

# JavaScript深入浅出

表达式和运算符

@Bosn

# 表达式

表达式是指能计算出值得任何可用程序单元。 ——Wiki

表达式是一种JS短语，可使JS解释器用来产生一个值。 ——《JS权威指南》

表达式

原始表达式

常量、直接量

3.14, "test"

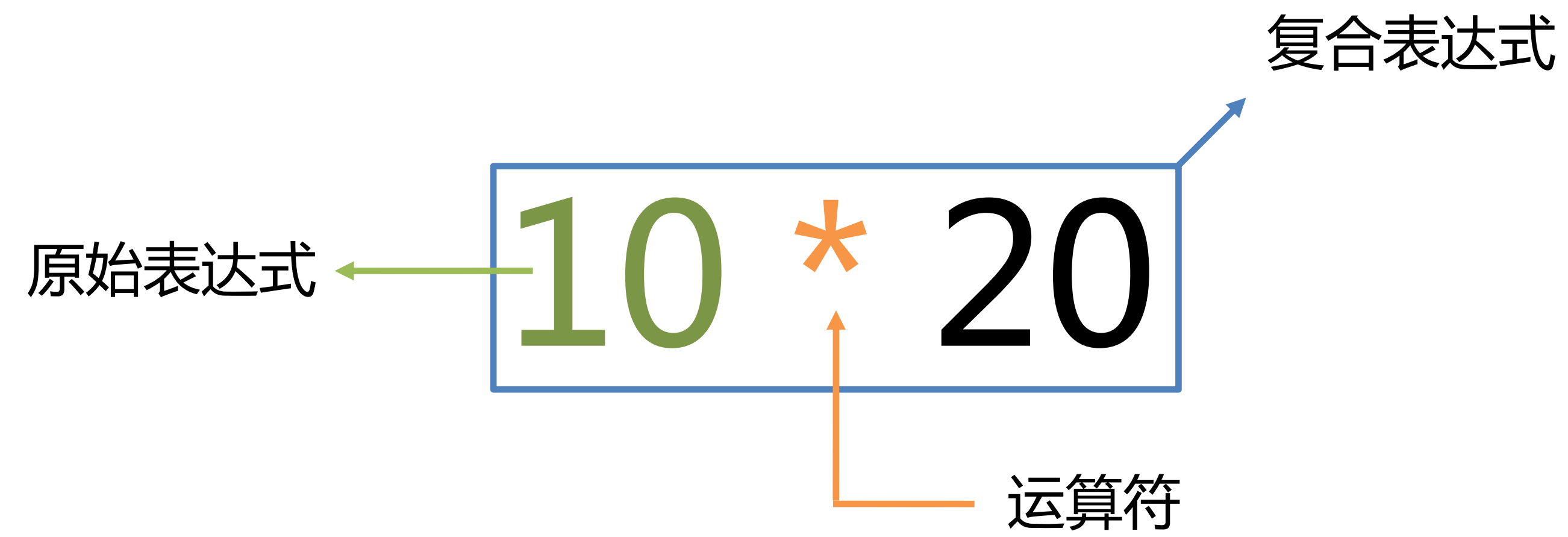
关键字

null, this, true

变量

i, k, j

表达式



## 数组、对象的初始化表达式

[1, 2]

new Array(1, 2);

[1, , , 4]

[1, undefined, undefined, 4]

{ x : 1, y : 2 }

var o = new Object();  
o.x = 1; o.y = 2;

表达式

## 函数表达式

```
var fe = function(){};
```

```
(function(){console.log('hello world');})();
```

表达式

## 属性访问表达式

```
var o = {x : 1};
```

`o.x`

`o['x']`

表达式

调用表达式

`func();`



表达式

## 对象创建表达式

```
new Func(1, 2);  
new Object;
```

表达式

原始表达式

初始化表达式

函数表达式

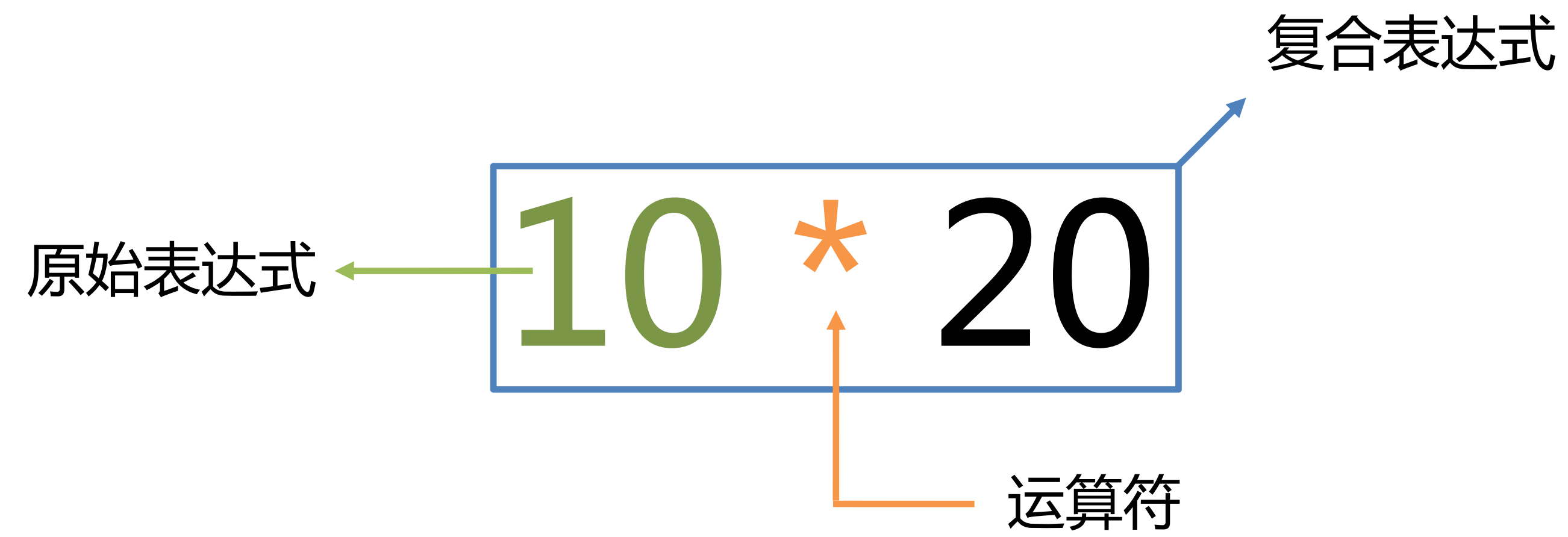
表达式

属性访问表达式

调用表达式

对象创建表达式

运算符



运算符

一元	+num
二元	a + b
三元	c ? a : b

赋值	x += 1
比较	a == b
算术	a - b
位	a   b
逻辑	exp1 && exp2
字符串	"a" + "b"
特殊	delete obj.x

运算符 ? :

$c ? a : b$

```
var val = true ? 1 : 2; // val = 1
```

运算符,

a, b

```
var val = (1, 2, 3); // val = 3
```

```
delete obj.x;
```

```
var obj = {x : 1};  
obj.x;           // 1  
delete obj.x;  
obj.x;           // undefined
```

```
var obj = {};  
Object.defineProperty(obj, 'x', {  
    configurable : false,  
    value : 1  
});  
delete obj.x; // false  
obj.x;        // 1
```



运算符 in

```
window.x = 1;  
'x' in window; // true
```

```
{ } isinstance Object // true  
typeof 100 === 'number' // true
```

```
function Foo(){  
  Foo.prototype.x = 1;  
var obj = new Foo();  
obj.x; // 1  
obj.hasOwnProperty('x'); // false  
obj.__proto__.hasOwnProperty('x'); // true
```

```
this; // window (浏览器)
var obj = {
  func : function(){return this;}
};
obj.func(); // obj
```

运算符 void

```
void 0 // undefined  
void(0) // undefined
```

# 特殊运算符

条件运算符

`c ? a : b`

逗号运算符

`a, b`

`delete`

`delete obj.x`

`in`

`"document" in window`

`instanceof`

`obj instanceof Func`

`new`

`new ClsName()`

`this`

`return this;`

`typeof`

`typeof 100`

`void`

`void 0`

运算优先级

成员	. []	位与	&
调用/new	() new	位异或	^
	! ~ - + ++ -- typeof void delete	位或	/
乘除	* / %	逻辑与	&&
加减	+ -	逻辑或	//
移位	<< >> >>>	条件	?:
关系	< <= > >= in instanceof	赋值	= += -= *= /= %= <<= >>= >>>= &= ^= /=
相等	== != === !==	逗号	,

小结

表达式 & 运算符



谢谢