http://www.ibeifeng.com/tech.php?id=66507

II:Razor视图引擎-基础语法

　　-基础-

　　所有以 @开头 或 @{ /\* 代码体 \*/ }  (在@与{直接不得添加任何空格) 的部分代码都会被ASP.NET引擎进行处理。在 @{ /\*代码体\*/ } 内的代码每一行都必须以";"结束，如：

@{  
 var i = 10;  
 var y = 20;  
}

　　而 @xxx 则不需要以";"作为结束符，如：

　　@i 输出 10

　　@y; 输出 20;

　　代码区内字母分大小写。  
　　字符类型常量必须用""括起例如: @{ string str = "my string"; }。

　　-注意-

　　如需要在页面输出”@”字符。  
　　可以使用HTML ASCII编码&#64;  
　　当然Razor也提供智能分析功能: 如果在@的前一个字符若是非空白字符,则ASP.NET不会对其进行处理。

　　如：<p>text@i xx</p> 输出 text@i xx。

　　单行语法：

@{ var I = 10; }

　　多行语法：

@{  
 var I = 10; Var y = 20;  
}

　　1. 使用局部变量,Razor不支持访问修饰符(public,private等,这个没任何意义)

　　在单行上定义局部变量：

@{ var total = 7; }  
@{ var myMessage = "Hello World";}

　　在多行上定义局部变量：

@{  
 var greeting = "Welcome to our site!";  
 var weekDay = DateTime.Now.DayOfWeek;  
 var greetingMessage = greeting + " Today is: " + weekDay;  
}

　　在上下文中使用变量：

<p>The value of your account is: @total </p>  
<p>The value of myMessage is: @myMessage</p>

　　注意：变量拼接输出。

@{ var i = 10; }  
<p>text @i text</p>   
将输出 text 10 text

　　但是如果你想要输出 text10text 呢?

<p>text@{@i}text</p>即可  
<p>text@i text</p> 将输出 text@i text  
<p>text@itext</p> 将输出 text@itext  
<p>text @itext</p> 将报错

　　如果是输出的是变量的方法名则不需要用@{}括住也可生效如：

<p>text@i.ToString()text</p>

　　使用变量对象可直接写: @var1 @var2 @myObject.xx

**2. 使用逻辑处理**

@{  
 if (xx)  
 {  
 //do something  
 }  
 else  
 {  
 //do anything  
 }  
}

**3. 在@{... }内部使用html标记**

@{  
 <p>text</P>  
 <div>div1</div>  
}

**4. 在@{...}内部输出文本**

　　利用@:进行单行输出：

@{  
 @:This is some text  
 @:This is text too  
 @:@i 也可输出变量  
}

　　利用<text />进行多行输出：

@{  
 <text>  
 tomorrow is good  
 some girl is nice  
 </text>  
}

**5. 在@{...}内部使用注释**

@{  
 //单行注释  
 var i = 10;  
 //defg  
}  
 @\* 多行注释 \*@  
 @\*  
 多行注释  
 多行注释  
 \*@  
@{  
 @\*  
 多行注释  
 多行注释  
 \*@  
 var i = 10; @\* asdfasf \*@  
}  
<-- 同时也可以使用C#默认的/\* ... \*/ -->  
@{  
 /\*  
 多行注释  
 \*/  
}

　　若在@{ ... }内部使用<!-- -->注释,则会输出到页面之中,如果在<!-- -->内部使用@变量,则会被处理：

@{  
<!-- time now: @DateTime.Now.ToString() -->  
}  
输出: <!-- time now: 4/9/2011 12:01 -->

**6. 类型转换**

　　AsInt(), IsInt()  
　　AsBool(),IsBool()  
　　AsFloat(),IsFloat()  
　　AsDecimal(),IsDecimal()  
　　AsDateTime(),IsDateTime()  
　　ToString()  
　　例子:

@{  
 var i = “10”;  
}  
<p> i = @i.AsInt() </p> <!-- 输出 i = 10 -->

**7. 使用循环**

<!--方式1-->  
@for (int i = 10; i < 11; i++)  
{  
 @:@i  
}  
<!--方式2-->  
@{  
 for (int i = 10; i < 11; i++)  
 {  
 //do something  
 }  
}  
<!--while同理-->