# day02

复习

复杂查询

聚合查询/分组查询

COUNT()/SUM()/MAX()/MIN()/AVG()

YEAR()/MONTH()

GROUP BY

子查询

多表查询

查询条件

内连接 t1 INNER JOIN t2 ON...

外连接 t1 LEFT OUTER JOIN t2 ON...

t1 RIGHT OUTER JOIN t2 ON...

全连接 UNION/UNION ALL

浏览器/服务器 nodejs

变量 var $\_a1=1;

常量 const pi=3.14;

日期时间存储的是 距离计算机元年的毫秒数

1秒=1000毫秒

1970年1月1日 0:0:0

2019-5-5 9:13:20

## 1.数据类型

原始类型和引用类型

### 1.原始类型

(1)数值型

整形和浮点型

8 16 10

314.15 31.415\*10 3.1414\*10^2

3.1514e2

(2)字符串型

数据被引号所包含就是字符串型，不区分单双引号

查看某个字符的Unicode码

'a'.charCodeAt() //97

(3)布尔型

在程序中表示真或者假的结果

true/false

常用语一些是否的结果，例如是否登录，是否注册，是否为会员，是否在售

var isLogin=false;

(4)未定义型

声明了变量未赋值，结果是undefined

(5)空

null，常用于引用类型

检测数据类型

typeof 数据

五种类型：

数值型：number

字符串型：string

布尔型：boolean

未定义型：undefined

空：object

### 2.数据类型转换

(1)隐式转换

①数值型+字符串型 数值型转成字符串型

1+'2' //'12'

②数值型+布尔型 布尔型转换成数值型 true->a false->0

1+true //2

③字符串型+布尔型 布尔型转成字符串型

'hello'+true //'hellotrue'

JS中加号(+)作用

执行加法运算

字符串的拼接

练习：声明变量保存姓名name，性别sex，年龄age；使用字符串拼接打印‘我的姓名(变量中的值),性别(变量中的值),我的年龄为(变量中的值)’。

练习：查看一下程序的执行结果

var a=3,b=true,c='tedu';

console.log(a+b+c);//'4tedu'

console.log(b+c+a);//'truetedu3'

console.log(c+a+b);//'tedu3true'

减法、乘法、除法隐式转换

将运算符两端的数据转成数值型，自动调用Number来转，如果字符串中含有非数字返回NaN(Not a Number),任何值和NaN执行减乘除都返回NaN

(2)强制转换

①将数据转为数值型

Number()

Number('9') //9

Number('2a') //NaN

Number('true') //1

Number(undefined) //NaN

Number(null) //0

②将数据转为整型

parseInt()

parseInt(2.5);//2

parseInt('2.9a');//2

parseInt('a2');//NaN

③将数据转为浮点型

parseFloat()

parseFloat(2.9a);//2.9

parseFloat('a2');//NaN

练习：将布尔型、未定义型、空转换为整型或者浮点型

parseInt和parseFloat在将布尔型、未定义型、空转数值的时候，都返回NaN

④将数值型和布尔型转成字符串型

toString()

var n=5;

n.toString() //'5'

### 3.运算符

由运算符连接操作的数据，所组成的形式就是表达式

(1)算数运算符

+ - \* / % ++ --

% 取余

++ 自增：在当前的基础之上加1

-- 自减：在当前的基础之上减1

console.log(num++ 先打印num的值，然后再执行自增)

console.log(++num) 先执行自增，然后打印num的值

练习：

var a=1;

console.log(a++ + ++a);

1 3

var b=2;

console.log(--b + b--);

1 1

(2)比较运算符

>= <= > < ==(等于) != ===(全等于) !===(不全等于)

返回布尔型(true/false)的值

== 只是比较两个值是否相等

=== 不仅比较值，还会比较数据类型

3>'10' //false

数字和字符串比较，字符串转为数值型

'3'>'10' //true

两个字符串比较，比较的是首个字符的Unicode码

3>NaN //false

3<NaN //false

3==NaN //false

NaN==NaN //false

NaN和任何值比较(> < >= <= == ===)都是false,包括自身

(3)逻辑运算符

并且&& 关联的两个条件都是true,结果才是true,否则false

或者|| 关联的两个条件都是false,结果才是false,否则true

非(反向)! !true->false !false->true

逻辑短路

&& 当第一个条件为false的时候，就不需要再执行第二个条件

|| 当第一个条件为true的时候，就不需要再执行第二个条件

注意：逻辑短路无需关注最终的结果是true还是false，重点看第二个表达式是否执行。

练习：查看以下程序是否会报错

var num=3;

num>5 && console.log(a);

num<1 || console.log(a);

(4)位运算符

在执行运算的时候，计算机会将数据转成二进制进行运算

1 10 11 100 101 110 111 1000 1001 1010 1110 1111

2 4 8

101101=100000+1000+100+1

32 8 4 1

0 1 2 3 4 5 6 7

1 2 4 8 16 32 64 128

25=16+8+1

10000+1000+1=11001

按位与(&):上下两位都是1，结果是1，否则为0

按位或(|):上下两位含有1，结果是1，否则为0

课后任务：

(1)复习，整理思维导图

(2)练习：声明变量保存一个年份，判断是否为闰年，结合着逻辑短路，如果是闰年，打印“闰年”。(普通闰年:能被4整除但不能被100整除,世纪闰年:能被400整除的为世纪闰年)。

var year=2012;

((year%4===0&&year%100!==0)||year%400===0) && console.log('闰年');