

CGI und Perl

http://www

Brigitte Jellinek

Plan

- Was ist CGI?
- Entwicklungsumgebung
- Typische Fehler, Fehlersuche
- Perl + CGI
- Web-Formular und CGI
- Web-Formular und Perl

http://www

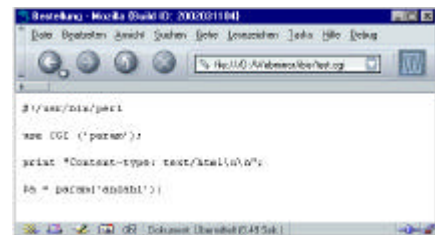
Common Gateway Interface

http://www

- Schnittstelle
Webserver <-> externes Programm
- Web Anfrage
- Programmstart
- Daten hinein über Kommandozeile und Umgebungsvariablen
- Daten retour über STDOUT, STDERR
- Programm endet ■

Entwicklung von CGI-Skripts

- Nur Browser ist nicht genug!



```
#!/usr/bin/perl
use CGI ('perl');
print "Content-type: text/html\n";
print "Hallo Welt!\n";
```

http://www

Entwicklung von CGI-Skripts

http://www

- Webserver !!!
- Editor
- Browser .



Mein Webserver auf Win

- www.nusphere.com
- Apache
- + Perl
- + PHP
- + MySQL



- in einer Installationsdatei .

http://www

http://www

Ohne Webserver auf Win

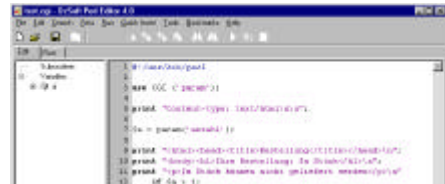
- DZ Soft Perl Editor
- nur 800 KB .



dzperl40.exe

Entwicklung von CGI-Skripts

- Spezieller Web-Editor
- Perl muss installiert sein
- emuliert den Server



http://www

http://www

Entwicklung von CGI-Skripts

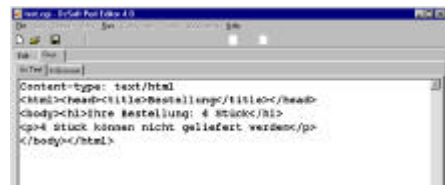
- Spezieller Web-Editor
- emuliert den Server



http://www

Entwicklung von CGI-Skripts

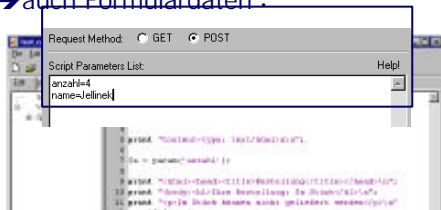
- Spezieller Web-Editor
- emuliert den Server



http://www

Entwicklung von CGI-Skripts

- Spezieller Web-Editor
- emuliert den Server
- auch Formulare Daten .



http://www

Entwicklung von CGI-Skripts

- Spezieller Web-Editor
- emuliert den Server
- auch Formulare Daten
- aber nicht Skript-->Skript-->Skript .

Fehler in CGIs

- Das Programm muss offline **funktionieren**
- erster Output:
Content-Type: text/html\n\n
- **Endung** und/oder **Ordner**
- **#!/usr/bin/perl**
- FTP Upload: **ASCII**
- **Zugriffsrechte** `rwxf-xf-x (755)` .

http://www

Fehlersuche in CGIs

- `stderr` geht an Webserver-Log
- oft nicht zugänglich
- Umleiten an den Browser
- **use CGI::Carp**
('fatalToBrowser');
- nur während der Entwicklung
- Wer es nicht macht verdient **kein Mitleid!** .

http://www

Perl + CGI

http://www

Perl unter UNIX

- Entweder:
auf der Kommandozeile mit
`perl dateiname`
- Oder:
 - Erste Zeile beginnt mit `#!`
 - Dann Pfad zu Perl: `/usr/bin/perl`
 - **`#!/usr/bin/perl`**
 - Kein Win-Zeilenumbruch am Ende!
 - Zugriffsrechte, z.B. `rwxf-xf-x` .

http://www

Das minimale CGI-Programm

```
#!/usr/bin/perl

print "Content-Type: text/html\n\n";
print "<h1>Hallo Welt</h1>";
.
```

http://www

Ausgabe

- `print ("Hallo ", $name, "\n");`
- `print "Hallo ", $name, "\n";`
- `print "Hallo $name!\n";`

- `print 'Datei in c:\windows kostet 3$';`
- `.`

http://www

http://www

HTML ausgeben

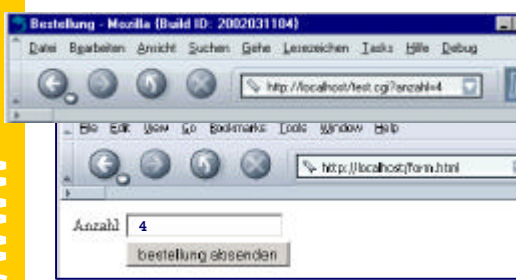
- print qq{<p align="\$a">H1</p>\n};
- print "<p>Hallo</p>\n";
- print "<p align='center'>Hi</p>\n";
- print "<p align='center'>Hi</p>\n";
- print '<p align="center">H1</p>\n';
- .

http://www

HTML ausgeben

```
print <<"ENDE":  
<p>Hallo</p>  
<p align="center">Hier kommt ganz  
viel Text, mit Variablen: $a  
ENDE  
if ($a > 1) {  
    ....  
};
```

http://www



http://www

Web-Formular und CGI (v1)

- Web-Formular
 <form action = "test.cgi"
 method = "get">
- URL
 test.cgi?n1=v1&n2=v2&n3=v3
- URL-Encoded:
 - Leerzeichen wird +
 - Sonderzeichen wird %xx
- Umgebungsvariable QUERY_STRING.

http://www

Web-Formular und Perl

```
use CGI ('param');  
  
$anz = param('anzahl');  
@allefelder = param();
```

http://www

altmodisch

```
read(STDIN, $Daten, $ENV{'CONTENT_LENGTH'});  
@Formularfelder = split(/&/, $Daten);  
foreach $Feld (@Formularfelder) {  
    ($name, $value) = split(/=/, $Feld);  
    $value =~ tr/+// ;  
    $value =~ s/%{[a-fA-F0-9]{2}}/pack("C",  
    hex($1))/eg;  
    $Formular{$name} = $value;  
}  
$p = %Formular{'plz'}
```

http://www

Drei Regeln der CGI-Programmierung

- Du sollst **use CGI** verwenden
 - Du sollst **use CGI** verwenden
 - Du sollst **use CGI** verwenden
- Wer es nicht macht verdient **kein Mitleid!** ■

http://www

Noch ein bisschen Perl

- Funktionen: Klammern weglassen
`print("Hallo");`
`print "Hallo";`
- Zuweisung: viele parallel:
`($a, $b) = (10, 20);`
`($a, $b, $c) = localtime();`

http://www

Noch ein bisschen Perl

- Zuweisung: viele parallel:
- `($a, $b) = (10, 20);`
- `($a, $b, $c) = localtime();`
- `@alle = localtime();`
funktion im listen-Kontext aufgerufen
- `$wasanders = localtime();`
funktion im skalaren Kontext aufgerufen.