◆ Web Services 平台元素

Web Services 总结 →

Web Service 实例

任何应用程序都可拥有 Web Service 组件。

Web Service 的创建与编程语言的种类无关。

本章节我们将为大家介绍使用 PHP 的 SOAP 扩展来创建 Web Service。

SOAP有两种操作方式, NO-WSDL与 WSDL。

NO-WSDL模式:使用参数来传递要使用的信息。

● WSDL模式: 使用WSDL文件名作为参数,并从WSDL中提取服务所需的信息。

一个实例: PHP Web Service

在开始实例前,我们需要确定PHP是否安装了SOAP扩展。查看 phpinfo,出现以下信息表明已经安装了 SOAP 扩展:

soap

Soap Client	enabled
Soap Server	enabled

Directive	Local Value	Master Value
soap.wsdl_cache	1	1
soap.wsdl_cache_dir	/tmp	/tmp
soap.wsdl_cache_enabled	1	1
soap.wsdl_cache_limit	5	5
soap.wsdl_cache_ttl	86400	86400

在这个例子中,我们会使用 PHP SOAP 来创建一个简单的 Web Service。

服务端 Server.php 文件代码如下:

```
<?php

// SiteInfo 类用于处理请求
Class SiteInfo
{
    /**
    * 返回网站名称
    * @return string
    *
    */
    public function getName(){
        return "菜鸟教程";
    }

    public function getUrl(){
        return "www.runoob.com";
    }
}
</pre>
```

```
}

// 创建 SoapServer 对象

$s = new SoapServer(null,array("location"=>"http://localhost/soap/Server.php","uri"=>"Server.php"));

// 导出 SiteInfo 类中的全部函数

$s->setClass("SiteInfo");

// 处理一个SOAP请求,调用必要的功能,并发送回一个响应。

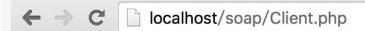
$s->handle();

?>
```

客户端 Client.php 文件代码如下:

```
<?php
try{
 // non-wsdl方式调用web service
 // 创建 SoapClient 对象
  $soap = new SoapClient(null,array('location'=>"http://localhost/soap/Server.php",'uri'=>'Server.php'
));
 // 调用函数
  $result1 = $soap->getName();
  $result2 = $soap->__soapCall("getUrl",array());
 echo $result1."<br/>";
  echo $result2;
} catch(SoapFault $e){
  echo $e->getMessage();
}catch(Exception $e){
  echo $e->getMessage();
}
```

这时我们访问 http://localhost/soap/Client.php,输出结果如下所示:



菜鸟教程

www.runoob.com

◆ Web Services 平台元素

Web Services 总结 →

② 点我分享笔记