

자바스크립트 객체 생성법

- 1. 객체리터럴 이용

```
var p1={ name:"전은수" }
```

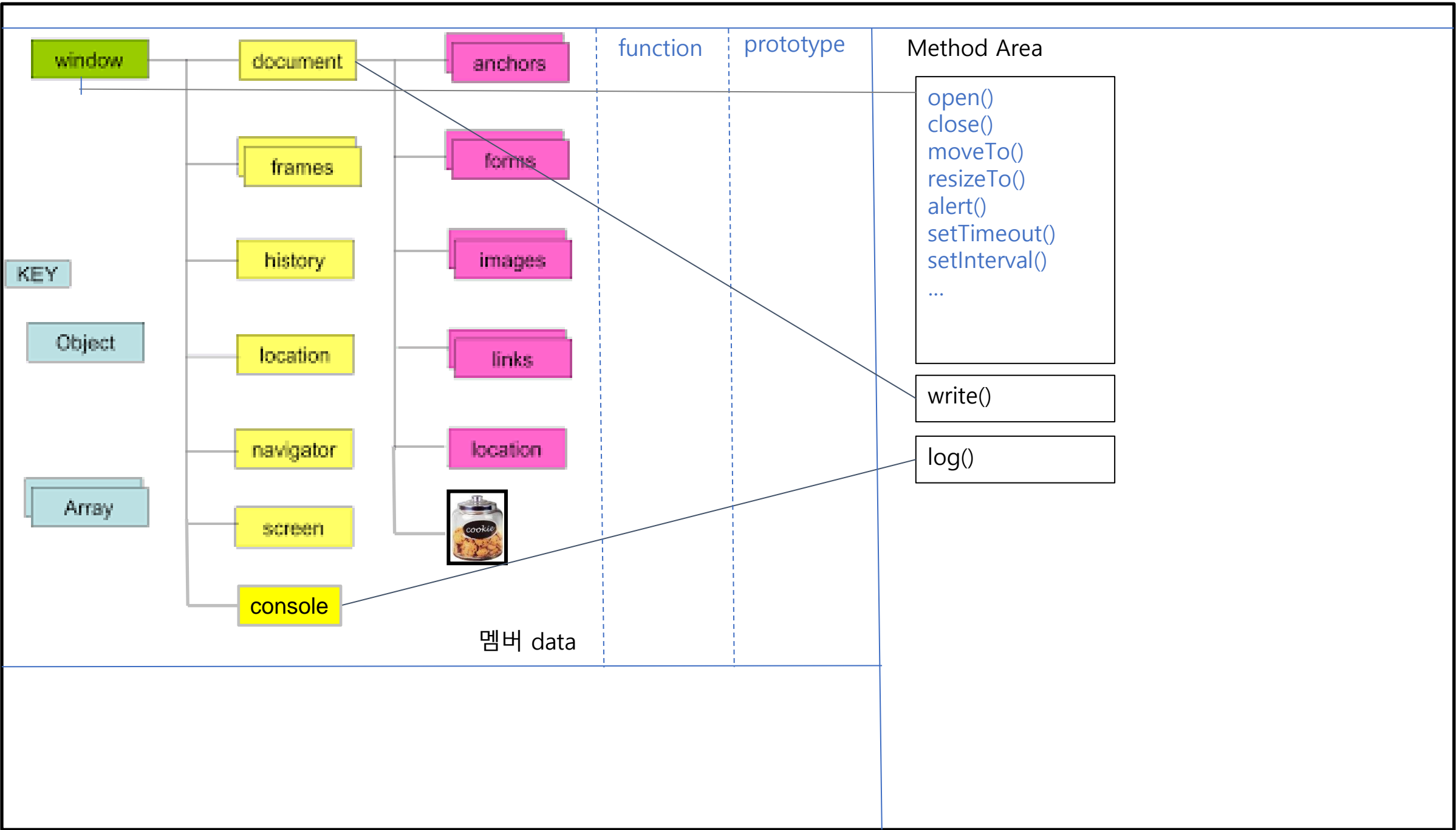
- 2. Object 생성자 이용

```
var p2=new Object(); p2.name="전은수";
```

- 3. 사용자 지정 생성자 이용

```
function Person(name){  
    this.name=name;  
}  
var p3=new Person("전은수");
```

```
var Person=function(name){  
    this.name=name;  
}  
var p4=new Person("전은수");
```



Window

멤버 data

function

prototype

Method Area

document



location

history

screen

navigator

console

...

v1

undefined

open()
close()
moveTo()
resizeTo()
alert()
setTimeout()
setInterval()
...

write()

log()

```
<!DOCTYPE html>
<html>
<body>
<script>
  var v1;
  console.log(window.v1); // undefined
  //var가 선언된 변수들은 변수가 선언된 실행 컨텍스트(execution context) 안에서
  //만들어짐. var가 선언되지 않은 변수들은 항상 전역변수가 됨.
</script>
</body>
</html>
```

Window

멤버 data

document



location

history

screen

navigator

console

...

v1

10

function

prototype

Method Area

open()
close()
moveTo()
resizeTo()
alert()
setTimeout()
setInterval()
...

write()

log()

```
<!DOCTYPE html>
<html>
<body>
<script>
  var v1=10;
  console.log (window.v1); // 10
</script>
</body>
</html>
```

Window

멤버 data

function

prototype

Method Area

document



location

history

screen

navigator

console

...

v1

undefined

open()
close()
moveTo()
resizeTo()
alert()
setTimeout()
setInterval()
...

write()

log()

```
<!DOCTYPE html>
<html>
<body>
<script>
  console.log (window.v1); // undefined
</script>
</body>
</html>
```

Window

멤버 data

function

prototype

Method Area

document



location

history

screen

navigator

console

...

open()
close()
moveTo()
resizeTo()
alert()
setTimeout()
setInterval()
...

write()

```
<!DOCTYPE html>
<html>
<body>
<script>
  console.log (v1); // ReferenceError ➡ .연산자가 없으면 선언과 기본할당이 되지 않음
</script>
</body>
</html>
```

Window

멤버 data

document



location

history

screen

navigator

console

...

v1

10

function

prototype

Method Area

open()
close()
moveTo()
resizeTo()
alert()
setTimeout()
setInterval()
...

write()

log()

```
<!DOCTYPE html>
<html>
<body>
<script>
  v1=10; //var 선언이 없으면 무조건 window의 것
  console.log (window.v1); // 10
</script>
</body>
</html>
```

Window

멤버 data

function

prototype

Method Area

document



location

history

screen

navigator

console

...

--	--

open()
close()
moveTo()
resizeTo()
alert()
setTimeout()
setInterval()
...

write()

log()

```
<!DOCTYPE html>
<html>
<body>
<script>
  v1; //var연산자 없을 때는 할당이 안되면 선언도 안됨
  console.log (window.v1); // 위 라인에서 ReferenceError
</script>
</body>
</html>
```


Window

멤버 data

document

location

history

screen

navigator

console

...

v1

undefined

function

prototype

Method Area

open()
close()
moveTo()
resizeTo()
alert()
setTimeout()
setInterval()
...

write()

```
<!DOCTYPE html>
<html>
<body>
<script>
  console.log (window.v1);
  var v1=10; //이구역의 맨 위로 선언문을
  올리겠다는 것
  console.log (window.v1);
</script>
</body>
</html>
```

hoisting

```
<!DOCTYPE html>
<html>
<body>
<script>
  var v1; //이것을 hoisting이라 함
  console.log (window.v1); // undefined.
  v1=10;
  console.log (window.v1); // 10
</script>
</body>
</html>
```

Window

멤버 data

document



location
history
screen
navigator
console

...
v1 @100

@100

function

prototype

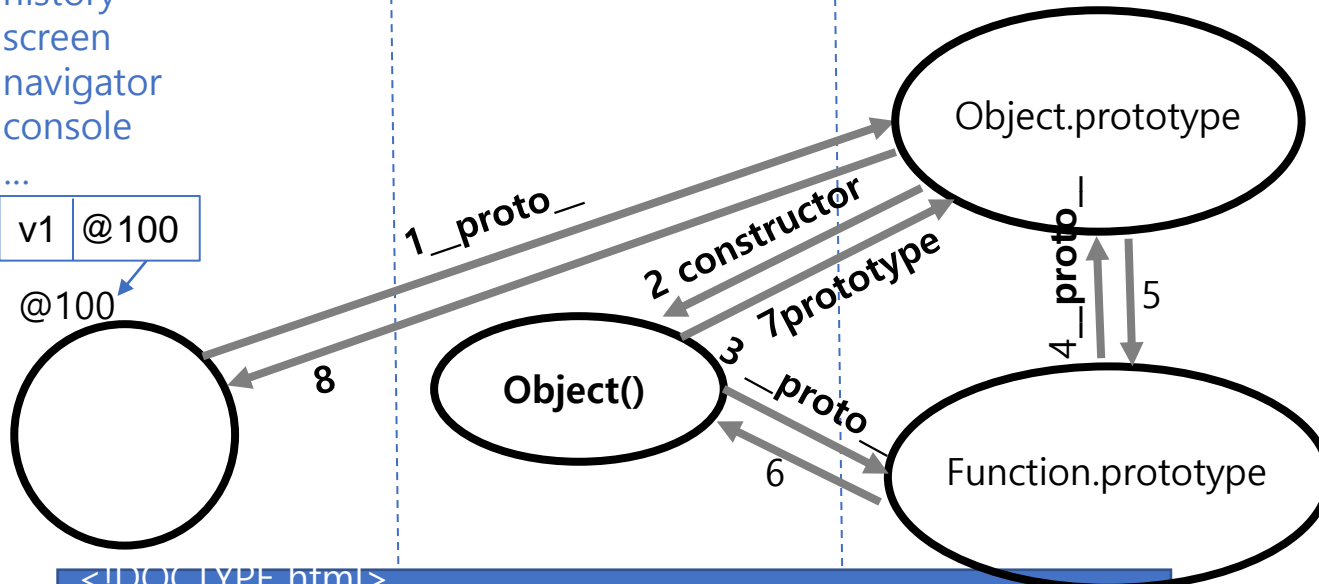
Method Area

open()
close()
moveTo()
resizeTo()
alert()
setTimeout()
setInterval()
...

Object()

write()

log()



```
<!DOCTYPE html>
<html>
<body>
<script>
  var v1=new Object();
  console.log (window.v1); //[object Object]
  console.log(v1.__proto__===Object.prototype);//true
</script>
</body>
</html>
```

Window

멤버 data

document



location

history

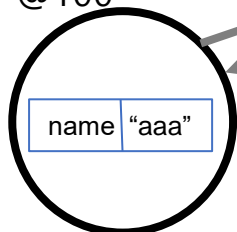
screen

navigator

console

...
v1 @100

@100



function

prototype

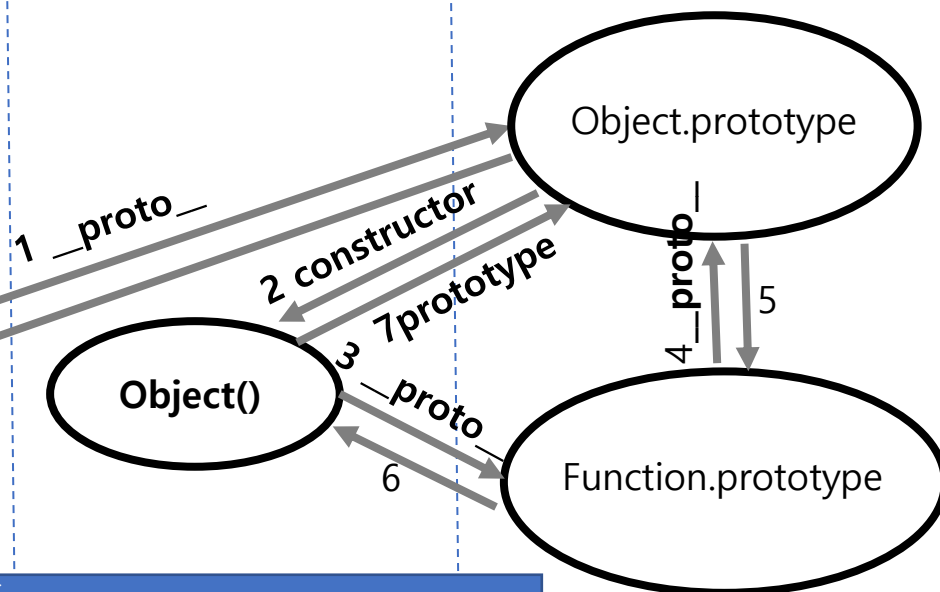
Method Area

open()
close()
moveTo()
resizeTo()
alert()
setTimeout()
setInterval()
...

Object()

write()

log()



```
<!DOCTYPE html>
<html>
<body>
<script>
  var v1=new Object();
  v1.name="aaa";
  console.log (v1.name); // aaa
</script>
</body>
</html>
```

Window

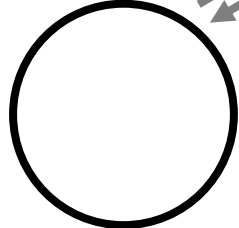
멤버 data

document

location
history
screen
navigator
console

...
p1 @100

@100



function

prototype

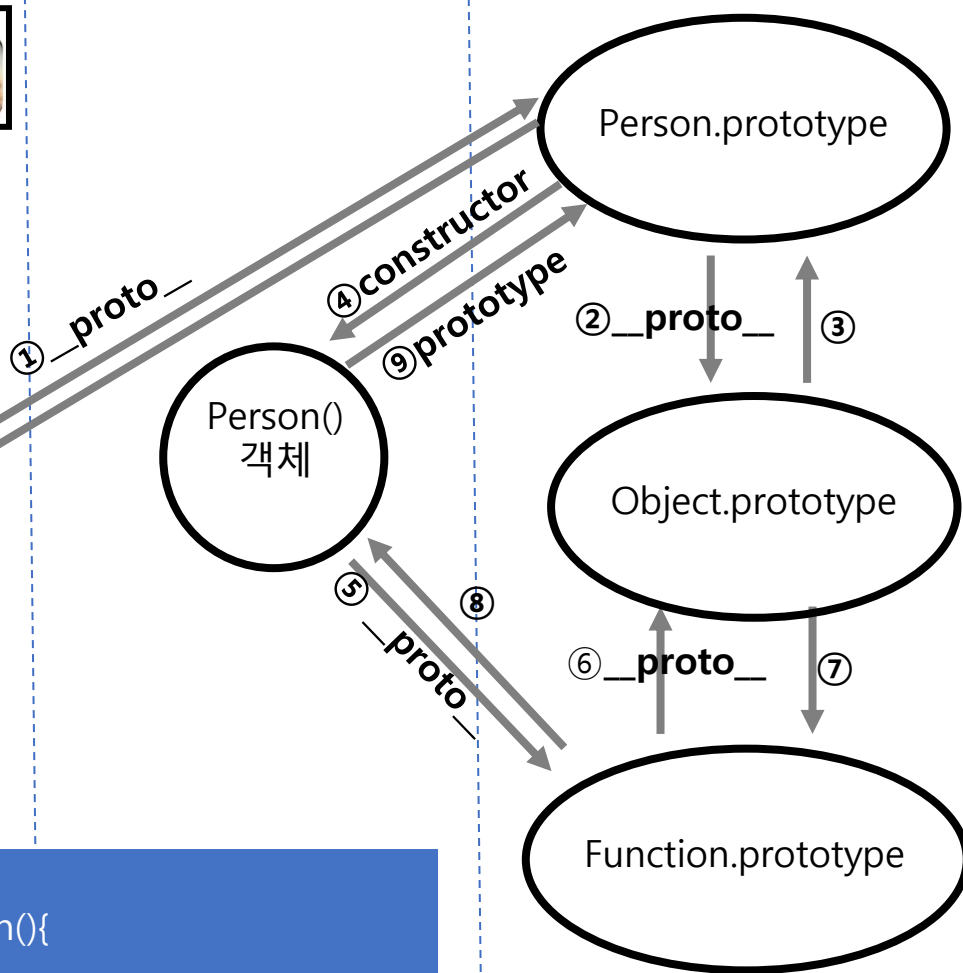
Method Area

open()
close()
moveTo()
resizeTo()
alert()
setTimeout()
setInterval()

...
Object()
Person()

write()

log()



```
<script>
function Person(){
}
var p1=new Person();
console.log (p1);//[object Object]
</script>
```

Window

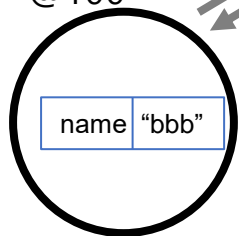
멤버 data

document

location
history
screen
navigator
console

...
p1 @100

@100



```
function Person(name){  
  this.name=name;  
}  
var p1=new Person("bbb");  
console.log(p1.name); //bbb
```

function

prototype

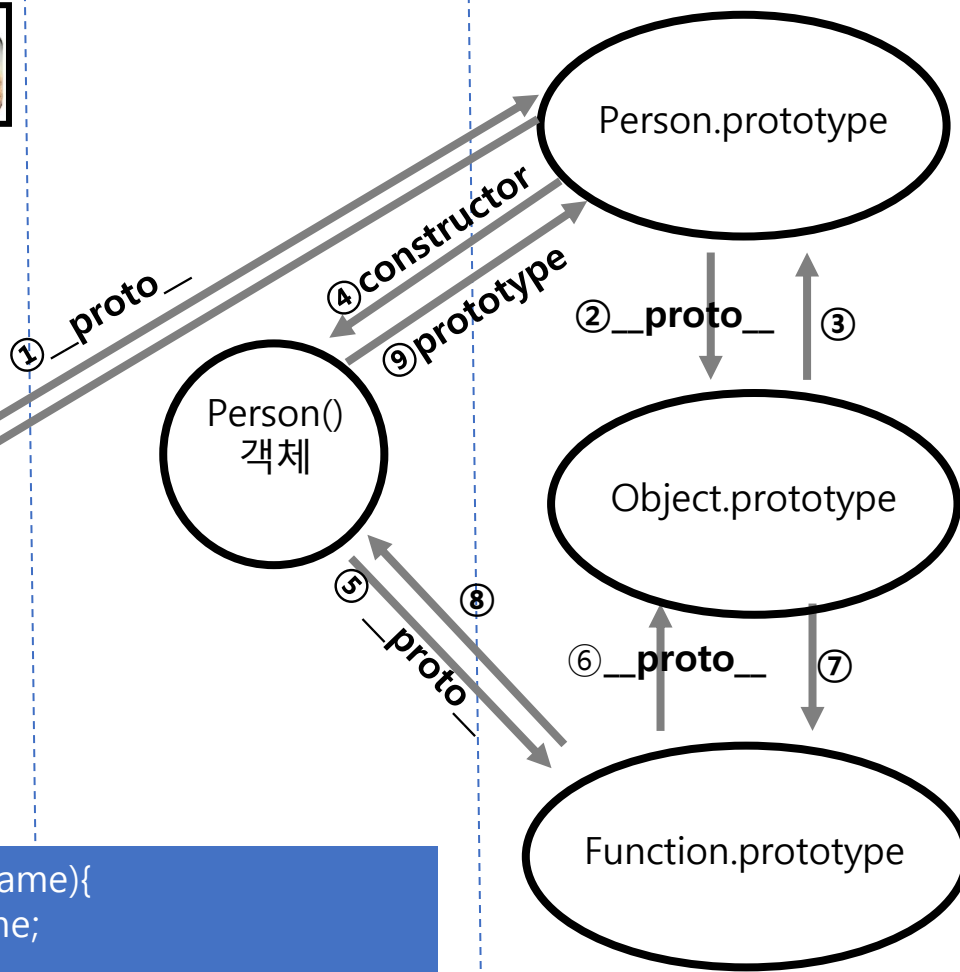
Method Area

open()
close()
moveTo()
resizeTo()
alert()
setTimeout()
setInterval()
...

Object()
Person(name)

write()

log()



Window

멤버 data

document



location

history

screen

navigator

console

...

p1	undefined
name	bbb

function

prototype

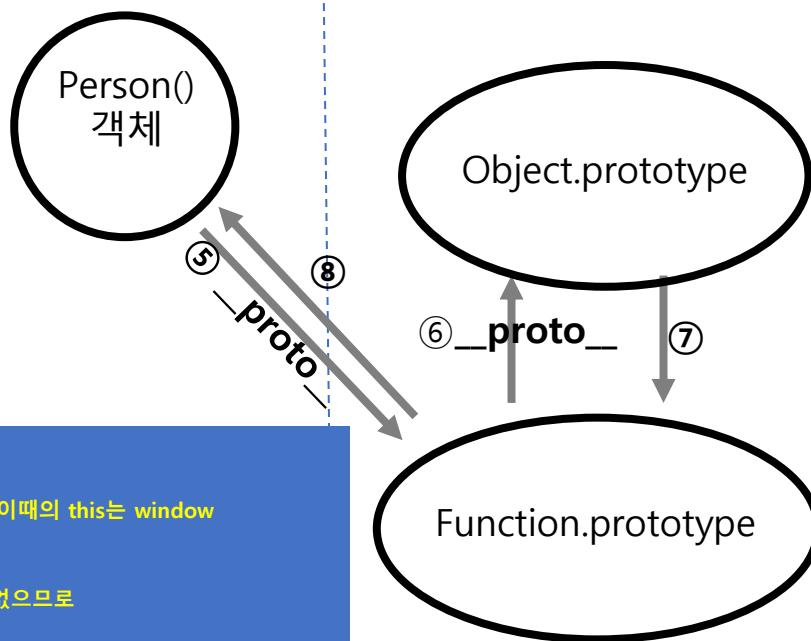
Method Area

open()
close()
moveTo()
resizeTo()
alert()
setTimeout()
setInterval()
...

Object()
Person(name)

write()

log()



```
function Person(name){  
    this.name=name; //이때의 this는 window  
}
```

```
var p1=Person("bbb"); //리턴 값이 없으므로
```

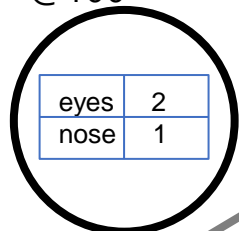
```
console.log(p1); //undefined  
console.log(p1.name); //TypeError  
console.log(window.name); // bbb
```

Window

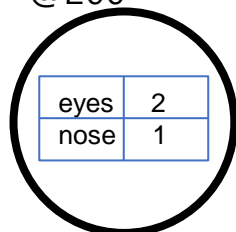
멤버 data

p1	@100
p2	@200

@100

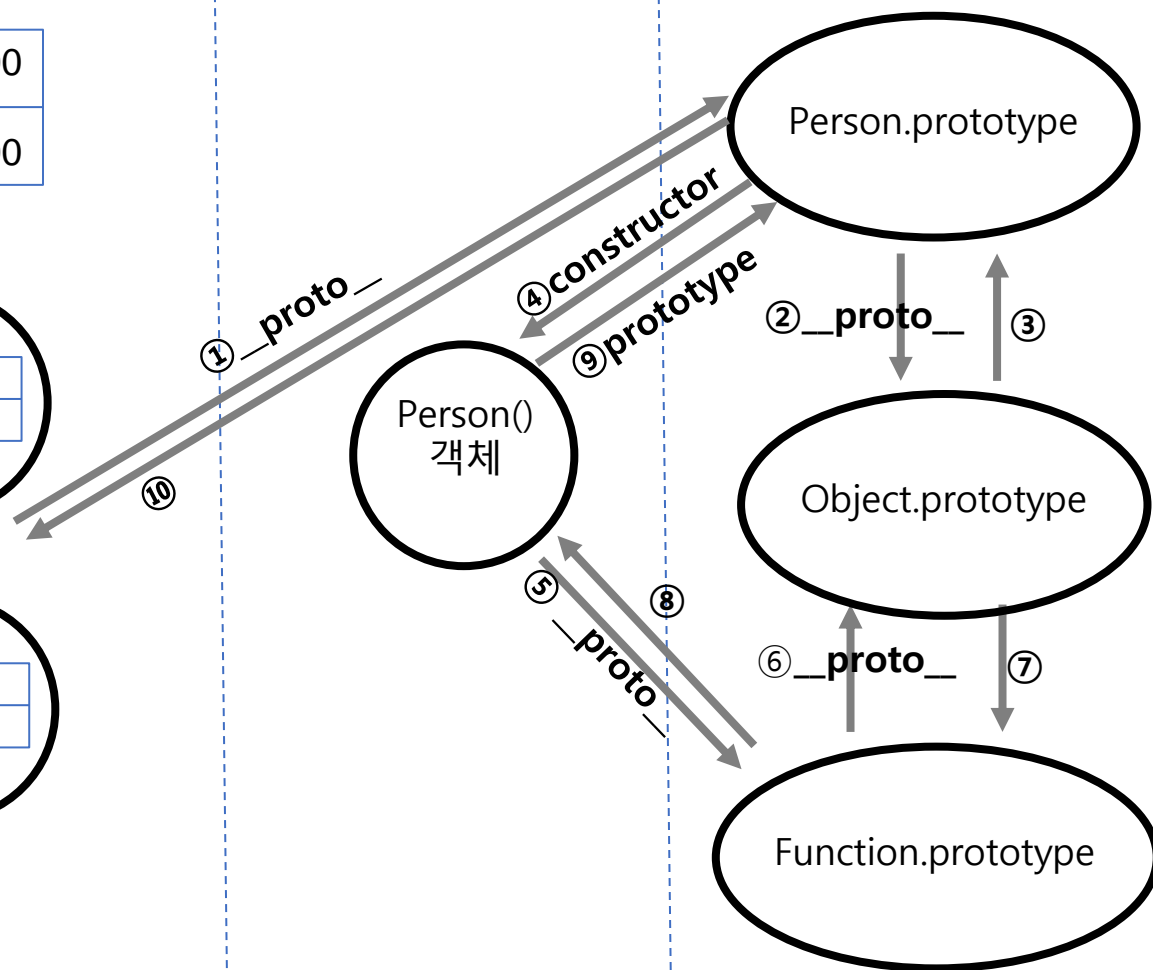


@200



function

prototype



Method Area

open()
close()
moveTo()
resizeTo()
alert()
setTimeout()
setInterval()
...

Object()
Person()

write()

log()

```
function Person(){
    this.eyes=2;
    this.nose=1;
}
var p1=new Person();
var p2=new Person();
console.log(p1.eyes+":"+p1.nose); // 2:1
console.log("<br>");
console.log(p2.eyes+":"+p2.nose); // 2:1
```

Window

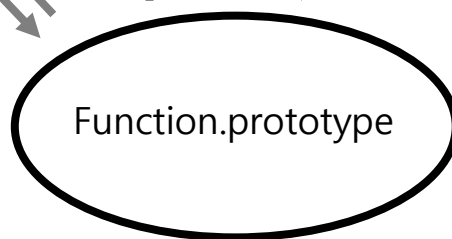
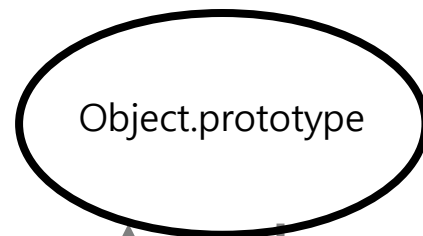
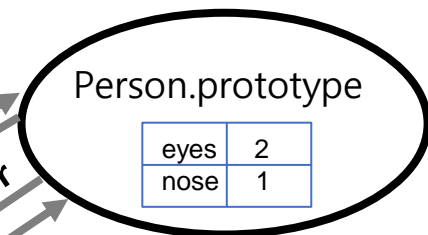
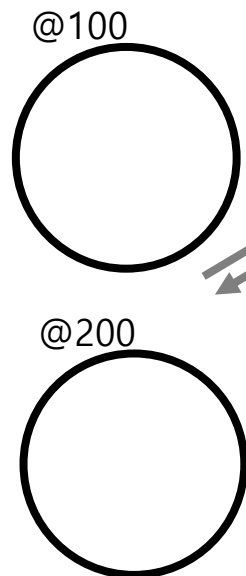
멤버 data

p1	@100
p2	@200

function

prototype

Method Area



open()
close()
moveTo()
resizeTo()
alert()
setTimeout()
setInterval()
...

Object()
Person()

write()

log()

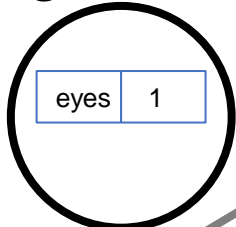
```
function Person(){
}
Person.prototype.eyes=2;
Person.prototype.nose=1;
var p1=new Person();
var p2=new Person();
console.log(p1.eyes+":"+p1.nose+"<br>"); // 2:1
console.log(p2.eyes+":"+p2.nose+"<br>"); // 2:1
```


Window

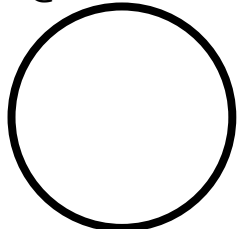
멤버 data

p1	@100
p2	@200

@100



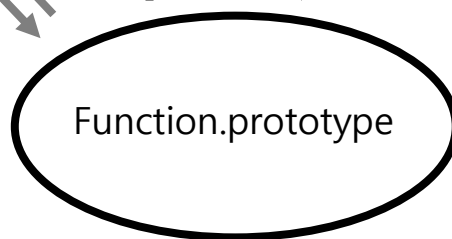
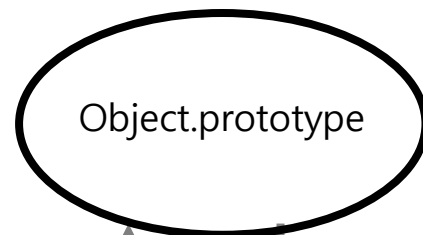
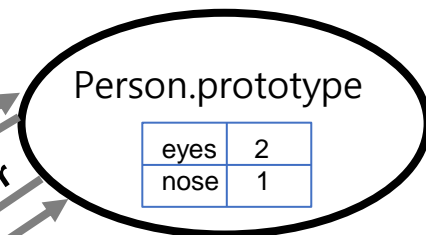
@200



function



prototype



① _proto_

④ constructor

⑨ prototype

② _proto_

③

⑩

⑤

⑧ _proto_

⑥ _proto_

⑦

Method Area

open()
close()
moveTo()
resizeTo()
alert()
setTimeout()
setInterval()

...
Object()
Person()

write()

log()

```
function Person(){
}
Person.prototype.eyes=2;
Person.prototype.nose=1;
var p1=new Person();
var p2=new Person();
console.log (p1.eyes+":"+p1.nose+"<br>"); // 2:1
console.log(p2.eyes+":"+p2.nose+"<br>"); // 2:1
p1.eyes=1;
console.log(p1.eyes+":"+p1.nose+"<br>"); // 1:1
console.log(p2.eyes+":"+p2.nose+"<br>"); // 2:1
console.log(Person.prototype.eyes+"<br>"); // 2
console.log(p1.eyes===Person.prototype.eyes);// false
```

Window

멤버 data

function

prototype

Method Area

p1 @100

@100

eyes	3
nose	4

8

1 __proto__

2 constructor

7 prototype

3 __proto__

6

Object()

Object.prototype

4 __proto__

5

Function.prototype

open()
close()
moveTo()
resizeTo()
alert()
setTimeout()
setInterval()
...

Object()

write()

log()

```
var p1 = {  
  eyes: 3,  
  nose: 4  
}  
console.log(p1.__proto__ === Object.prototype); // true  
console.log(Object === p1.constructor); // true
```

Window

멤버 data

function

prototype

Method Area

p1 @100

@100

eyes	3
nose	4

8

1 _proto_

2 constructor

7 prototype

Object()

3 _proto_

6

Object.prototype

4 _proto_

5

Function.prototype

open()
close()
moveTo()
resizeTo()
alert()
setTimeout()
setInterval()
...
Object()

write()

log()

```
function  
hasOwnProperty(property){  
  property가 있는지 찾아서  
  리턴  
}
```

```
var p1 = {  
  eyes: 3,  
  nose: 4  
}  
console.log(p1.hasOwnProperty('eyes')); // true
```

Window

멤버 data

document



location

history

screen

navigator

console

...

fn

undefined

function

prototype

Method Area

open()
close()
moveTo()
resizeTo()
alert()
setTimeout()
setInterval()

...

Object()

(){ ... }

fn2() { ... }

write()

log()

hoisting

```
fn(); //error : fn is not a function  
var fn = function () { alert("test!"); }
```

```
fn2(); //ok  
function fn2() { alert("test!"); }
```