张东

18301092802

三样东西:

1. 犀牛书——JavaScript权威指南

2. 公众号——前端大全

3. 先拿到师兄的笔记和视频——预习

上课规矩: 禁止记笔记

内容:

JS高级: 面向对象，闭包

ES5 ES6：

DOM/jQuery：

Vue：

案例的使用方法: 每个案例写三遍:

1. 注释+代码抄一遍

2. 删除代码，自己把代码填回来

3. 删除注释和代码，自己试着都写回来

——专治: 上课听的懂，下课做不会

其实，注释远比代码更重要！因为没地方查去！

强调: 如果发现函数名忘了，或者函数不会用，先去思维导图！！！！！！

正课:

1. 正则表达式:

2. String提供的正则函数:

1. 正则表达式（Regular Expression）:

什么是: 计算机中描述一个字符串中字符出现规律的规则

为什么: 因为计算机不懂人类的词汇的规则，所以需要人教计算机认识人类语言中词汇的规则：手机号，身份证号，车牌号。。。

何时: 2种:

1. 验证输入的字符串格式

2. 在字符串中查找敏感词时

如何:

1. 最简单的规则其实就是敏感词原文：

比如: 我草

规定：必须是2个字

第一个字必须是我

第二个字必须是草

问题: 只要敏感词一变，就找不到了！

2. 字符集:

什么是：描述一位字符上多个备选字的集合

何时：如果敏感词中一位字符有多种备选字的可能时。

如何: [备选字列表]

比如: 想匹配多种草字: 草，艹，操，槽

[草艹操槽]

强调: 字和字之间不要带任何分隔符！

效果: 多个字每次选其一匹配

简写：如果备选字列表中部分字符是连续的

比如: 一位数字：[0123456789]

一位大写字母: [ABC...]

可用-省略中间字符，-读作"到"

重要的简写:

一位数字：[0-9]

一位大写字母: [A-Z]

一位小写字母:[a-z]

一位字母：[A-Za-z]

不能是[A-z]

因为在ASCII表中，大写字母和小写字母不连续。A-Z: 65~90 [\]\_` a-z:97~122。选择范围大于字母的范围。可能错选。

一位字母或数字: [0-9A-Za-z]

一位汉字: [\u4e00-\u9fa5]

3. 预定义字符集:

什么是: 对常用的字符集提供的更简化写法

包括:

一位数字： \d 等效于 [0-9]

一位字母、数字或\_:

\w 等效于 [0-9A-Za-z\_]

一位空字符：\s 可匹配:

空格、制表符Tab等一切看不见的空字符

所有文字（通配符）： .

4. 数量词:

什么是: 专门定义一个字符集出现次数的规则

何时: 只要一个字符集可能出现多次，都要用量词修饰。

如何: 2大类：

1. 有明确数量边界:

字符集{5} 必须重复出现5次

字符集{4,6} 4次到6次

字符集{6,} 6次以上，多了不限

2. 没有明确数量边界：

字符集\* 可有可无，多了不限

字符集? 可有可无，最多一次

字符集+ 至少一个，多了不限

5. 选择:

什么是: 在多个子规则选其一匹配即可

何时: 如果希望在多个子规则中选其一匹配时，就用选择。

如何: 子规则1 | 子规则2

或

意思: 要么选择左边的子规则1匹配，要么选择右边的子规则2匹配

6. 分组:

什么是: 将多个规则分为一组先匹配，再和分组外的规则联合匹配。

何时: 希望多个规则局部先匹配，再和外部组合时。

如何: (多个规则)

贪婪模式 和 |

默认情况: 正则表达式都是采用贪婪模式匹配尽量长的满足规则的字符串

比如: 只想匹配4位或6位数字的验证码，不想匹配五位的数字。

\d{4}(\d{2})?

先固定4位,后两位整体可有可无最多一次

但是: |不遵守贪婪模式，会按顺序优先匹配。一旦前一个规则匹配上了，就算后一个规则也匹配，也不再执行。

比如: \d{4}|\d{6} 几乎永远只能匹配4位，无法匹配6位。因为即使是六位数字也会先匹配\d{4}，而导致后边的\d{6}不再起作用。

如果非用或，也要颠倒位置: \d{6}|\d{4}

7. 特殊位置:

三个位置: 字符串开头^，字符串结尾$，单词边界\b

何时: 如果不想匹配全部，而是只想匹配指定位置上的敏感词时，才用特殊位置。

比如: 开头的空字符: ^\s+

结尾的空字符：\s+$

开头或结尾的空字符: ^\s+|\s+$

单词边界: \b[a-z]

强调: \b等于空格吗？等于标点符号？都不等于。\b是一个位置，不是字符，不等于空格和标点符号。

2. String提供的正则相关函数:

js中用正则可以做三件事:

1. 查找敏感词：4种不同场景:

1. 在字符串中查找一个固定的敏感词出现的位置：

var i=str.indexOf("敏感词",fromi)

位置<的<敏感词

在str中，从fromi位置向后，找下一个"敏感词"的位置。

返回值: 如果找到，返回敏感词第一个字的位置i。

如果没找到返回-1.

问题: 只能查找一个写死的固定的敏感词。一旦敏感词有任何稍微的修改，都会导致查找不到。

2. 用正则表达式查找多种敏感词:

何时: 如果敏感词有多种变化，就改为用search

如何: var i=str.search(/正则/i)

返回值: 如果找到，返回敏感词第一个字的位置i。

如果没找到返回-1.

问题: 所有正则表达式默认是区分大小写的。

解决: 忽略大小写: ignore 忽略

在第二个/后加后缀i简写

2. 替换敏感词

3. 切割字符串

英文单词:

1. Regular: 规则

2. Expression: 表达式 做一件事的一条程序的语句

3. digit 数字

4. space 空格

5. ignore 忽略

6. document 文档 在js中特指当前网页