1. Expression 表达式.

Grammar.

Grammar Tree 语法树

VS

Priority 优先级

Runtime

Type Conversion 类型转换

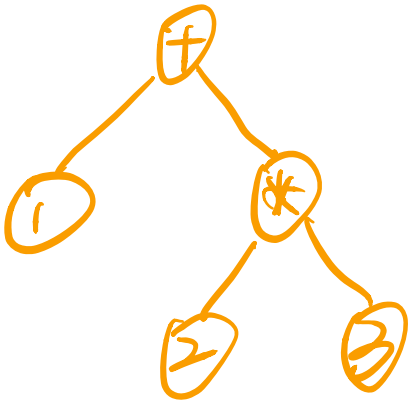
Reference 引用类型

Left hand side

&

Right hand side

2. 1+2*3



中缀树.

3. Member 表达式 | New 表达式.

a.b
a[b]
成员访问

foo`string`
super.b
super['b']
new.target
new Foo()

new Foo

优先级更高.

4. Reference
Object, key
string/Symbol

delete, assign.

5. Call (expression)
优先级低于 New. 低于 Member

```
foo()
super()
foo()['b']
foo().b
foo()`abc`
```

6. Left Handside & Right Handside

7. Update

```
a++
a--
--a
++a
```

8. Unary 单目

delete a.b

void foo()

typeof a

+a

-a

~a

!a

await a

9. Exponental 乘方

** (右结合)

10. Multiplicative

*, /, %

Additive +, -

Shift <<, >>, >>>

Relationship

<, >, <=, >=, instanceof, in

11. Equality

==

!=

===

!==

Bitwise

&, ^, |

12. Logical

&&, ||, ← 短路原则.

Conditional

?: (三目运算符)

13. Type Convertion 类型转换.

	Number	String	Boolean	Undefined	Null	Object	Symbol
Number	-		0 false	×	×	Boxing	×
String		-	"" false	×	×	Boxing	×
Boolean	true 1 false 0	'true' 'false'	-	×	×	Boxing	×
Undefined	NaN	'Undefined'	false	-	×	×	×
Null	0	'null'	false	×	-	×	×
Object	valueOf	valueOf toString	true	×	×	-	×
Symbol	×	×	×	×	×	Boxing	-

14. Unboxing : Object → 基本类型

toPrimative

toString vs valueOf

Symbol.toPrimitive

15. Boxing

类型	对象	值
Number	new Number()	
String	new String("a")	"a"
Boolean	new Boolean(true)	true
Symbol	new Object(Symbol("a"))	Symbol("a")