

# CSP 语言规范

# Language Specification

CSP 1.1

**Copyright©2007-2010 Akee H.D**

作者: Akee Yang [akee.yang@gmail.com](mailto:akee.yang@gmail.com)

(转载请注明本书作者! 未经允许, 严禁把本书用于商业目的!)

# 目录

目录 .....	2
前言 .....	5
第 1 章 CSP入门.....	6
1.1 基础知识.....	6
1.1.1 MYCP .....	6
1.1.2 CSP 文件.....	6
1.1.3 开发流程.....	6
1.1.4 虚拟主机.....	7
1.1.5 CSP例子.....	7
1.2 CSP+应用方向 .....	7
1.3 测试CSP例子.....	7
第 2 章 基本CSP语句.....	9
2.1 注释语句 CSP Comment .....	9
2.2 输出语句 Output.....	9
2.2.1 <%=variable%> .....	9
2.2.2 <csp:write id="" scopy="" /> .....	9
2.2.3 <csp:write name="" property="" /> .....	9
2.2.4 提示.....	10
2.3 例子Example .....	10
第 3 章 变量 CSP Variable .....	11
3.1 CSP变量类型Type.....	11
3.2 变量存放区域scopy .....	11
3.3 变量数据类型DataType .....	12
3.4 定义Define.....	12
3.4.1 定义变量Define Variable.....	12
3.4.2 定义只读变量(常量) Define Constant .....	13
3.4.3 提示.....	13
3.5 例子Example .....	13
第 4 章 比较控制if .....	14
4.1 格式 Format .....	14
4.2 简单比较Simple Compare.....	14
4.3 带else语句if else.....	14
4.4 多级比较Multi-Level Compare.....	14
4.5 全部语句 All csp:if Compare .....	15
4.6 例子Example .....	17
第 5 章 循环while.....	18
5.1 格式Format .....	18
5.2 临时变量Temp Variable.....	18
5.3 全部语句All csp:while Compare.....	18
5.4 例子Example .....	20
第 6 章 foreach循环.....	21
6.1 格式Format .....	21

6.2	临时变量Temp Variable.....	21
6.3	例子Example .....	22
第 7 章	其他控制Other Control .....	23
7.1	退出循环break.....	23
7.1.1	格式Format .....	23
7.2	重新执行循环contine.....	23
7.2.1	格式Format .....	23
第 8 章	CSP运算符Operator .....	25
8.1	加 增加 add.....	25
8.2	减 subtract.....	25
8.3	乘 multiplication .....	25
8.4	除 division.....	25
8.5	加 1 increate .....	25
8.6	减 1 decrease.....	26
8.7	例子Example .....	26
第 9 章	其他变量操作Basic Operate .....	28
9.1	判断变量是否为空empty.....	28
9.2	清空变量的值勤reset .....	28
9.3	获取变量的大小size.....	28
9.4	获取变量的类型type.....	28
9.5	转换变量类型totype.....	28
9.6	获取变量指定序列值index.....	29
9.7	例子Example .....	29
第 10 章	页面操作Page Operate .....	30
10.1	设置页面类型Set ContentType.....	30
10.2	退出当前页面Return Current Page.....	30
10.3	清空输出Reset Output.....	30
10.4	转发跳转Forward .....	30
10.4.1	格式Format .....	30
10.4.2	例子Example.....	30
10.5	重定向跳转Location.....	31
10.5.1	格式Format .....	31
10.5.2	例子Example.....	31
第 11 章	HTTP/HTML .....	32
11.1	HTTP自动验证Authorization.....	32
11.1.1	格式Format .....	32
11.1.2	例子Example.....	32
11.2	HTTP头信息HTTP Header .....	32
11.2.1	系统变量System Variable.....	33
11.2.2	自定义变量Other Header Variable.....	33
11.2.3	例子Example.....	34
11.3	表单参数Parameter.....	35
11.3.1	系统变量System Variable.....	35
11.3.2	自定义变量Other Parameter Variable .....	35

11.3.3	例子Example.....	36
第 12 章	C++ APP .....	37
12.1	组件交互概述.....	37
12.1.1	执行应用组件函数.....	37
12.1.2	调用应用组件服务接口方法.....	37
12.1.3	管理应用组件服务接口参数.....	37
12.2	执行组件函数Execute APP Function .....	38
12.2.1	格式Format .....	38
12.2.2	例子Example.....	38
12.3	调用组件服务接口方法Call APP Function .....	38
12.3.1	定义组件实例Define APP Variable.....	38
12.3.2	初始化组件Init APP .....	39
12.3.3	注销退出组件Final APP .....	39
12.3.4	调用组件服务接口方法APP Call.....	39
12.4	管理应用组件接口参数APP Property .....	39
12.4.1	获取组件参数Get APP Property .....	39
12.4.2	设置组件参数Set APP Property .....	39
12.4.3	添加组件参数Add APP Property .....	40
12.4.4	删除组件参数Delete APP Property.....	40
12.4.5	获取组件信息Get APP Info .....	40
12.5	例子Example .....	41
第 13 章	CDBC.....	42
13.1	CDBC使用概述.....	42
13.2	定义CDBC实例Define CDBC Variable.....	42
13.3	更新操作Execute SQL .....	42
13.4	查询操作Select SQL .....	42
13.5	获取第一条记录Get First.....	43
13.6	获取下一条记录Get Next .....	43
13.7	获取上一条记录Get Previous .....	43
13.8	获取最后一条记录Get Last .....	43
13.9	清空记录集Reset Result.....	43
13.10	获取记录集大小Result Size.....	43
13.11	获取当前记录索引Result Index.....	43
13.12	例子Example .....	44
第 14 章	附录A系统变量System Variable.....	46

# 前言

CSP (C++ Server Pages) 是一套基于 C++ 的服务端页面开发技术，是基于 MYCP 平台的一套 web 应用开发标签语言；MYCP 服务平台利用标准 C++ 语言，实现 HTTP 服务器和 C++ web 容器功能，配合 CSP+Servlet+APP+CDBC (以下简称 CSP+)。

利用 C++ 语言的强大功能，“JAVA/J2EE 能做的，CSP+，做的更好！”

CSP 语法是标准 XML 格式标签，CSP 帮助开发人员严格按照 MVC (模型-视图-控制器) 框架进行 web 开发，通过在 HTML 文件嵌套标准 CSP 标签，调用业务功能组件；

CSP 语法简单，普通开发人员，不用半小时可以学习完成，而熟悉 JSP、PHP 等其他页面开发技术，十分钟足够；

最新 CSP 1.1 规范可以查看以下地址：

[http://code.google.com/p/mycp/wiki/CSP\\_1\\_1](http://code.google.com/p/mycp/wiki/CSP_1_1)

# 第1章 CSP 入门

CSP 是标准 XML 格式标签脚本语言，通过在 HTML 增加 CSP 标签，实现页面输出、业务处理、页面流转等服务端功能；

CSP 帮助开发人员，严格执行 MVC 开发模式，从而让业务系统的 UI 和业务逻辑分开，让开发团队角色更加明确；

在大型企业 web 应用系统中，利用 MVC 框架，将极大提高系统的开发效率和和系统可维护性。

## 1.1 基础知识

### 1.1.1 MYCP

MYCP 是一套集 HTTP 服务器、C++ web (CSP+) 容器功能于一身的 web 应用开发平台；MYCP 采用标准 C++语言编写，集合所有 C++语言的优势，高效率、稳定、功能强大。

CSP 文件及其他 CSP+组件部署到 MYCP 平台提供 web 服务。

### 1.1.2 CSP 文件

统一使用 .csp （小写）扩展名，文件内容使用标准 HTML 语言+CSP 标签语句，在某些环境中，可能使用 XML+CSP 标签，比如 ajax。

### 1.1.3 开发流程

- 需要先确定需求；
- 然后根据系统功能，进行划分，设计出需要多少 CSP 或者 Servlet 页面；
- 列出对应的业务处理模型（C++ APP 应用组件）；
- 如果需要数据库功能，利用已经有的 CDBC 驱动程序操作数据库系统。
- 根据设计后，对开发人员进行分工，UI 人员设计编写 HTML 页面，其他人员开发 C++ APP，或者 C++ Servlet；

- HTML 经过简单修改，增加相应 CSP 标签，连接 APP 应用组件；
- 测试、修改、部署发布、更新完善；

### 1.1.4 虚拟主机

MYCP CSP 支持虚拟主机技术，可以实现指定 IP 地址、端口、域名、主机名，或者任何二种之间的组合来实现不同 Web 系统；

### 1.1.5 CSP 例子

下载最新版本的 MYCP，在 bin/web/samples 目录下有所有的 CSP 源码例子；CSP 源码例子基本展现了所有 CSP 语言规范和特性，参考学习 CSP 例子，是最好的学习 CSP 方法。

## 1.2 CSP+应用方向

CSP+技术可以应用于以下方向：

- 普通公司网站
- 企业 Web 应用开发
- 大型分布式应用系统
- 异构系统通讯应用（例如 C/S 跟 B/S 系统，或者 web 系统调用 C++算法等）
- TV 端（类似路由器、电视盒等硬件）WEB 服务器
- .....

## 1.3 测试 CSP 例子

直接启动 CGCP 应用服务，然后打开浏览器，输入访问 CSP 地址 <http://127.0.0.1:81>，如下图：

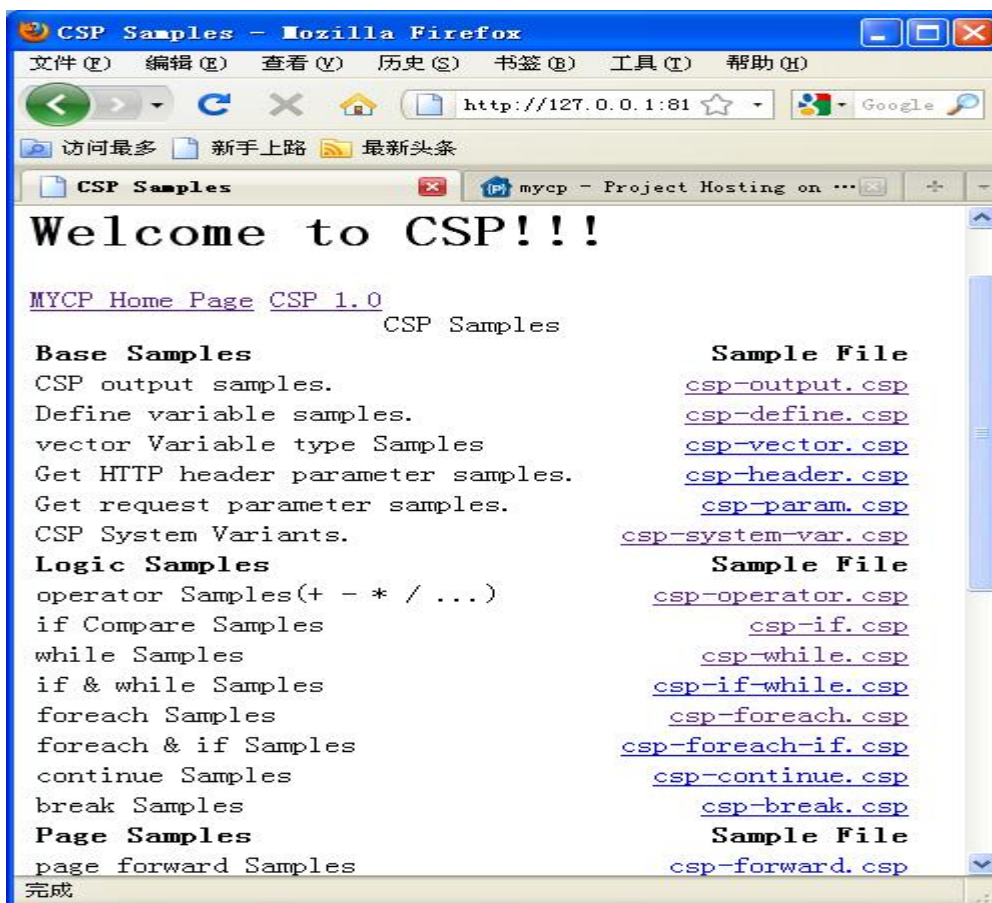
```

C:\D:\WorkRecord\googlecode\mycp\trunk\bin\win32\CGCP.exe
Starting CGCP0 Service.....
SystemParams = 0
CDBCInfos = 2
DataSources = 2
MODULES = 12
CGCP0: [INFO] MODULE 'LogService' load succeeded
CGCP0: [INFO] MODULE 'BodbService' load succeeded
CGCP0: [INFO] MODULE 'DLLTest' load succeeded
CGCP0: [INFO] MODULE 'FileSystemService' load succeeded
CGCP0: [INFO] MODULE 'HttpServer' load succeeded
CGCP0: [INFO] MODULE 'ParserHttp' load succeeded
CGCP0: [INFO] MODULE 'ParserSotp' load succeeded
CGCP0: [INFO] MODULE 'cspApp' load succeeded
CGCP0: [INFO] MODULE 'cspServlet' load succeeded
CommTcpServer: [INFO] **** [TCPSERVER:81] Start succeeded ****
CGCP0: [INFO] MODULE 'CommTcpServer' load succeeded
CommUdpServer: [INFO] **** [UDPSERVER:8012] Start succeeded ****
CGCP0: [INFO] MODULE 'CommUdpServer' load succeeded
CommTcpServer: [INFO] **** [TCPSERVER:82] Start succeeded ****
CGCP0: [INFO] MODULE 'CommTcpServer' load succeeded
CGCP0: [INFO] **** CGCP0 Server start succeeded ****

***** App Help *****
help          Print this help.
start         Start MYCP Service.
stop          Stop MYCP Service.
restart       Restart MYCP Service.
exit          Exit MYCP Server.

CMD:

```





## 第2章 基本 CSP 语句

### 2.1 注释语句 CSP Comment

`<!-- Comment 单行注释 -->`

`<!--`

`Comment 多行注释`

`-->`

### 2.2 输出语句 Output

#### 2.2.1 `<%=variable%>`

`<%=variable%>`

variable	可以是用户变量 <code>\$variable</code> ，临时变量 <code>_\$variable</code> ，也可以是普通字符串 <code>"variable"</code> ；（以下同）
----------	--

例子 Sample: `<%=$var1%>`

例子 Sample: `<%=P$UserName%>`

#### 2.2.2 `<csp:write id="" scopy="" />`

`<csp:write id="" scopy="" />`

打印 id 的值；

id	可以是用户变量 <code>\$variable</code> ，临时变量 <code>_\$variable</code> ，也可以是普通字符串 <code>"variable"</code> ；（以下同）
scopy	如果是用户变量，指定变量存放区域；默认不填为 page 区域；

#### 2.2.3 `<csp:write name="" property="" />`

`<csp:write name="" property="" />`

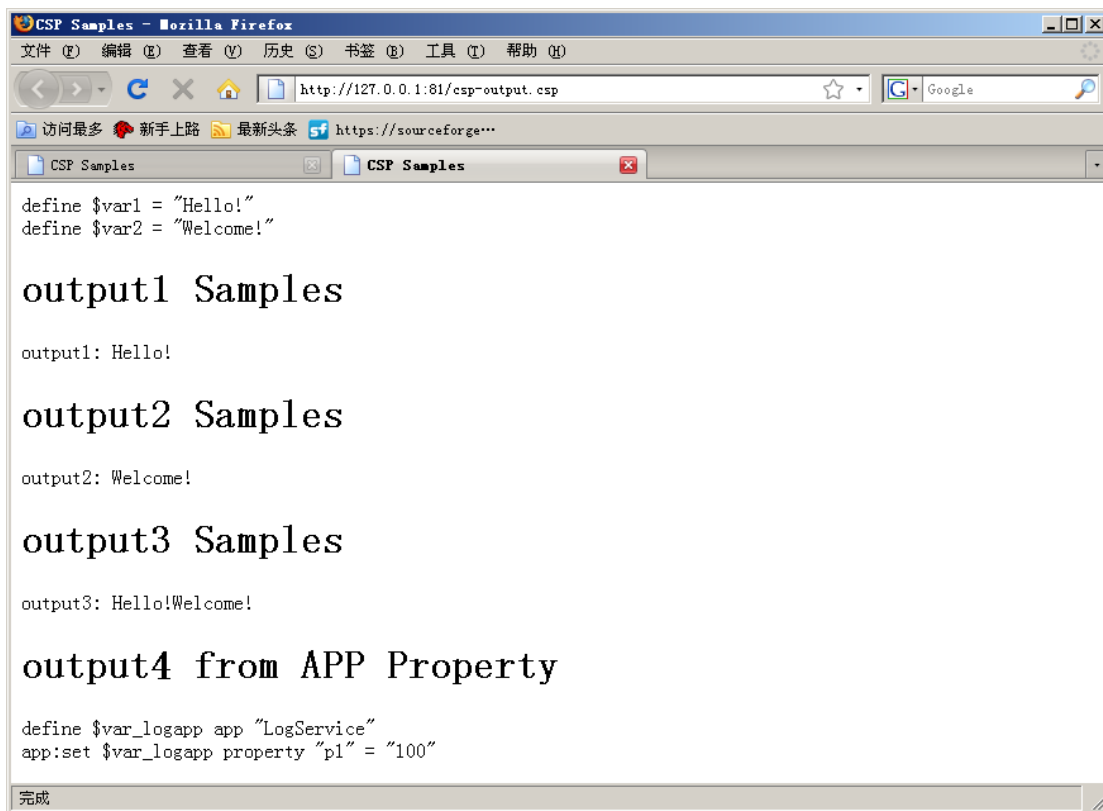
打印 C++ APP 应用组件，指定属性的参数值。

name	APP 应用组件名称
property	属性名称

## 2.2.4 提示

- `<%=>`可以存在于 CSP 文件内任何位置，用于打印指定信息；不包含在 `<%-- -->`注释语句内；
- `<csp:write />`语句必须占满一行位置；
- 可以组合使用，如：`<%= $var1%><csp:write id="$var2" />`，中间有允许有其他字符；

## 2.3 例子 Example



```
define $var1 = "Hello!"
define $var2 = "Welcome!"

output1 Samples

output1: Hello!

output2 Samples

output2: Welcome!

output3 Samples

output3: Hello!Welcome!

output4 from APP Property

define $var_logapp app "LogService"
app:set $var_logapp property "p1" = "100"
```

## 第3章 变量 CSP Variable

### 3.1 CSP 变量类型 Type

CSP 包括有以下变量类型：

变量类型	定义头	描述
用户变量	\$	例如\$var_user
系统变量	S\$	用于获取系统信息，请看《附录系统变量表》 例如 S\$ ParamNames、S\$RequestURL
PARAMETER 变量	P\$	用于获取用户表单信息； 例如 P\$UserName、P\$UserAccount
HEADER 变量	H\$	用于获取 HTTP 头信息；后面由用户自己确定； 例如 H\$Content-Type
临时变量	_\$	请看《附录临时变量表》，详细看各 CSP 标签使用； 例如_\$index、_\$value、_\$result
应用组件变量	A\$	用于访问 C++ APP 应用组件，通过 apps.xml 配置；
数据源变量	D\$	用于访问数据源接口，通过 datasources.xml 配置；

### 3.2 变量存放区域 scopy

CSP 变量可以存放于以下不同区域：

page	页面区域，对当前页面有效；默认存放区域
request	请求区域，当前 request 有效，可以用于 forward
session	会话区域，当前整个 SESSION 周期有效
application	应用区域，当前应用（虚拟主机内）有效
server	WEB SERVER 区域，当前 HTTP SERVER 下所有应用有效

### 3.3 变量数据类型 DataType

CSP 支持以下数据类型：

int	C++ int 数据类型
bitint	C++ __int64/long long 数据类型
time	C++ time_t 数据类型
float	C++ double 数据类型
boolean	C++ bool 数据类型
string	C++ std::string 数据类型； <b>默认数据类型</b>
vector	C++ std::vector 数据类型，可以包含以上基础数据类型

### 3.4 定义 Define

#### 3.4.1 定义变量 Define Variable

```
<csp:define id="var_name" value="var_value" />
```

定义变量 var\_name，初始化"var\_value"，默认使用字符串类型；

```
<csp:define id="var_name" type="var_type" value="var_value" />
```

定义变量 var\_name，初始化"var\_value"，指定 type 类型；

```
<csp:define id="var_name" name="app_name" property="app_property" />
```

定义变量 var\_name，初始化值，为 app\_name 应用的 app\_property 参数值；

**例子 Sample:**

```
<csp:define id="$var1" value="100" />
```

定义字符串类型变量\$var1，值为"100"；

```
<csp:define id="$var1" type="int" value="100" />
```

定义整数类型变量\$var1，值为 100；

### 3.4.2 定义只读变量(常量) Define Constant

```
<csp:define id="cons_name" value="var_value" property="const" />
```

定义常量 `cons_name`，该常量值初始化后，不可更改；

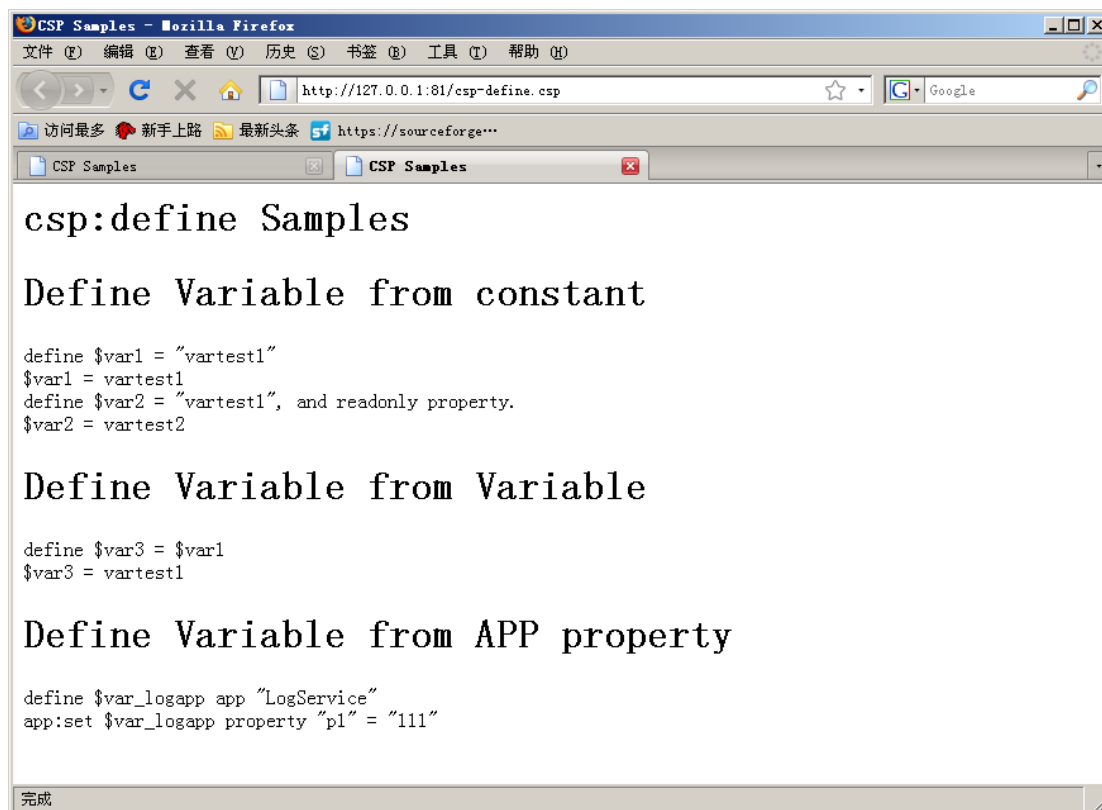
例子 Sample :

```
<csp:define id="$max_second" value="1000" type="int" property="const" />
```

### 3.4.3 提示

- 用户定义变量必须以\$开头，如`$var1`，依此类推；
- CSP 支持首先使用定义变量功能，即不用通过定义，直接使用指定变量名称保存；

## 3.5 例子 Example



## 第4章 比较控制 if

CSP 的 if 类似 C++的 if 语句，支持类似 if elseif else 多级控制。

### 4.1 格式 Format

```
<csp:if:equal id="var_name" value="compare_value" />
<csp:if:equal name="app_name" property="app_property" value="compare_value"
/>
<csp:if:equal name="app_name" property="app_property" id="compare_id" />
<csp:if:equal id="compare_id" name="app_name" property="app_property" />
```

### 4.2 简单比较 Simple Compare

```
<csp:if:equal id="var_name" value="compare_value" />
...
<csp:end>
```

true: 执行...语句，直到 end 语句结束。

### 4.3 带 else 语句 if else

```
<csp:if:equal id="var_name" value="compare_value" />
true: ...
<csp:if:else>
else: ...
<csp:end>
```

true: 执行 true: ... 语句

false: 执行 else: ... 语句

### 4.4 多级比较 Multi-Level Compare

```
<csp:if:equal id="var_name" value="compare_value" />
...
<csp:if:equal id="var_name" value="compare_value" />
...
```

```
<csp:if:else>  
...  
<csp:end>
```

从上往下执行，比较成功，执行对应... 语句后，退出<csp:end>

## 4.5 全部语句 All csp:if Compare

```
<csp:if:equal id="var_name" value="compare_value" />
```

判断是否相等 Compare if equal.

```
<csp:if:notEqual id="var_name" value="compare_value" />
```

判断是否不相等 Compare if not equal

```
<csp:if:greater id="var_name" value="compare_value" />
```

判断是否大于 Compare if greater.

```
<csp:if:greaterEqual id="var_name" value="compare_value" />
```

判断是否大于或者等于 Compare if greater or equal.

```
<csp:if:less id="var_name" value="compare_value" />
```

判断是否小于 Compare if less.

```
<csp:if:lessEqual id="var_name" value="compare_value" />
```

判断是否小于或者等于 Compare if less or equal.

```
<csp:if:empty id="var_name" />
```

判断是否为空 Compare variable if empty.

```
<csp:if:notEmpty id="var_name" />
```

判断是否不为空 Compare variable if not empty.

```
<csp:if:exist id="var_name" />
```

判断变量是否存在 Compare variable if exist.

```
<csp:if:notExist id="var_name" />
```

判断变量是否不存在 Compare variable if not exist.

```
<csp:if:else>
```

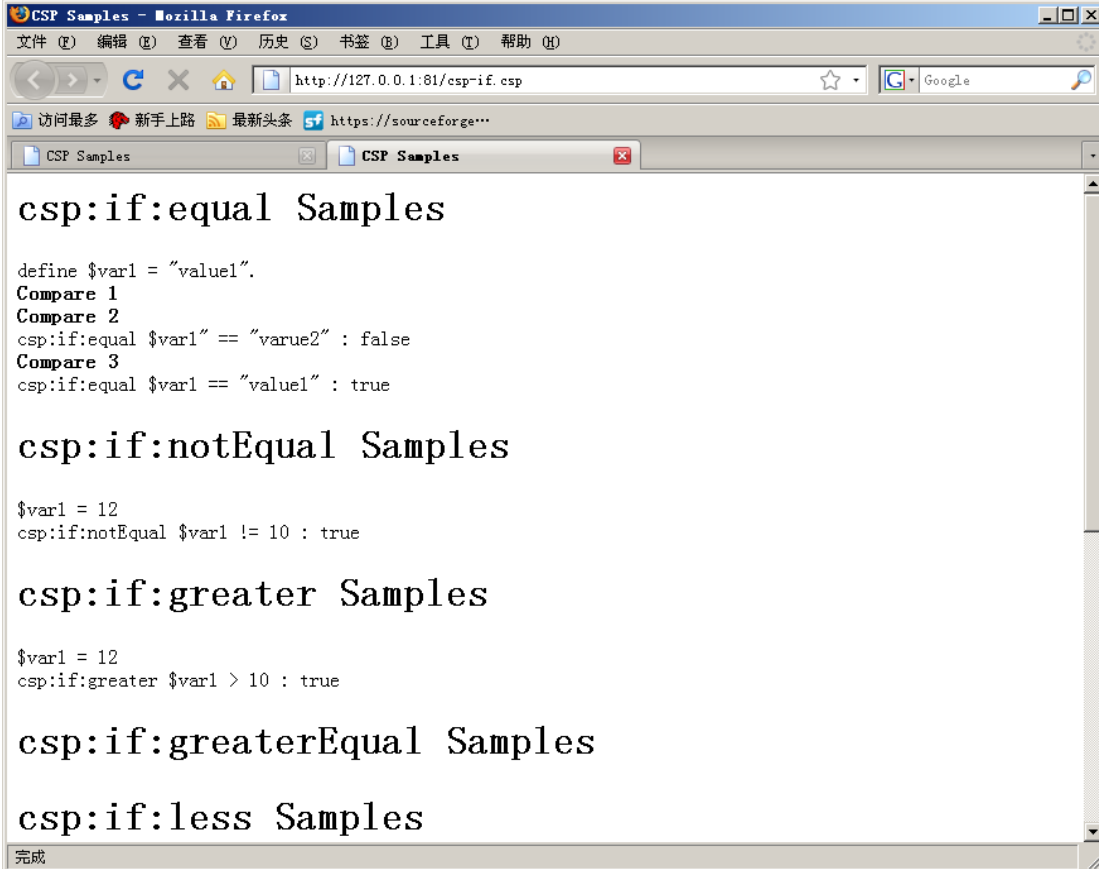
判断为 false, 执行语句 Compare all else.

```
<csp:end>
```

结束语句 end.



## 4.6 例子 Example



```
csp:if:equal Samples

define $var1 = "value1".
Compare 1
Compare 2
csp:if:equal $var1 == "varue2" : false
Compare 3
csp:if:equal $var1 == "value1" : true

csp:if:notEqual Samples

$var1 = 12
csp:if:notEqual $var1 != 10 : true

csp:if:greater Samples

$var1 = 12
csp:if:greater $var1 > 10 : true

csp:if:greaterEqual Samples

csp:if:less Samples
```

完成

## 第5章 循环 while

csp:while 跟 csp:if 语句完全一样比较方式。

### 5.1 格式 Format

```
<csp:while:equal id="var_name" value="compare_value" />
...
<csp:end>
```

循环比较变量 id 跟值 value 的值；

true: 执行... 语句，直到 false，退出<csp:end>；

### 5.2 临时变量 Temp Variable

\$_index	保存当前索引序列数，从 0 开始，while 每循环一次，\$_index 加 1；为避免进行死循环，最大可以执行 1000 次
----------	--

### 5.3 全部语句 All csp:while Compare

```
<csp:while:equal id="var_name" value="compare_value" />
```

循环判断是否相等

```
<csp:while:notEqual id="var_name" value="compare_value" />
```

循环判断是否不相等

```
<csp:while:greater id="var_name" value="compare_value" />
```

循环判断是否大于

```
<csp:while:greaterEqual id="var_name" value="compare_value" />
```

循环判断是否大于或者等于

```
<csp:while:less id="var_name" value="compare_value" />
```

循环判断是否小于

```
<csp:while:lessEqual id="var_name" value="compare_value" />
```

循环判断是否小于或者等于

```
<csp:while:empty id="var_name" />
```

循环判断是否为空

```
<csp:while:notEmpty id="var_name" />
```

循环判断是否不为空

```
<csp:while:exist id="var_name" />
```

循环判断是否存在变量

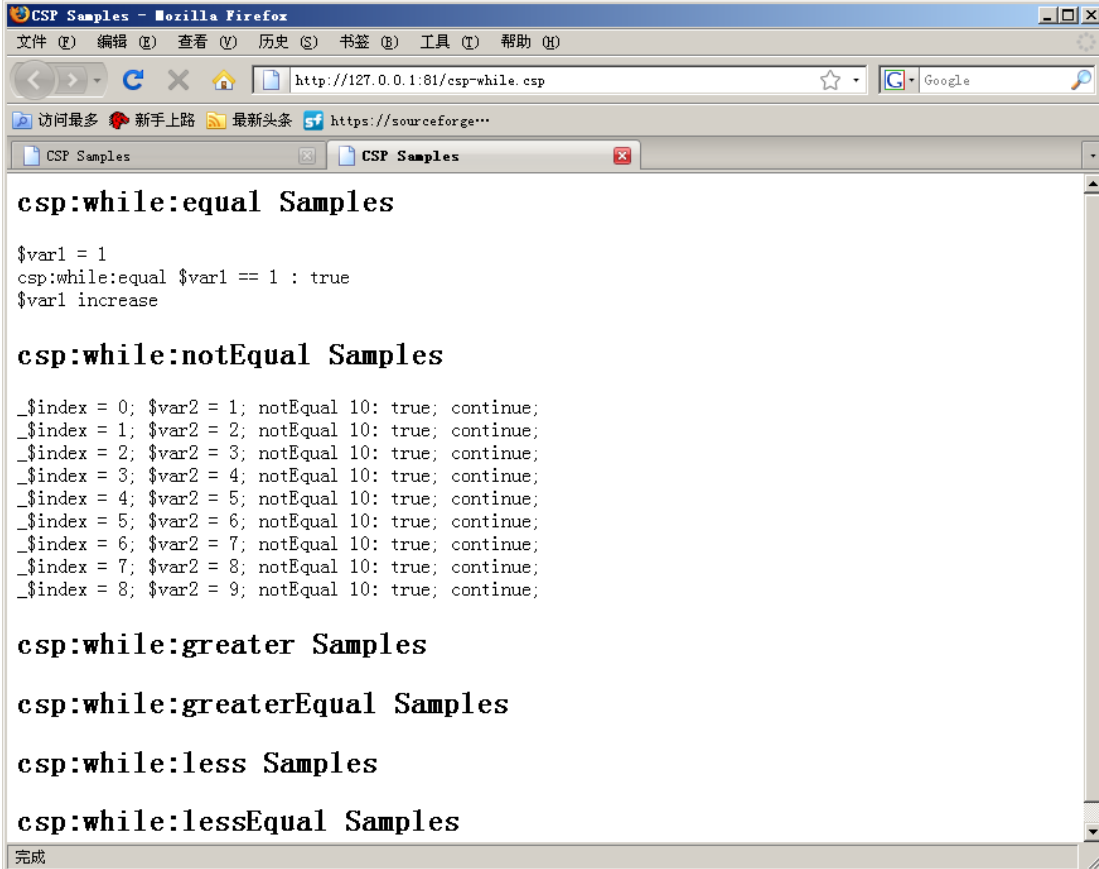
```
<csp:while:notExist id="var_name" />
```

循环判断是否不存在变量

```
<csp:end>
```

结束语句

## 5.4 例子 Example



The screenshot shows a Mozilla Firefox browser window with the title "CSP Samples - Mozilla Firefox". The address bar shows "http://127.0.0.1:81/csp-while.csp". The browser has two tabs, both titled "CSP Samples". The main content area displays the following CSP code examples:

```
csp:while:equal Samples  
  
$var1 = 1  
csp:while:equal $var1 == 1 : true  
$var1 increase  
  
csp:while:notEqual Samples  
  
_ $index = 0; $var2 = 1; notEqual 10: true; continue;  
_ $index = 1; $var2 = 2; notEqual 10: true; continue;  
_ $index = 2; $var2 = 3; notEqual 10: true; continue;  
_ $index = 3; $var2 = 4; notEqual 10: true; continue;  
_ $index = 4; $var2 = 5; notEqual 10: true; continue;  
_ $index = 5; $var2 = 6; notEqual 10: true; continue;  
_ $index = 6; $var2 = 7; notEqual 10: true; continue;  
_ $index = 7; $var2 = 8; notEqual 10: true; continue;  
_ $index = 8; $var2 = 9; notEqual 10: true; continue;  
  
csp:while:greater Samples  
  
csp:while:greaterEqual Samples  
  
csp:while:less Samples  
  
csp:while:lessEqual Samples
```

The status bar at the bottom of the browser window shows "完成".

## 第6章 foreach 循环

foreach 用于处理 VECTOR 变量。

### 6.1 格式 Format

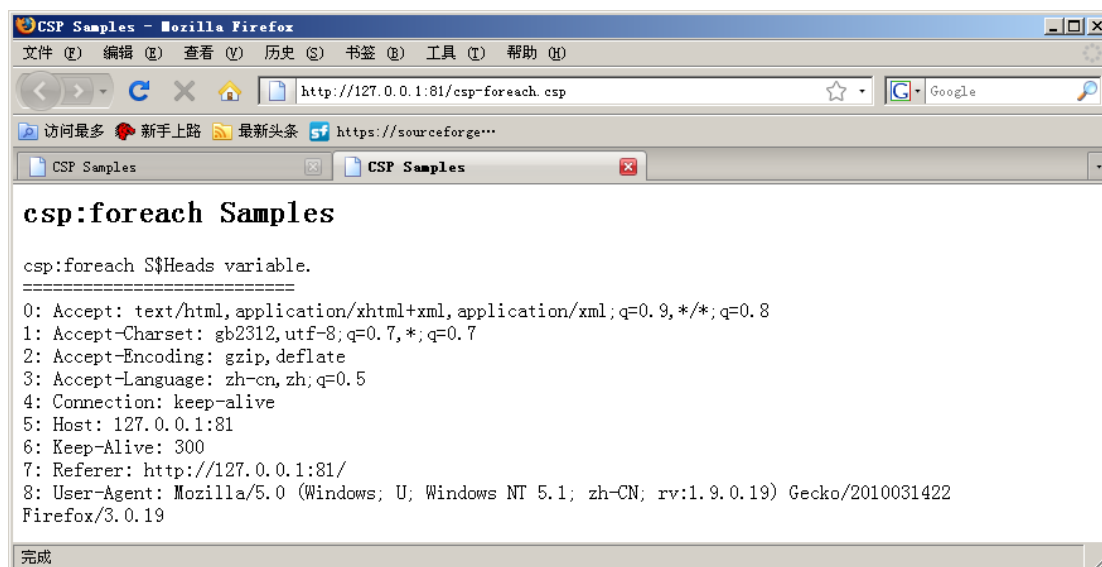
```
<csp:foreach id="var_vector" />
...
<csp:end>
```

循环处理类型 VECTOR 变量 id 的值；同时执行... 语句

### 6.2 临时变量 Temp Variable

_\$index	当前索引序列 Current Index Variable
_\$value	当前索引值 Current Value Variable
_\$filename	文件名称 用于<csp:foreach id="\$UploadFiles" />; (以下同)
_\$filepath	文件保存路径
_\$filesize	文件大小
_\$filetype	文件类型

## 6.3 例子 Example



## 第7章 其他控制 Other Control

### 7.1 退出循环 break

退出循环语句，用于 `csp:while` 和 `csp:foreach` 循环。

#### 7.1.1 格式 Format

```
<csp:break id="var_name" value="compare_value" />
```

如果 `id` 等于 `value` 值，退出循环；

```
<csp:break name="app_name" property="app_property" value="compare_value" />
```

如果应用组件的参数等于 `value` 值，退出循环；

```
<csp:break name="app_name" property="app_property" id="compare_id" />
```

```
<csp:break id="compare_id" name="app_name" property="app_property" />
```

如果应用组件参数等于 `id` 值，退出循环；

### 7.2 重新执行循环 contine

重新执行循环语句，用于 `csp:while` 和 `csp:foreah` 循环，跳过后面的语句，直接重新执行循环。

#### 7.2.1 格式 Format

```
<csp:continue id="var_name" value="value" />
```

如果 `id` 等于 `value`，重新执行循环；

```
<csp:continue name="app_name" property="app_property" value="value" />
```

如果应用组件参数等于 `value` 值，重新执行循环；

```
<csp:continue name="app_name" property="app_property" id="compare_id" />  
<csp:continue id="compare_id" name="app_name" property="app_property" />
```

如果参数等于 id 值，重新执行循环；



## 第8章 CSP 运算符 Operator

### 8.1 加 增加 add

```
<csp:+= id="" value="" />
```

id 变量加上 value 的值;

string: 添加字符串到末尾

vector: 添加元素到末尾

### 8.2 减 subtract

```
<csp:-= id="" value="" />
```

id 变量减去 value 的值;

### 8.3 乘 multiplication

```
<csp:*= id="" value="" />
```

id 变量乘以 value 的值;

### 8.4 除 division

```
<csp:/= id="" value="" />
```

id 变量除以 value 的值;

### 8.5 加 1 increase

```
<csp:increase id="" />
```

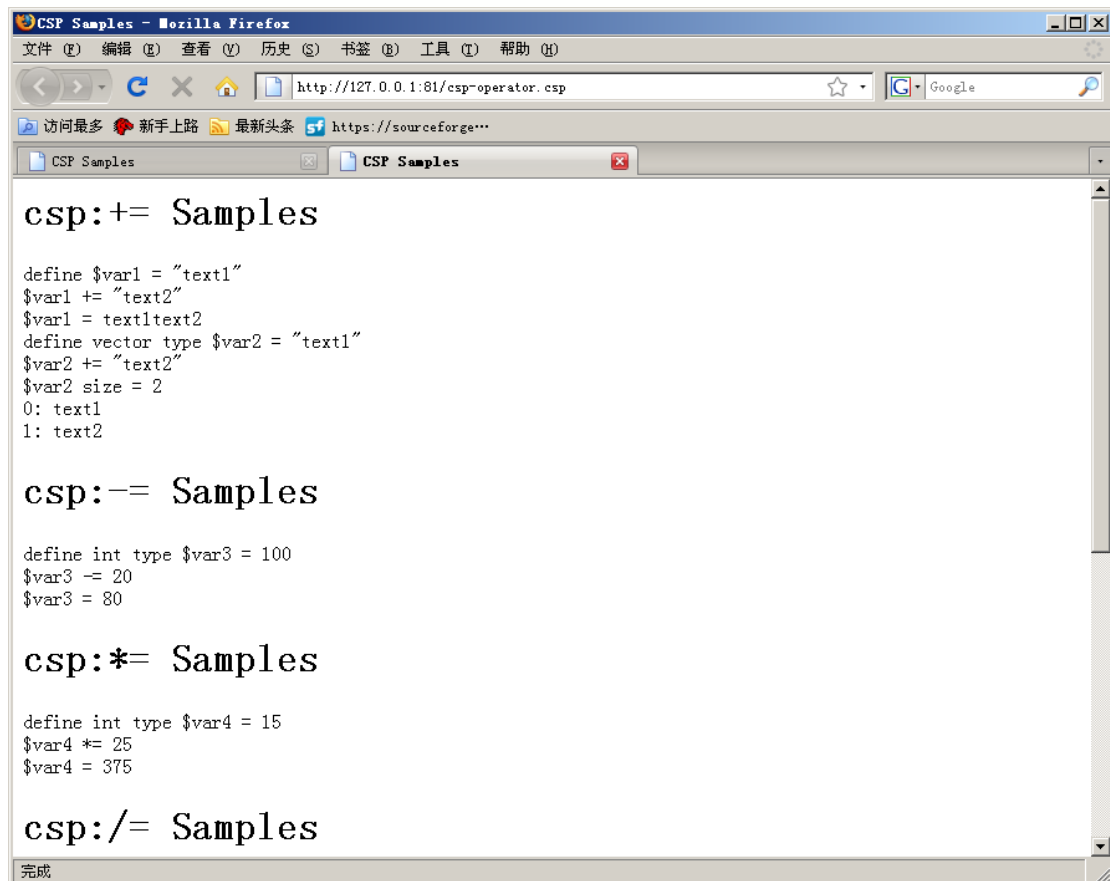
id 变量值加 1, 变量必须为 int 整数类型, 非整数类型转换为整数类型;

## 8.6 减 1 decrease

```
<csp:decrease id="" />
```

id 变量值减 1，变量必须为 int 整数类型，非整数类型转换为整数类型；

## 8.7 例子 Example



The screenshot shows a Mozilla Firefox browser window with the address bar at `http://127.0.0.1:81/csp-operator.csp`. The page content displays four examples of CSP operators: `csp:+= Samples`, `csp:-= Samples`, `csp:*= Samples`, and `csp:/= Samples`. Each example is followed by its corresponding code snippet.

```
csp:+= Samples

define $var1 = "text1"
$var1 += "text2"
$var1 = text1text2
define vector type $var2 = "text1"
$var2 += "text2"
$var2 size = 2
0: text1
1: text2

csp:-= Samples

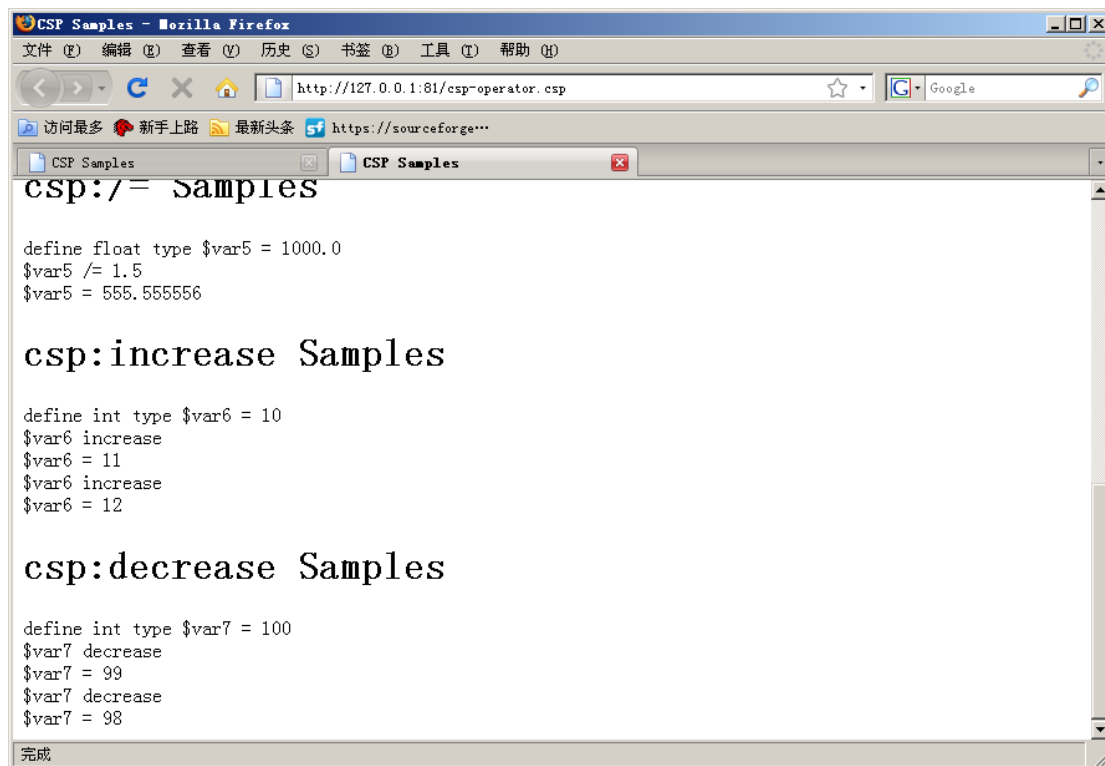
define int type $var3 = 100
$var3 -= 20
$var3 = 80

csp:*= Samples

define int type $var4 = 15
$var4 *= 25
$var4 = 375

csp:/= Samples
```

完成



```
csp:/= samples

define float type $var5 = 1000.0
$var5 /= 1.5
$var5 = 555.555556

csp:increase Samples

define int type $var6 = 10
$var6 increase
$var6 = 11
$var6 increase
$var6 = 12

csp:decrease Samples

define int type $var7 = 100
$var7 decrease
$var7 = 99
$var7 decrease
$var7 = 98
```

## 第9章 其他变量操作 Basic Operate

### 9.1 判断变量是否为空 empty

```
<csp:empty id="" out="var_saveto" />
```

判断 id 变量是否为空，放到 value 变量中；id 变量不存在，返回 true；

### 9.2 清空变量的值 reset

```
<csp:reset id="" scopy="" />
```

清空 id 变量的值；

### 9.3 获取变量的大小 size

```
<csp:size id="var_name" />
```

获取 id 变量的 size，放到 `_%size` 临时变量中；比如 VECTOR 的 size 大小、string 的长度等；id 变量不存在，返回-1；

### 9.4 获取变量的类型 type

```
<csp:type id="var_name" out="var_saveto" />
```

获取 id 变量类型信息，放到 out 变量中；类型信息如下，id 变量不存在，返回 null；

int, bitint, time, boolean, float, string, vector, app

### 9.5 转换变量类型 totype

```
<csp:totype id="" type="" />
```

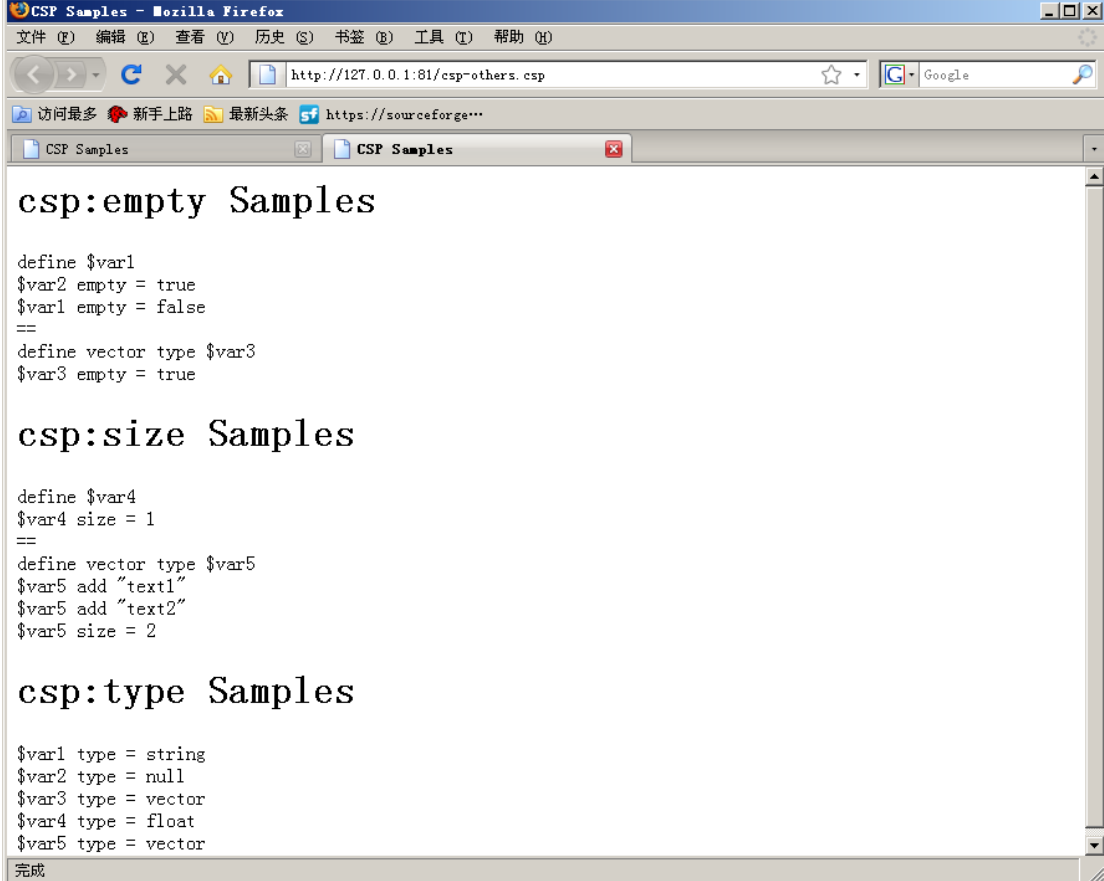
把 id 变量转换为 type 指定类型；

## 9.6 获取变量指定序列值 index

```
<csp:index id="" index="" out="" />
```

获取 id 变量的指定 index 值, 存放到 out 用户变量中;

## 9.7 例子 Example



```
csp:empty Samples

define $var1
$var2 empty = true
$var1 empty = false
==
define vector type $var3
$var3 empty = true

csp:size Samples

define $var4
$var4 size = 1
==
define vector type $var5
$var5 add "text1"
$var5 add "text2"
$var5 size = 2

csp:type Samples

$var1 type = string
$var2 type = null
$var3 type = vector
$var4 type = float
$var5 type = vector

完成
```

## 第10章 页面操作 Page Operate

### 10.1 设置页面类型 Set ContentType

```
<page:contentType type="" />
```

设置当前页面的 ContentType 内容，默认为“text/html”

### 10.2 退出当前页面 Return Current Page

```
<page:return>
```

退出当前页面，不执行后面操作

### 10.3 清空输出 Reset Output

```
<page:reset>
```

清空页面输出内容；

### 10.4 转发跳转 Forward

#### 10.4.1 格式 Format

```
<page:forward url="" />
```

转发跳转操作；执行该操作，会自动退出该页面后面语句；

url	转发的新 URL 地址
-----	-------------

#### 10.4.2 例子 Example

CSP 转发可以包含参数，例如 Sample：

```
<page:forward url="/index.csp" />
```

```
<page:forward url="/index.csp?a=b" />
```

## 10.5 重定向跳转 Location

### 10.5.1 格式 Format

```
<page:location url="" property="" />
```

重定向跳转操作；执行该操作，会自动退出该页面后面语句；

url	重定向的新 URL；该 URL 将显示中客户浏览器地址栏上； Location URL address。
property	重定向状态；301 ： 永久重定向；302 ： 临时重定向；默认不 填为 301。

### 10.5.2 例子 Example

以下为重定向跳转例子 Sample：

```
<page:location url="http://www.google.com" />
```

```
<page:location url="/index.csp" />
```

# 第11章 HTTP/HTML

## 11.1 HTTP 自动验证 Authorization

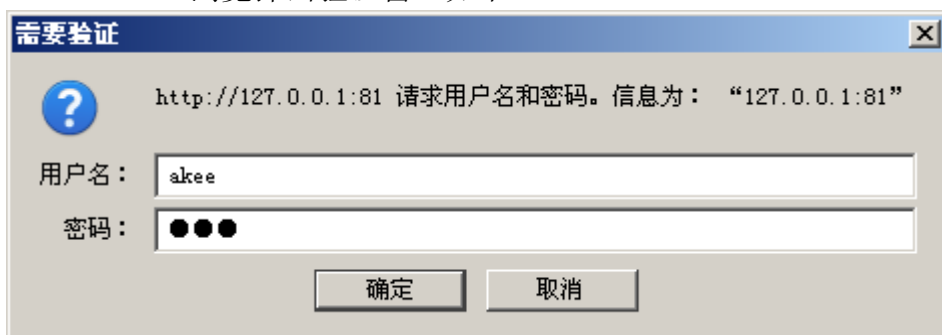
### 11.1.1 格式 Format

<csp:authenticate>

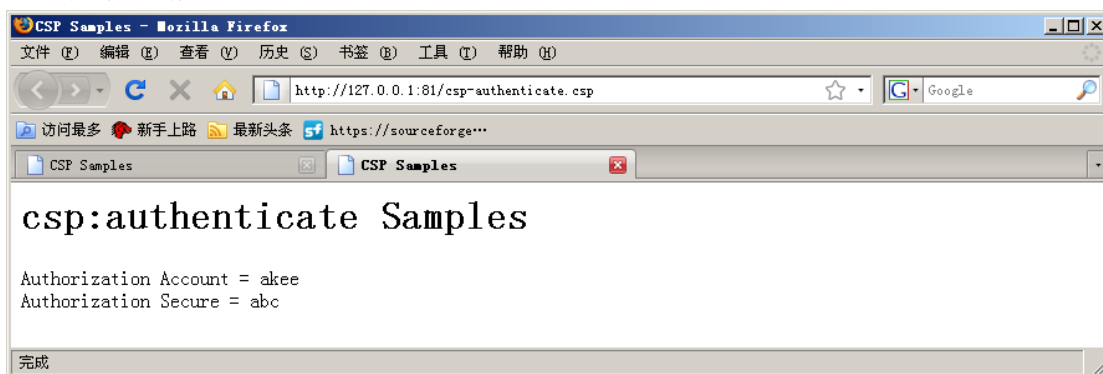
自动请求浏览器，验证用户帐号信息；

### 11.1.2 例子 Example

Firefox 浏览弹出验证窗口如下：



CSP 后台通过 S\$AuthAccount 和 S\$AuthSecure 二个系统变量可以获得验证用户帐号信息：



## 11.2 HTTP 头信息 HTTP Header

用于获取浏览器相关 HTTP 协议头信息。



## 11.2.1 系统变量 System Variable

变量名称	数据类型	描述
S\$HeadNames	vector	所有 HEADER 名称 All Header Name
S\$HeadValues	vector	所有 HEADER 值 All Header Value
S\$Heads	vector	所有 HEADER 名称值 格式 Format: HEADER_NAME: VALUE

## 11.2.2 自定义变量 Other Header Variable

格式 Format : H\$HEADER\_NAME

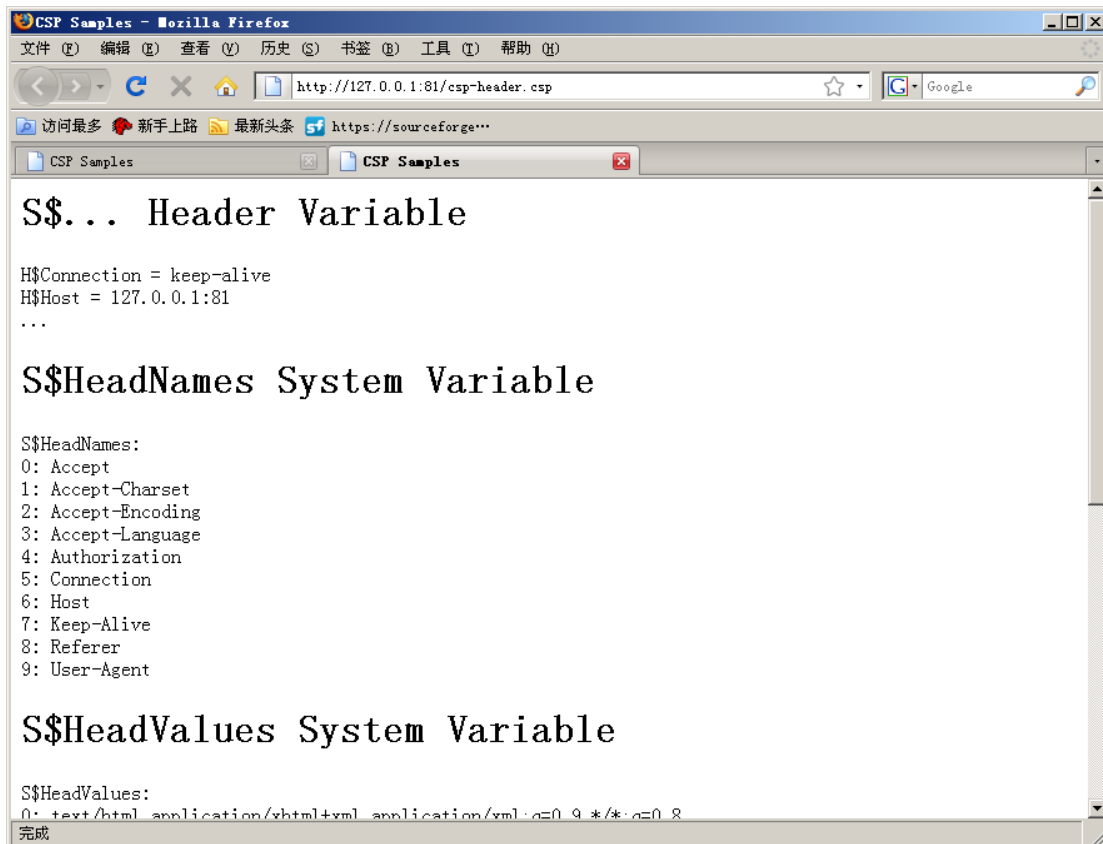
例子 Sample:

H\$Connection

H\$Content-Type

H\$...

## 11.2.3 例子 Example



CSP Samples - Mozilla Firefox

文件(F) 编辑(E) 查看(V) 历史(S) 书签(B) 工具(T) 帮助(H)

http://127.0.0.1:81/csp-header.csp

访问最多 新手上路 最新头条 https://sourceforge...

CSP Samples CSP Samples

### S\$.... Header Variable

```
H$Connection = keep-alive
H$Host = 127.0.0.1:81
...
```

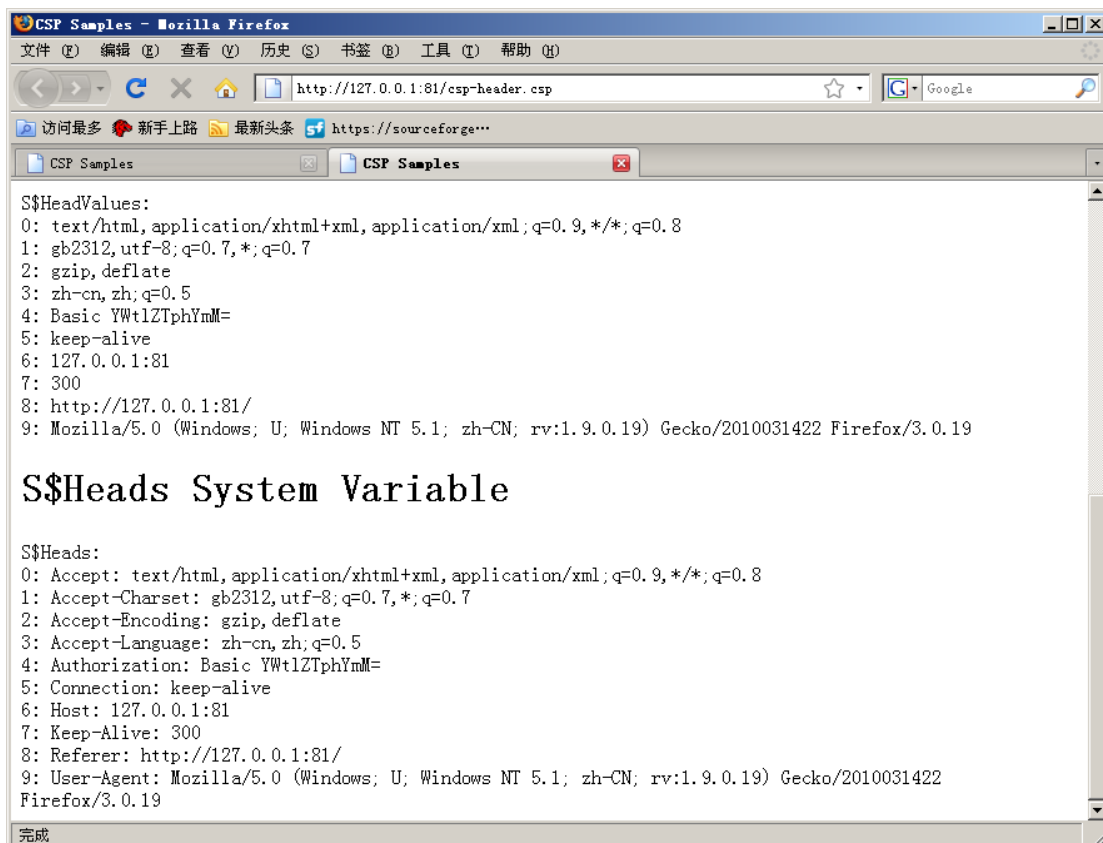
### S\$HeadNames System Variable

```
S$HeadNames:
0: Accept
1: Accept-Charset
2: Accept-Encoding
3: Accept-Language
4: Authorization
5: Connection
6: Host
7: Keep-Alive
8: Referer
9: User-Agent
```

### S\$HeadValues System Variable

```
S$HeadValues:
0: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
```

完成



CSP Samples - Mozilla Firefox

文件(F) 编辑(E) 查看(V) 历史(S) 书签(B) 工具(T) 帮助(H)

http://127.0.0.1:81/csp-header.csp

访问最多 新手上路 最新头条 https://sourceforge...

CSP Samples CSP Samples

```
S$HeadValues:
0: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
1: gb2312,utf-8;q=0.7,*;q=0.7
2: gzip,deflate
3: zh-cn,zh;q=0.5
4: Basic YWt1ZTphYmM=
5: keep-alive
6: 127.0.0.1:81
7: 300
8: http://127.0.0.1:81/
9: Mozilla/5.0 (Windows; U; Windows NT 5.1; zh-CN; rv:1.9.0.19) Gecko/2010031422 Firefox/3.0.19
```

### S\$Heads System Variable

```
S$Heads:
0: Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
1: Accept-Charset: gb2312,utf-8;q=0.7,*;q=0.7
2: Accept-Encoding: gzip,deflate
3: Accept-Language: zh-cn,zh;q=0.5
4: Authorization: Basic YWt1ZTphYmM=
5: Connection: keep-alive
6: Host: 127.0.0.1:81
7: Keep-Alive: 300
8: Referer: http://127.0.0.1:81/
9: User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; zh-CN; rv:1.9.0.19) Gecko/2010031422 Firefox/3.0.19
```

完成

## 11.3 表单参数 Parameter

用于获取 HTML 表单参数信息。

### 11.3.1 系统变量 System Variable

变量名称	数据类型	描述
S\$ParamNames	vector	所有参数名称; All Parameter Name
S\$ParamValues	vector	所有参数值 All Parameter Value
S\$Params	vector	所有参数名称值 格式 Format: PARAM_NAME=PARAM_VALUE

### 11.3.2 自定义变量 Other Parameter Variable

格式 Format : P\$PARAM\_NAME

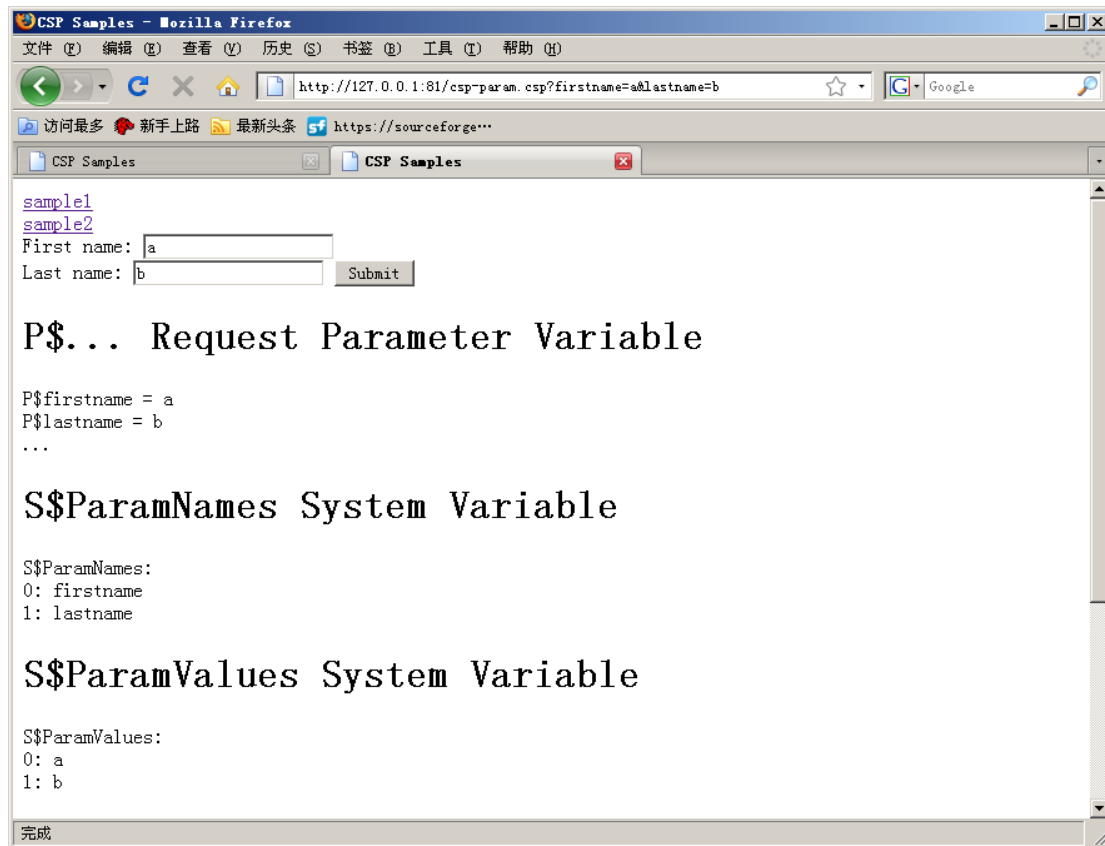
例子 Sample :

P\$User

P\$FirstName

P\$...

## 11.3.3 例子 Example



## 第12章 C++ APP

支持在 CSP 页面跟 C++ APP 应用组件进行交互，包括执行组件函数，调用服务接口方法，管理组件参数等等。

### 12.1 组件交互概述

可以通过配置 apps.xml 文件，不用在 CSP 页面定义和初始化，直接使用 C++ APP 应用组件；通过 A\$Variable 进行访问。

通过 apps.xml 配置的应用组件变量，scopy 保存区域为 application；通过 define 语句定义的应用组件变量，可以自定义 scopy 保存区域。

通过 apps.xml 文件配置的应用组件变量，不必使用 final 语句注销退出。

#### 12.1.1 执行应用组件函数

以下格式应用组件函数，可以在 CSP 页面中直接调用执行：

```
extern "C" HTTP_STATUSCODE CGC_API doServiceTest(const cgcHttpRequest::pointer  
& request, cgcHttpResponse::pointer response)
```

#### 12.1.2 调用应用组件服务接口方法

可以在 CSP 页面中，调用 C++ APP 应用组件服务接口方法，服务接口例子：

```
class CAppService  
: public cgcServiceInterface
```

#### 12.1.3 管理应用组件服务接口参数

可以在 CSP 页面中，管理 C++ APP 应用组件服务接口参数，包括有 get set add del 等方法。

## 12.2 执行组件函数 Execute APP Function

### 12.2.1 格式 Format

```
<csp:execute id="" function="" />
```

执行 id 组件的 function 方法;

### 12.2.2 例子 Example

```
<csp:execute id="cspApp" function="ServiceTest" />
```



## 12.3 调用组件服务接口方法 Call APP Function

### 12.3.1 定义组件实例 Define APP Variable

```
<csp:define type="app" id="" name="" scopy="" property="" />
```

定义一个 C++ APP 应用组件类型变量。

id	存放实例变量 APP Variable
name	组件名称 APP Name
scopy	(可选) 生命周期, 默认不填 page; optional
property	(可选) 组件默认参数 optional

## 12.3.2 初始化组件 **Init APP**

```
<csp:app:init id="" in="" />
```

id	组件变量；（以下同） APP Variable
in	输入参数 Input Property
_%result	返回结果 Out Result, boolean

## 12.3.3 注销退出组件 **Final APP**

```
<csp:app:final id="" />
```

可选：退出组件，同时删除 id 变量； Optional

## 12.3.4 调用组件服务接口方法 **APP Call**

```
<csp:app:call id="" name="" in="" out="" />
```

name	接口方法 Call Name
in	输入参数 Input Property
out	输出参数 Output Value

## 12.4 管理应用组件接口参数 **APP Property**

### 12.4.1 获取组件参数 **Get APP Property**

```
<csp:app:get id="" name="" out="" />
```

name	参数名称 Property Name
out	输出参数 Output Value

### 12.4.2 设置组件参数 **Set APP Property**

```
<csp:app:set id="" name="" in="" />
```

name	参数名称 Property Name
------	--------------------

in	输入参数 Input Property
----	---------------------

### 12.4.3 添加组件参数 Add APP Property

```
<csp:app:add id="" name="" in="" />
```

name	参数名称 Property Name
in	输入参数 Input Property

### 12.4.4 删除组件参数 Delete APP Property

```
<csp:app:del id="" name="" />
```

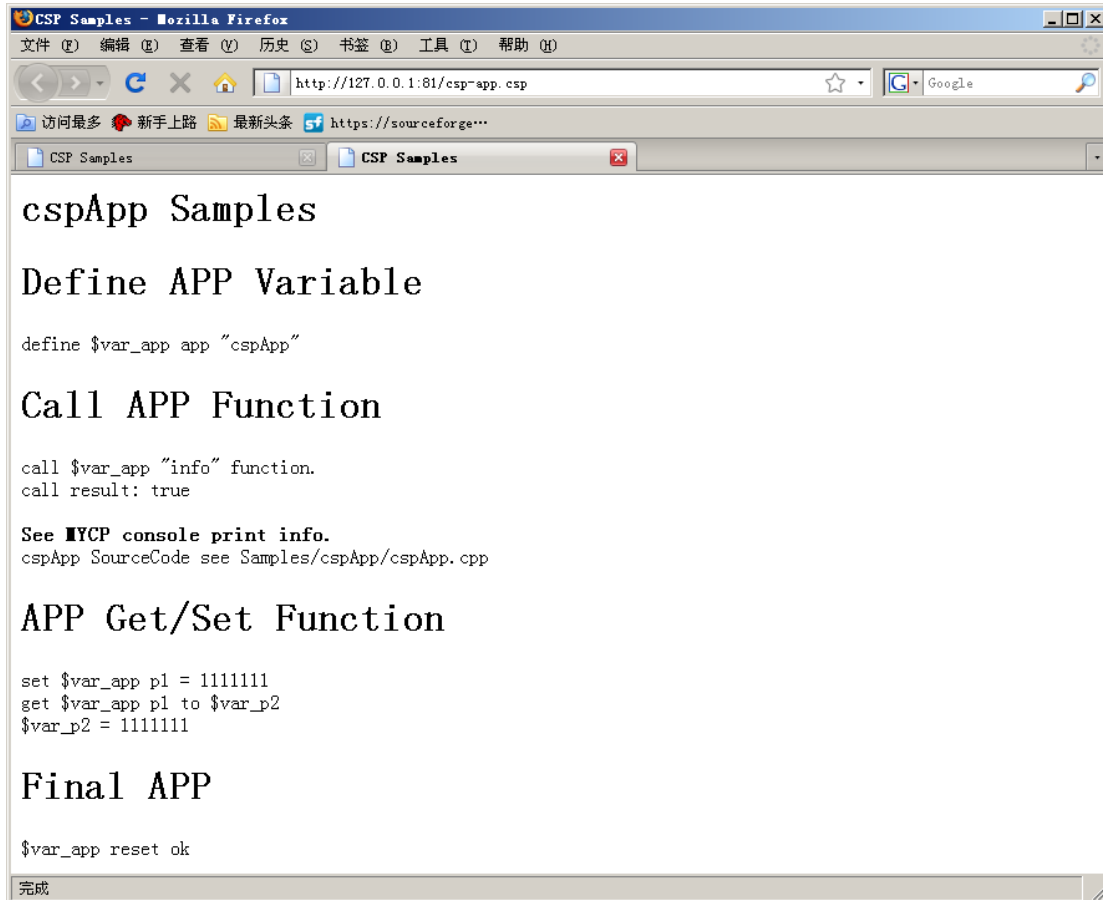
name	参数名称 Property Name
------	--------------------

### 12.4.5 获取组件信息 Get APP Info

```
<csp:app:info id="" out="" />
```



## 12.5 例子 Example



```
CSP Samples - Mozilla Firefox
文件(F) 编辑(E) 查看(V) 历史(S) 书签(B) 工具(T) 帮助(H)
http://127.0.0.1:81/csp-app.csp
访问最多 新手上路 最新头条 https://sourceforge...
CSP Samples CSP Samples

cspApp Samples

Define APP Variable

define $var_app app "cspApp"

Call APP Function

call $var_app "info" function.
call result: true

See MYCP console print info.
cspApp SourceCode see Samples/cspApp/cspApp.cpp

APP Get/Set Function

set $var_app p1 = 1111111
get $var_app p1 to $var_p2
$var_p2 = 1111111

Final APP

$var_app reset ok

完成
```

## 第13章 CDBC

CDBC (C/C++ Data Base Connectivity) 用于连接数据源 DataSource, 执行 SQL 语句。

### 13.1 CDBC 使用概述

可以通过配置 datasources.xml 文件, 不必在 CSP 页面定义 CDBC 数据源, 直接在 CSP 页面进行使用, 调用 CDBC 数据源, 执行 SQL 语句; 通过 D\$Variable 进行访问。

通过 datasources.xml 配置的 CDBC 数据源变量, scopy 保存区域为 application; 通过 define 语句定义的 CDBC 数据源变量, 可以自定义 scopy 保存区域。

### 13.2 定义 CDBC 实例 Define CDBC Variable

```
<csp:define type="cdbc" id="" name="" scopy="" />
```

id	存放实例变量 CDBC Variable
name	CDBC DataSource, cdbc.xml file
scopy	(可选)生命周期, 默认不填 page

### 13.3 更新操作 Execute SQL

```
<csp:cdbc:exec id="" sql="" />
```

sql	SQL 语句 INSERT, UPDATE, DELETE, ...
-----	------------------------------------

### 13.4 查询操作 Select SQL

```
<csp:cdbc:select id="" sql="" out="" />
```

sql	SELECT SQL 语句
out	成功返回 RESULTSET COOKIE

## 13.5 获取第一条记录 Get First

```
<csp:dbc:first id="" in="" />
```

in	RESULTSET COOKIE
\$_result	返回指定记录，失败返回 null 数据；使用“csp:size”和“csp:foreach”等语句提取数据；（下同）

## 13.6 获取下一条记录 Get Next

```
<csp:dbc:next id="" in="" />
```

## 13.7 获取上一条记录 Get Previous

```
<csp:dbc:prev id="" in="" />
```

## 13.8 获取最后一条记录 Get Last

```
<csp:dbc:last id="" in="" />
```

## 13.9 清空记录集 Reset Result

```
<csp:dbc:reset id="" in="" />
```

成功 SELECT 的记录，必须 reset 清空资源。

## 13.10 获取记录集大小 Result Size

```
<csp:dbc:size id="" in="" />
```

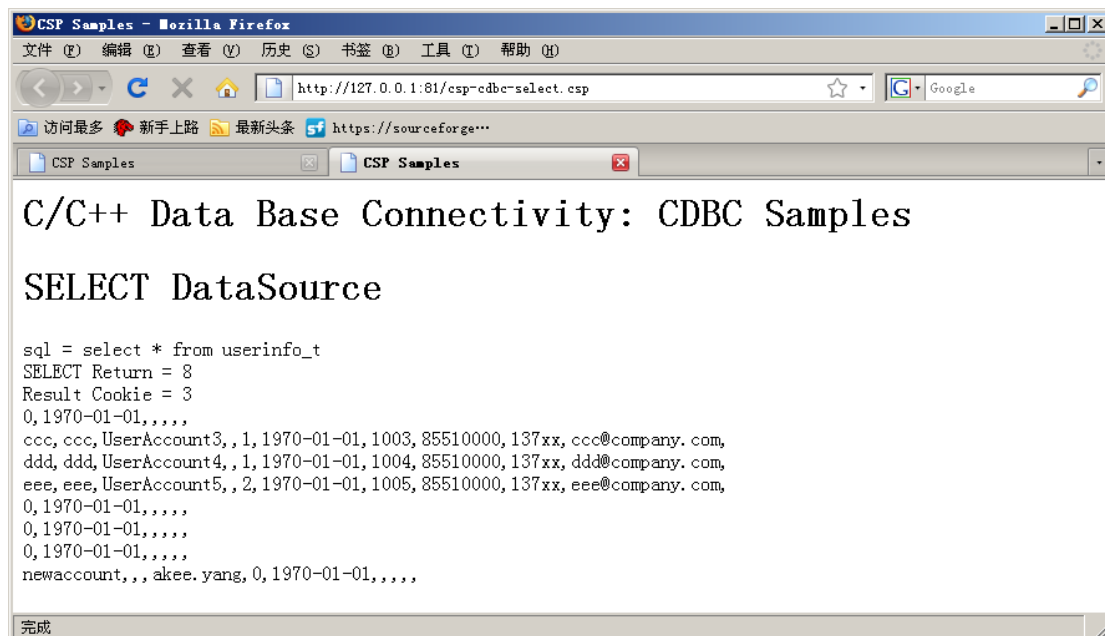
\$_size	返回记录集大小
---------	---------

## 13.11 获取当前记录索引 Result Index

```
<csp:dbc:index id="" in="" />
```

\$_index	返回当前记录集索引
----------	-----------

## 13.12 例子 Example





## 第14章 附录 A 系统变量 System Variable

变量名称	数据类型	描述
S\$HeadNames	vector	所有 HEADER 名称 All Header Name
S\$HeadValues	vector	所有 HEADER 值 All Header Value
S\$Heads	vector	所有 HEADER 名称值 格式 Format: HEADER_NAME: VALUE
S\$ParamNames	vector	所有参数名称; All Parameter Name
S\$ParamValues	vector	所有参数值 All Parameter Value
S\$Params	vector	所有参数名称值 格式 Format: PARAM_NAME=PARAM_VALUE
S\$Scheme	string	
S\$Method	int	
S\$MethodString	string	
S\$Protocol	string	
S\$ContentLength	int	
S\$ContentType	string	
S\$RequestURL	string	
S\$RequestURI	string	
S\$QueryString	string	
S\$RemoteAddr	string	
S\$RemoteHost	string	
S\$AuthAccount	string	
S\$AuthSecure	string	

S\$ServerName	string	
S\$ServerPort	int	
S\$ContextPath	string	
S\$ServletName	string	
S\$IsNewSession	boolean	
S\$IsInvalidate	boolean	
S\$SessionId	string	
S\$SessionCreationTime	time	
S\$SessionLastAccessed	time	
S\$UploadFiles	vector	