

PWA

- description : PWA
- author :
- email : shlim@repia.com
- lastupdate : 2021-02-01

Ref

- : “Do it!” , : , 2020 8 ,

Promise []

- .
- .
- 가 .

var

- (function scope) -> var .
- (block scope) -> var .
- .
- var .
 - (scope) :
 - :

mutable[, 가] vs immutable[,]

- mutable : -> let
- immutable : -> const

let

```
{
  let      ;      //
  let      = ;      //
}
```

const

```
{
  const    = ;      //
}
```

(block scope) - let const

```
{
  let apple = 3;
  console.log(apple); // 3
  apple = 5;          // let
  console.log(apple); // 5
  let apple = 7;      // !

  const car;          // !
  const carname = "   ";
  console.log(carname); // '   '
  carname = "   "      // const ;
}
console.log(apple);    // !
console.log(carname);  // !
```

3

- let const 가

1. const 가 . const let .
2. const let .
3. var ES6+ . 가

[Template Literal]

- ES6 (``)
- (\$) ({ })
- +
-

```
console.log(' : ' + apple + ', : ' + carname); //
console.log(` : ${apple}, : ${carname}`); // ES6 (
)
```

- (arrow function) ES6 가
- ⇒ (fat arrow)'

```
const fnPlusNumbers = function (pNum1, pNum2) {
  return pNum1 + pNum2;
}
```

```
const fnPlusNumbers = (pNum1, pNum2) => {
  return pNum1 + pNum2;
}
```

return

```
const fnPlusNumbers = (num1, num2) => num1 + num2;
```

- 가 (statement)가
- return
- num1 + num2

- 가 1 () , 가

- 가 1 가

```
const fnPlusNumbers4 = pNum => pNum + 1;
```

- 가

```
const fnSayHello = () => ' !';
```

	가	
	가	가
		가

- (callback function)
- (Object)
- (callback function)
- 'callback'

```
function fnFunctionA(pNumA) {
  console.log(pNumA);
}
function fnFunctionB(pNumB, fnCallback) {
  fnCallback(pNumB); //
}

fnFunctionB(" !", fnFunctionA);
```

- fnFunctionB()

```
function fnFunctionB(pNumB, fnCallback) {
  fnCallback(pNumB);
}

//
fnFunctionB("!", function fnFunctionA(pNumA) {
  console.log(pNumA);
});
```

- 가 . dot notation
 ○ (.)
 ○ ([...]) **<sp>bracket notation</sup>**
- 가 , 가
- 가

```
var person = {
  name: 'Lee'
};

//
console.log(person.name); // Lee

//
console.log(person['name']); // Lee
```

```
var person = {
  name: 'Lee'
};

console.log(person[name]); // ReferenceError: name is not defined
```

* ReferenceError가
 name 가 name . *
undefined . ReferenceError가

```
var person = {
```

```
    name: 'Lee'
};

console.log(person.age); // undefined
*      가                      ,      가      가
                        .      ,      가
                        .      .
```

ES11, ES6 Syntax Summery

- : by , (ES6, ES11) |
- Optional Chaining (ES11)
 - [Optional Chaining](#)
- Nullish Coalescing Operator (ES11)
 - [Nullish Coalescing Operator](#)
- Shorthand property names (ES6)
 -
- Destructuring Assignment (ES6)
 -
- Spread Syntax (ES6)
 -
- Default parameter (Es6)
 -
- Ternary Operator (ES6)
 -
- Template Literals (ES6)

o

, javascript, note,, ,

From:

<http://rwiki.repia.com/> -

. - 2023.12

Permanent link:

<http://rwiki.repia.com/doku.php?id=wiki:miscellaneous:pwa&rev=1616749501>Last update: **2022/03/10 19:52**