

Node.js Web 模块

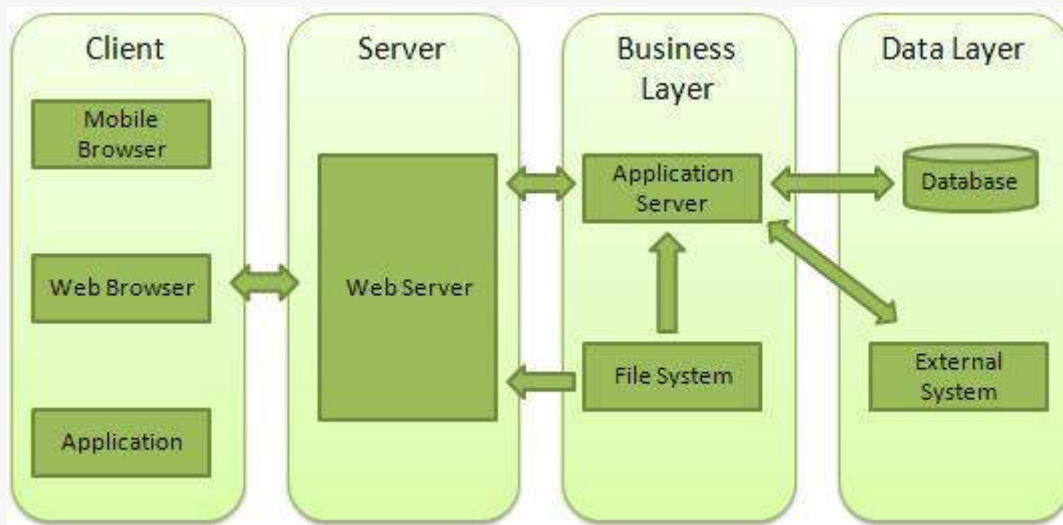
什么是 Web 服务器？

Web服务器一般指网站服务器，是指驻留于因特网上某种类型计算机的程序，Web服务器的基本功能就是提供Web信息浏览服务。它只需支持HTTP协议、HTML文档格式及URL，与客户端的网络浏览器配合。

大多数 web 服务器都支持服务端的脚本语言（php、python、ruby）等，并通过脚本语言从数据库获取数据，将结果返回给客户端浏览器。

目前最主流的三个Web服务器是Apache、Nginx、IIS。

Web 应用架构



- **Client** - 客户端，一般指浏览器，浏览器可以通过 HTTP 协议向服务器请求数据。
- **Server** - 服务端，一般指 Web 服务器，可以接收客户端请求，并向客户端发送响应数据。
- **Business** - 业务层，通过 Web 服务器处理应用程序，如与数据库交互，逻辑运算，调用外部程序等。
- **Data** - 数据层，一般由数据库组成。

使用 Node 创建 Web 服务器

Node.js 提供了 http 模块，http 模块主要用于搭建 HTTP 服务端和客户端，使用 HTTP 服务器或客户端功能必须调用 http 模块，代码如下：

```
var http = require('http');
```

以下是演示一个最基本的 HTTP 服务器架构(使用 8080 端口)，创建 server.js 文件，代码如下所示：

实例

```
var http = require('http');
var fs = require('fs');
var url = require('url');
// 创建服务器
http.createServer( function (request, response) {
// 解析请求, 包括文件名
var pathname = url.parse(request.url).pathname;
// 输出请求的文件名
console.log("Request for " + pathname + " received.");
// 从文件系统中读取请求的文件内容
fs.readFile(pathname.substr(1), function (err, data) {
if (err) {
console.log(err);
// HTTP 状态码: 404 : NOT FOUND
// Content Type: text/plain
response.writeHead(404, {'Content-Type': 'text/html'});
}else{
// HTTP 状态码: 200 : OK
// Content Type: text/plain
response.writeHead(200, {'Content-Type': 'text/html'});
// 响应文件内容
response.write(data.toString());
}
// 发送响应数据
response.end();
});
}).listen(8080);
// 控制台会输出以下信息
console.log('Server running at http://127.0.0.1:8080/');
```

接下来我们在该目录下创建一个 index.html 文件，代码如下：

index.html 文件

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>菜鸟教程(runoob.com)</title>
</head>
<body>
<h1>我的第一个标题</h1>
<p>我的第一个段落。</p>
</body>
</html>
```

执行 server.js 文件：

```
$ node server.js
Server running at http://127.0.0.1:8080/
```

接着我们在浏览器中打开地址：`http://127.0.0.1:8080/index.html`，显示如下图所示：



执行 `server.js` 的控制台输出信息如下：

```
Server running at http://127.0.0.1:8080/  
Request for /index.html received.      # 客户端请求信息
```

使用 Node 创建 Web 客户端

Node 创建 Web 客户端需要引入 `http` 模块，创建 `client.js` 文件，代码如下所示：

实例

```
var http = require('http');  
// 用于请求的选项  
var options = {  
  host: 'localhost',  
  port: '8080',  
  path: '/index.html'  
};  
// 处理响应的回调函数  
var callback = function(response){  
  // 不断更新数据  
  var body = '';  
  response.on('data', function(data) {  
    body += data;  
  });  
  response.on('end', function() {  
    // 数据接收完成  
    console.log(body);  
  });  
}  
// 向服务端发送请求  
var req = http.request(options, callback);  
req.end();
```

新开一个终端，执行 `client.js` 文件，输出结果如下：

```
$ node client.js  
<!DOCTYPE html>
```

```
<html>
<head>
<meta charset="utf-8">
<title>菜鸟教程(runoob.com)</title>
</head>
<body>
  <h1>我的第一个标题</h1>
  <p>我的第一个段落。</p>
</body>
</html>
```

执行 server.js 的控制台输出信息如下：

```
Server running at http://127.0.0.1:8080/
Request for /index.html received.    # 客户端请求信息
```

[← Node.js Domain 模块](#)

[Node.js Express 框架 →](#)

[✎ 点我分享笔记](#)