

Web技术基础

信息工程学院李树青

2017秋

11 下一步学习

- HTML->XHTML->XML
- JS->服务器端动态语言

14.1 XML

- XML成为目前推荐的标准网络数据存储格式
 - XML是eXtensible Markup Language的缩写
 - XML是一种类似于HTML的标记语言，不过标记是自己自由定义的
 - XML是用来存储数据的，不负责显示格式
 - XML现在已经成为最普遍的数据操纵和数据传输的标准工具

一个简单的XML例子

```
<?xml version="1.0" encoding="UTF-8"?>
<students>
  <student>
    <number>000001</number>
    <name>黎明</name>
  </student>
  <student>
    <number>000002</number>
    <name>赵怡春</name>
  </student>
</students>
```

```
<?xml version="1.0" encoding="UTF-8"?>
<students>
  <student>
    <number>000001</number>
    <name>黎明</name>
  </student>
  <student>
    <number>000002</number>
    <name>赵怡春</name>
  </student>
</students>
```

XMLFile1.xml* XMLFile1.xml* 起始页

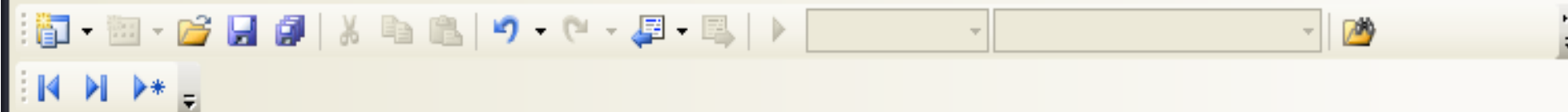
工具箱

解决方案资源管理器

错误列表

就绪 行 12 列 1 Ch 1 Ins

文件(F) 编辑(E) 视图(V) 项目(P) 调试(D) 工具(T) 窗口(W) 社区(C) 帮助(H)



XMLFile1.xml* XMLFile1.xml* 起始页

数据表:

student

数据:

student 的数据

	number	name
▶	000001	黎明
	000002	赵怡春
*		

错误列表

就绪

解决方案资源管理器

一个复杂的XML例子

```
<?xml version="1.0" encoding="UTF-8"?>
<students>
  <student>
    <number>000001</number>
    <name>黎明</name>
    <grades>
      <course>A03</course>
      <grade>56</grade>
    </grades>
    <grades>
      <course>B01</course>
      <grade>78</grade>
    </grades>
  </student>

```

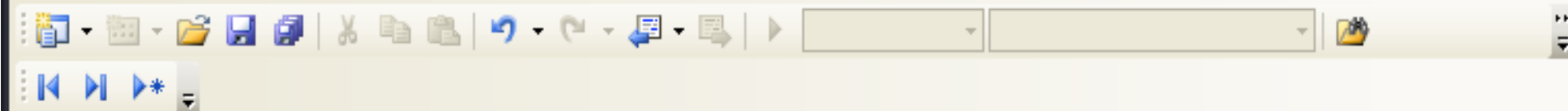
一个复杂的XML例子

```
<student>  
  <number>000002</number>  
  <name>赵怡春</name>  
  <grades>  
    <course>A03</course>  
    <grade>81</grade>  
  </grades>  
</student>  
</students>
```



```
<?xml version="1.0" encoding="UTF-8"?>
<students>
  <student>
    <number>000001</number>
    <name>黎明</name>
    <grades>
      <course>A03</course>
      <grade>56</grade>
    </grades>
    <grades>
      <course>B01</course>
      <grade>78</grade>
    </grades>
  </student>
  <student>
    <number>000002</number>
    <name>赵怡春</name>
    <grades>
      <course>A03</course>
      <grade>81</grade>
    </grades>
  </student>
</students>
```

就绪 行 19 列 27 Ch 27 Ins



数据表:

student
grades

数据:

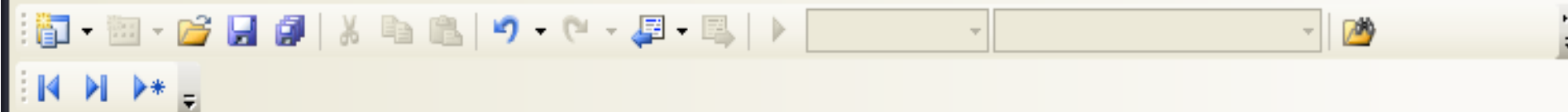
student 的数据

	number	name
▶ ⊕	000001	黎明
⊕	000002	赵怡春
*		

错误列表

就绪

文件(F) 编辑(E) 视图(V) 项目(P) 调试(D) 工具(T) 窗口(W) 社区(C) 帮助(H)



XMLFile1.xml* XMLFile1.xml* 起始页

数据表:

student
grades

数据:

grades 的数据

	course	grade
▶	A03	56
	B01	78
	A03	81
*		

错误列表

就绪

解决方案资源管理器

网页中的XML

- XML文件本身的显示
 - CSS
 - XSL
- XML对一般网页的数据支持
 - 客户端数据岛技术
 - 服务器端编程技术

网页处理XML

- JS读取XML
- Ajax处理方法

14.2 JSON

- JavaScript 对象表示法
 - JavaScript Object Notation
- 轻量级的文本数据交换格式
- 独立于语言
- 具有自我描述性，更易理解
- 类似 XML，比 XML 更小更快，更易解析

JSON数据示例

- ```
{
 "employees": [
 { "firstName": "Bill" , "lastName": "Gates" },
 { "firstName": "George" , "lastName": "Bush" },
 { "firstName": "Thomas" , "lastName": "Carter" }
]
}
```

- 这个 employee 对象是包含 3 个员工对象的数组

# 对应的XML数据

- `<?xml version="1.0" encoding="UTF-8" ?>`
- `<root>`
- `<employees>`
- `<firstName>Bill</firstName>`
- `<lastName>Gates</lastName>`
- `</employees>`
- `<employees>`
- `<firstName>George</firstName>`
- `<lastName>Bush</lastName>`
- `</employees>`
- `<employees>`
- `<firstName>Thomas</firstName>`
- `<lastName>Carter</lastName>`
- `</employees>`
- `</root>`



### 在线XML、JSON数据互转

如题

XML

```
<?xml version="1.0" encoding="UTF-8" ?>
<employees>
 <firstName>Bill</firstName>
 <lastName>Gates</lastName>
</employees>
<employees>
 <firstName>George</firstName>
 <lastName>Bush</lastName>
</employees>
<employees>
 <firstName>Thomas</firstName>
 <lastName>Carter</lastName>
</employees>
```

JSON

```
{
 "employees": [
 { "firstName": "Bill", "lastName": "Gates" },
 { "firstName": "George", "lastName": "Bush" },
 { "firstName": "Thomas", "lastName": "Carter" }
]
}
```

转义编码

→

←

清空

格式化

压缩

验证

JSON 转 XML

JSON 转 CSV

JSON 转 YAML

JSON 教程

JSON 在线解析

```
1 {
2 "employees": [
3 {
4 "firstName": "Bill",
5 "lastName": "Gates"
6 },
7 {
8 "firstName": "George",
9 "lastName": "Bush"
10 },
11 {
12 "firstName": "Thomas",
13 "lastName": "Carter"
14 }
15]
16 }
```

Tree

- object {1}
- employees [3]
- 0 {2}
- firstName : Bill
- lastName : Gates
- 1 {2}
- firstName : George
- lastName : Bush
- 2 {2}
- firstName : Thomas
- lastName : Carter

# JavaScript访问JSON数据

- 利用eval函数可以创建JSON对象，更方便使用

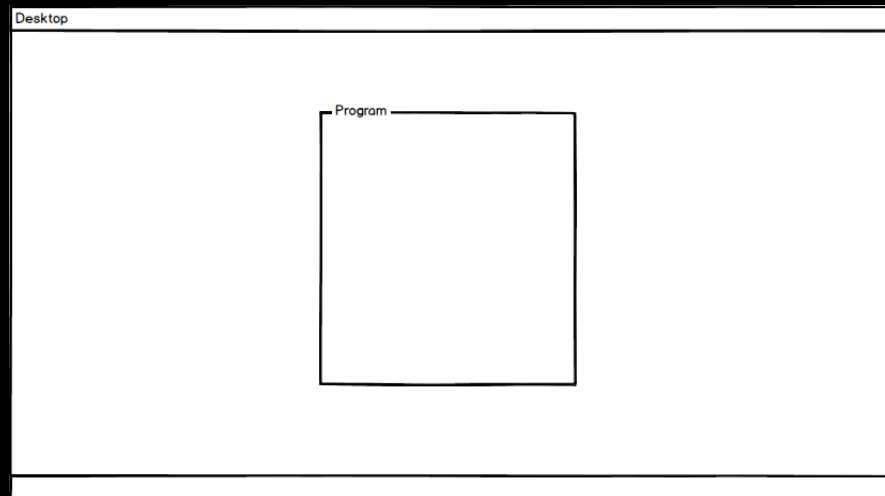
## 14.2 服务器端动态语言

- ASP
- ASP.Net
- JSP ( JavaEE )
- PHP
- ...

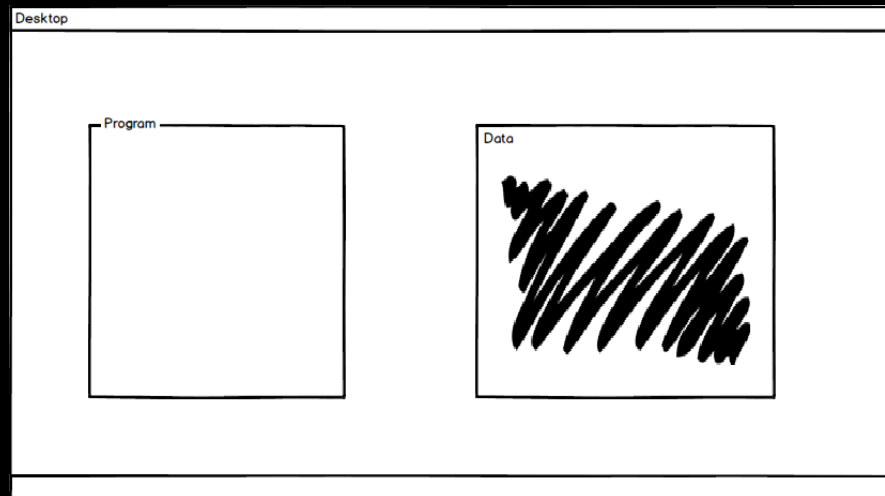
- 什么是服务器端动态？
- 为什么要服务器端动态？

# 结语

- 互联网改变了一切

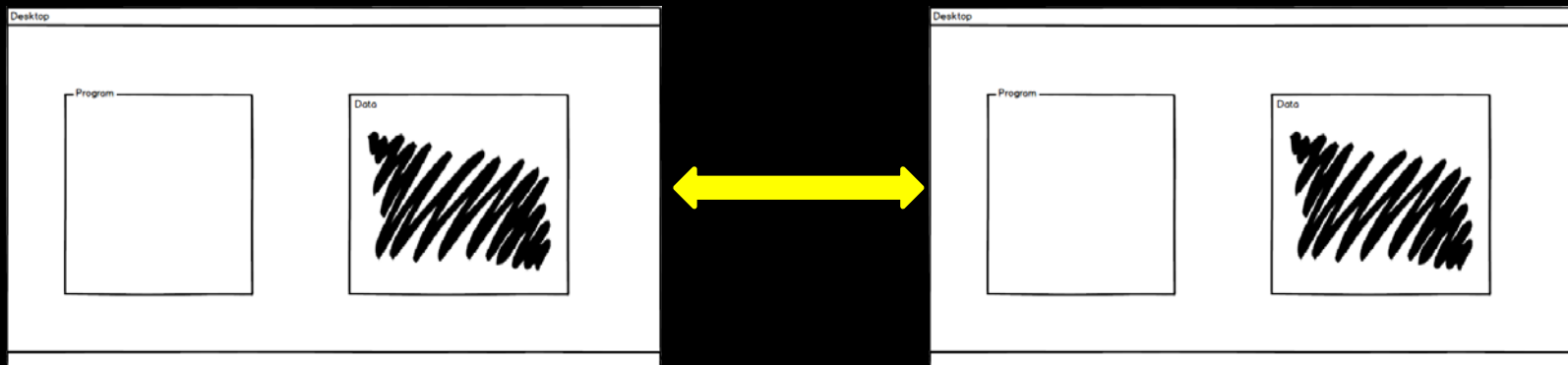


**Program**



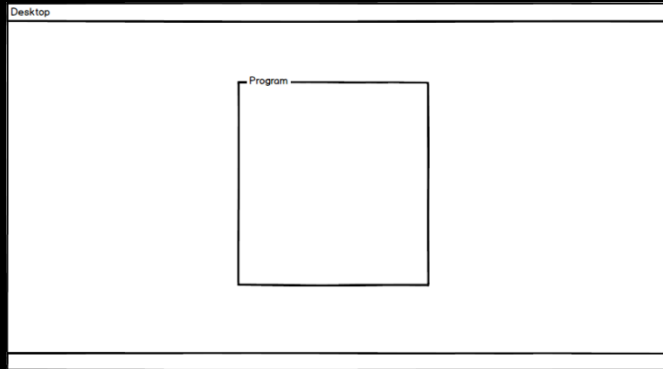
**Program+Data**



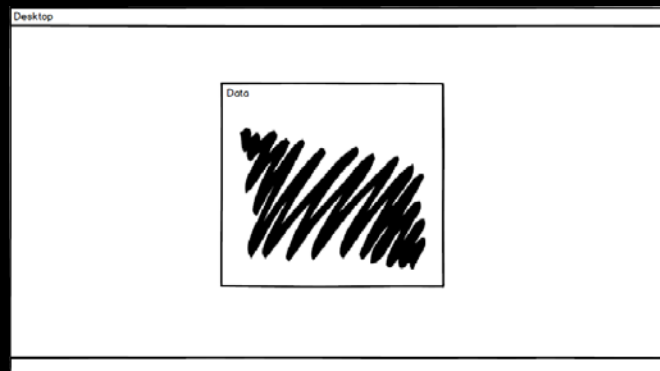
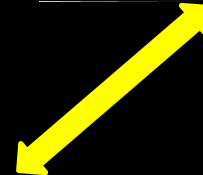
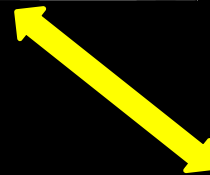
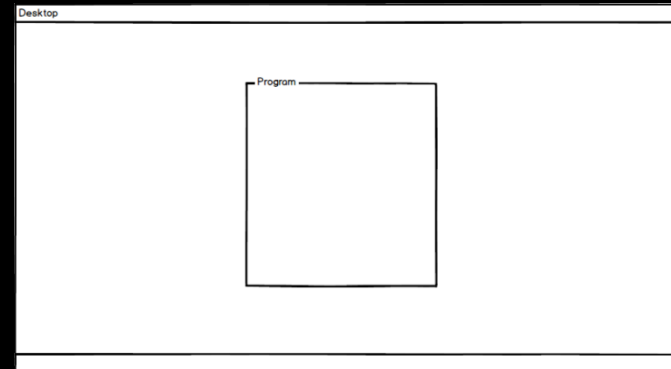


**Network**

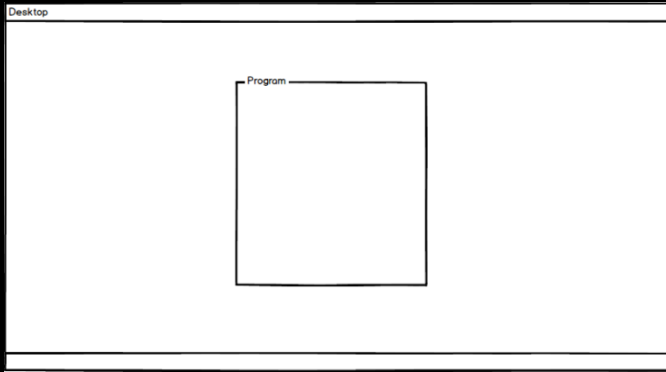
# Program



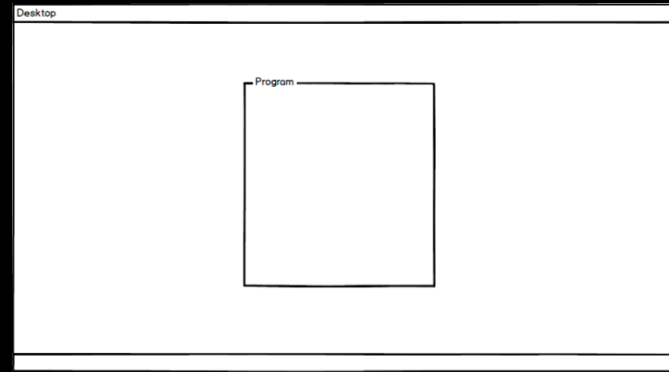
# Program



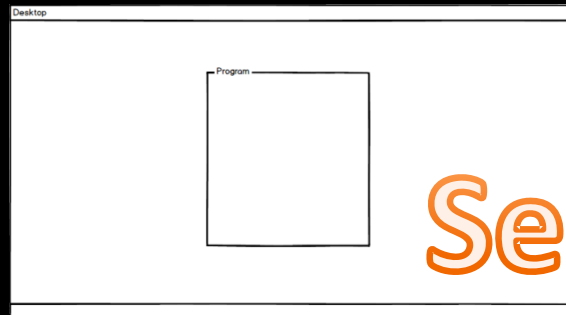
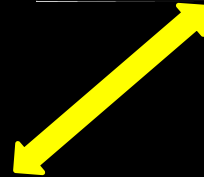
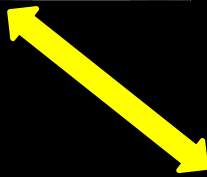
# Data



Client

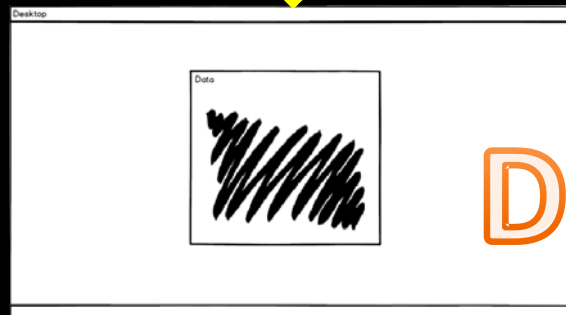


Client

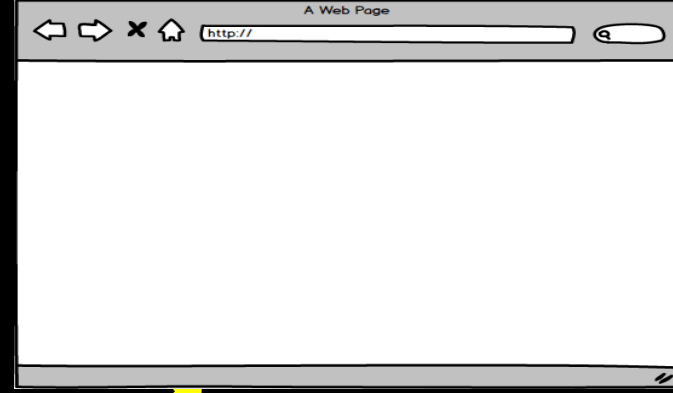
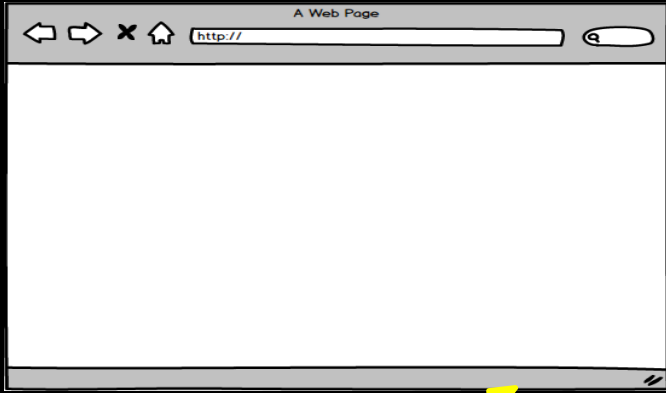


Server

C/S

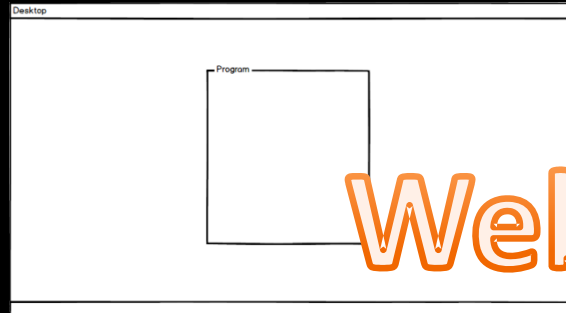


Data



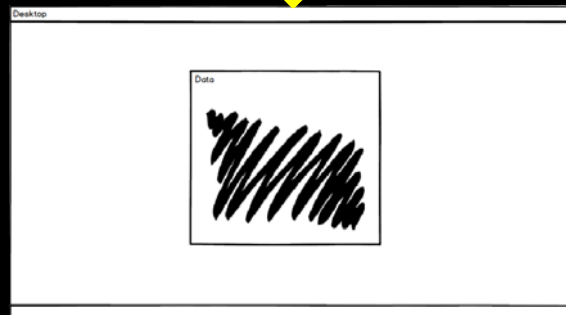
**Browser**

**Browser**



**Web Server**

**B/S**



**RDBMS  
XML,JSON  
Text**

