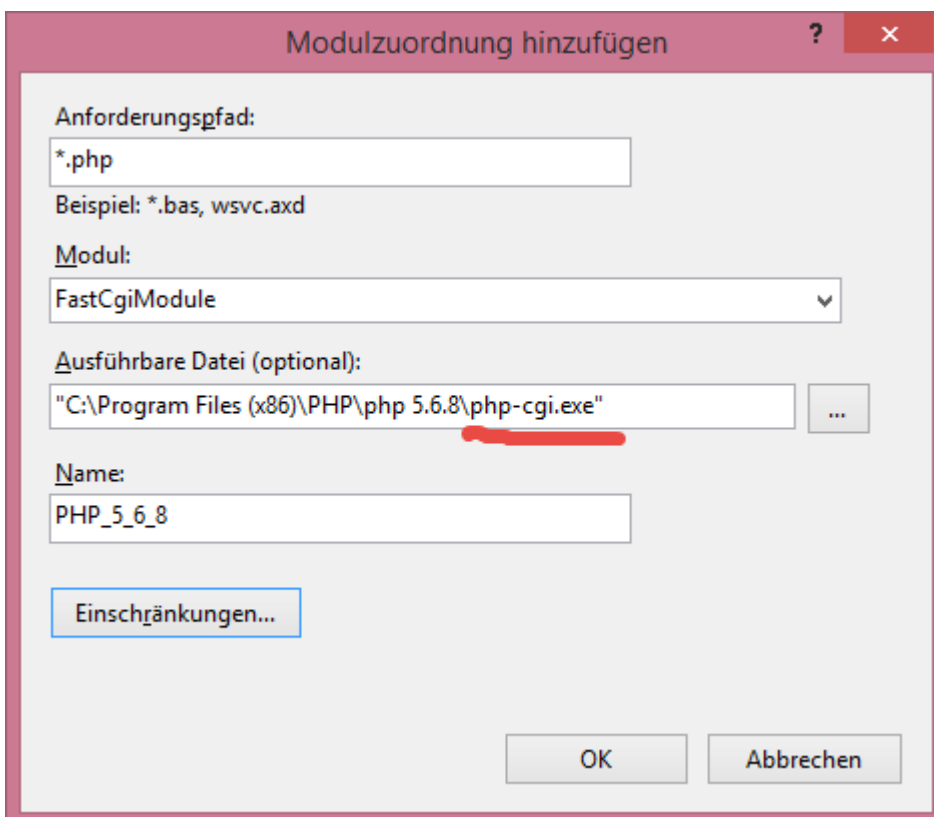
unter Start „IIS“ eintippen und den IIS-Manager wählen.



Dort auf Händlerzuordnungen und



Modulzuordnung hinzufügen...



Fenster folgend ausfüllen. Wichtig, bei der Ausführbaren Datei die **php-chi.exe** auswählen!

Dann auf OK und die Meldung mit **JA** auswählen.

Nun kann man eine info.php Datei mit dem Inhalt

```
<?php phpinfo() ?> in C:\inetpub\wwwroot speichern und  
schauen, was passiert wenn wir diese nun  
unter http://localhost/info.php versuchen zu öffnen.
```

Werden uns die Werte richtig dargestellt, ist alles richtig
installiert. Kann die Seite aber nicht geöffnet werden, fehlt
dem höchstwahrscheinlich das C++ Redistributable, welches man
hier [wiederum](http://www.microsoft.com/en-us/download/details.aspx?id=30679) [herunterladen](http://www.microsoft.com/en-us/download/details.aspx?id=30679)
kann <http://www.microsoft.com/en-us/download/details.aspx?id=30679>