

사용자 정의 객체



소프트웨어학부
진혜진

사용자 정의 객체

2

- 내장 객체 뿐만 아니라 사용자가 필요할 때마다 자신의 객체를 정의해서 사용할 수 있음
 - ▣ 예) 여러 가지 책 정보를 함께 모아서 book변수 지정할 때 book이 사용자 정의 객체임
 - book
 - 제목 : 웹 클라이언트, 저자 : 국민대 , 쪽수 : 300쪽, 가격 : 20,000원

```
> var book = {  
  title : "웹 클라이언트",  
  author : "국민대",  
  pages : 300,  
  price : 20000  
}
```



사용자 정의 객체

3

```
> var book = {  
  title : "웹 클라이언트",  
  author : "국민대",  
  pages : 300,  
  price : 20000  
}
```

- 리터럴 표기법으로 사용해 객체를 만드는 방법
 - 객체 리터럴을 사용해 객체를 정의할 때는 { }안에 속성이름:값을 하나의 쌍으로 지정함
- 속성이 여러 개 일 경우 속성이름:값 쌍마다 쉼표를 넣어 구분함.
 - 객체를 정의해 변수에 할당할 때는 닫는 중괄호 다음에 ;을 붙여줌
- 함수 이름: 다음에 함수를 정의하며, 함수 이름:function(){...}형식을 사용함

사용자 정의 객체

4

리터럴 표기법 사용해 book 객체 만들기

```
> var book={  
  title : "웹 클라이언트",  
  author : "국민대",  
  pages : 300,  
  price : 20000,  
  
  info:function(){  
    alert(this.title+"책의 저자는"+this.author+"입니다.");  
  }  
};
```

< undefined

> book.price

< 20000

> book.author

< "국민대"

> book.info()

www.google.com 내용:

웹 클라이언트책의 저자는국민대입니다.

확인

사용자 정의 객체

5

- 사용자 정의 객체에서도 속성 값을 가져오려면 .를 사용함

```
> book.author  
< "국민대"  
> book.info()
```

- 새로운 속성을 추가하려면
 - ▣ field속성을 추가한 다음 원하는 값을 넣으면 됨
- 객체를 만들 때 공통으로 들어가는 속성과 함수가 별로 없다면 굳이 여러 객체를 하나의 틀로 묶을 필요는 없다.
- 이렇게 각 객체를 따로 정의하는 것이 편할 때 리터럴 표기법을 사용한다.

사용자 정의 객체

6

```
> book.field="IT"
< "IT"

> book
< {title: "웹 클라이언트", author: "국민대", pages: 300, price: 20000, info: f, ...} i
  author: "국민대"
  field: "IT"
  ▶ info: f ()
    pages: 300
    price: 20000
    title: "웹 클라이언트"
  ▶ __proto__: Object
```

예) 리터럴 표기법으로 책 정보 객체 만들기

7

□ object.js 파일

▣ book 객체 선언

- 5개의 속성과 1개의 함수
- showStock() : 재고를 화면에 표시하는데 사용함
- 함수 정의할 때 객체의 속성 값을 사용하려면 this 예약어와 함께 입력하면 됨
 - this : 함수를 포함하는 현재 객체(book)를 가리킴

```
<!DOCTYPE html>
<head>
  <title>사용자 정의 객체</title>
  <link rel="stylesheet" href="obj.css">
</head>
<body>
  <div id="display"></div>
  <script src="object.js"></script>
</body>
</html>
```

```
#display {
  width:500px;
  margin:20px auto;
  padding:10px;
  border:1px solid #ccc;
  text-align:center;
  font-size:20px;
  line-height:30px;
}
```

```
var book={
  title:"웹 클라이언트",
  author:"국민대",
  pages:300,
  price:20000,
  quantity:3,
```

웹 클라이언트 도서는 3개 남아있습니다.

```
showStock:function(){
  document.querySelector('#display').innerHTML=this.title + " 도서는 " + this.quantity + "개 남아있습니다.";
}
};
```

```
book.showStock();
```


이 페이지 내용:

웹 클라이언트 도서의 가격은 20000입니다.

확인

```
var book={
  title:"웹 클라이언트",
  author:"국민대",
  pages:300,
  price:20000,
  quantity:3,

  showStock:function(){
    document.querySelector('#display').innerHTML=this.title + " 도서는 " + this.quantity + "개 남아있습니다.";
  },

  showPrice:function(){
    alert(this.title + " 도서의 가격은 " + this.price + "입니다.");
  }
};

book.showPrice();
book.showStock();
```

사용자 정의 객체

10

- ▣ 리터럴 표기법을 사용해 객체를 만드는 방법은 정해진 값을 가진 객체를 한 번만 만들어 냄
 - 이 방법으로 도서 관리 프로그램을 만든다면 매번 똑같은 객체와 똑같은 속성과 함수를 입력하고, 책마다 그 값을 다르게 넣어 새롭게 객체를 만들어야 함
 - 리터럴 표기법은 객체 틀을 사용하지 않고 개별적으로 객체를 선언하고 사용하는 방법임
- ▣ 항상 필요한 속성과 함수는 틀처럼 미리 만들어두고, 필요할 때마다 그 틀을 복제한 인스턴스를 만들어 책 정보 값을 담는 것이 편리함
 - 이때 생성자 함수를 사용해 객체를 만듦

사용자 정의 객체

11

□ 생성자 함수를 사용해 객체를 만드는 방법

□ 생성자 함수

- 객체를 만들어 내는 함수를 말함
- function 예약어를 사용해 선언함
- 생성자 함수 안에 객체의 속성과 함수를 정의할 때는 this 예약어 다음에 .와 속성 이름을 입력함
- this가 가리키는 것은 선언하고 있는 객체 자체임
- 객체의 속성은 객체에서 사용하는 변수
- 객체의 함수는 해당 객체에서 사용하는 함수

사용자 정의 객체

12

□ Book 객체 정의

▣ this는 Book 객체를 가리킴

```
> function Book(author, pages, price, title) {  
    this.author = author;  
    this.pages = pages;  
    this.price = price;  
    this.title = title;  
}
```

사용자 정의 객체

13

- Book 객체 만들었으면 다음과 같이 인스턴스를 만들 수 있음

```
> kookBook = new Book("국민대", 300, 30000, "웹 클라이언트")
< Book {author: "국민대", pages: 300, price: 30000, title:
  "웹 클라이언트"} i
  author: "국민대"
  pages: 300
  price: 30000
  title: "웹 클라이언트"
  ▶ __proto__: Object

> kookBook.title
< "웹 클라이언트"
```

예) 생성자 함수로 도서 목록 객체 만들기

14

- 도서 관리 프로그램을 만들려면 도서마다 제목, 저자, 가격 등 포함해야 할 정보가 많음
- 하나의 변수에 포함할 정보가 많을 경우 객체 형태로 저장하는 것이 좋음
- Book 객체를 사용해 정보를 저장하고 가져오는 방법
 - ▣ 1. Book.js 파일에 Book 객체 정의
 - ▣ 기본 틀
 - `function Book() {`
`}`
 - `function Book() {...}`를 Book 객체 생성자라고 부름

사용자 정의 객체

15

Book.js

```
function Book(title, author, pages, price) {  
  this.title = title;  
  this.author = author;  
  this.pages = pages;  
  this.price = price;  
}
```

- 객체 정의했으면 인스턴스를 만들어야 그 객체를 비로소 사용할 수 있음
 - ▣ 인스턴스를 만들 때는 Book 객체를 만들 때 사용한 매개 변수에 넣을 값도 함께 지정해야 함
 - html 객체는 Book 객체의 인스턴스임

```
var html = new Book("웹 클라이언트", "국민대", 300, 20000);
```

사용자 정의 객체

16

- html 객체에 어떤 값이 저장되어 있는 지 확인
 - ▣ html은 Book 객체로 찍어 낸 인스턴스이기 때문에 콘솔창에 html이 Book 객체로 나타남

```
> html
< Book {title: "웹 클라이언트", author: "국민대", pages: 300, price: 20000} ⓘ
  ▶ author: "국민대"
  ▶ pages: 300
  ▶ price: 20000
  ▶ title: "웹 클라이언트"
  ▶ __proto__: Object
```


사용자 정의 객체

17

□ Book.js에 인스턴스 추가

```
var html = new Book("웹 클라이언트", "국민대", 300, 20000);  
var network = new Book("네트워크", "진혜진", 350, 24000);  
var python = new Book("파이썬", "김혜숙", 200, 15000);
```

□ 책 제목만 표시하는 코드 작성

- BookList배열을 만들고 객체 저장
- `var BookList = [html, network, python]`

> BookList

```
< ▼ (3) [Book, Book, Book] ⓘ  
  ▶ 0: Book {title: "웹 클라이언트", author: "국민대", pages: 300...  
  ▶ 1: Book {title: "네트워크", author: "진혜진", pages: 350, pri...  
  ▶ 2: Book {title: "파이썬", author: "김혜숙", pages: 200, pric...  
    length: 3  
  ▶ __proto__: Array(0)
```

사용자 정의 객체

18

```
> BookList
< ▼ (3) [Book, Book, Book] ⓘ
  ▼ 0: Book
    author: "국민대"
    pages: 300
    price: 20000
    title: "웹 클라이언트"
    ▶ __proto__: Object
  ▼ 1: Book
    author: "진혜진"
    pages: 350
    price: 24000
    title: "네트워크"
    ▶ __proto__: Object
  ▼ 2: Book
    author: "김혜숙"
    pages: 200
    price: 15000
    title: "파이썬"
    ▶ __proto__: Object
  length: 3
  ▶ __proto__: Array(0)
```

사용자 정의 객체

19

- BookList 배열에 저장된 도서 객체에서 책 제목만 표시

```
<!DOCTYPE html>
<head>
  <meta charset="UTF-8">
  <title>도서 관리 프로그램 - 객체 만들기</title>
  <link rel="stylesheet">
</head>
<body>
  <script src="Book.js"></script>
</body>
</html>
```

사용자 정의 객체

20

책 제목으로 살펴보기

웹 클라이언트

네트워크

파이썬

Book.js

```
function Book(title, author, pages, price) {  
    this.title = title;  
    this.author = author;  
    this.pages = pages;  
    this.price = price;  
}  
  
var html = new Book("웹 클라이언트", "국민대", 300, 20000);  
var network = new Book("네트워크", "진혜진", 350, 24000);  
var python = new Book("파이썬", "김혜숙", 200, 15000);  
  
var BookList = [html, network, python];  
  
document.write("<h1>책 제목으로 살펴보기</h1>");  
for(var i=0; i<BookList.length; i++) {  
    document.write("<p>" + BookList[i].title + "</p>");  
}
```

사용자 정의 객체

21

□ new Object()로 객체를 만드는 방법

- 자바스크립트 코어 객체 중 Object타입을 이용하면 사용자 객체를 손쉽게 만들 수 있음
- 1. new Object()로 빈 객체 생성
- 2. 빈 객체에 속성 추가
 - 새로운 속성추가(속성 이름과 초기값 지정)
- 3. 빈 객체에 메소드 추가
 - 메소드로 사용할 함수 미리 작성
 - 새 메소드 추가(메소드 이름에 함수 지정)

사용자 정의 객체

22

□ 속성 추가하는 코드

- ▣ 속성 추가 시 이름과 함께 값을 주어야 함

```
> var book = new Object();  
  book.title = "network";  
  book.author = "jin";  
  book.pages = 350;  
  book.price = 24000;
```

- ▣ 4개의 속성을 가진 book 객체가 만들어짐

□ 사용자 객체의 속성 사용

- ▣ 만들어진 book 객체의 속성은 다양하게 접근 가능함

```
> var name = book["author"];  
  book["price"] = 20000;  
  book.price += 5000;
```

사용자 정의 객체

23

□ 사용자 객체에 메소드 만들기

- ▣ 객체에 메소드를 만들려면 먼저 메소드 이름과 동일한 함수를 작성함
- ▣ 그리고 객체의 메소드 이름을 함수 이름으로 초기화하면 됨
- ▣ book 객체에 totalPrice()메소드 만드는 소스코드

```
function totalPrice(quantity) {  
    this.price *= quantity;  
}  
var book = new Object();  
book.title = "network";  
book.author = "jin";  
book.pages = 350;  
book.price = 24000;  
book.totalPrice = totalPrice;
```

사용자 정의 객체

24

□ this

- totalPrice()함수에서 this는 이 객체를 뜻함
- this.price는 이 객체의 price속성을 뜻함
- this 없이 price만 작성하면
 - 전역 변수나, 지역 변수 price를 접근하는 것임
- 객체의 멤버 변수인 속성을 접근할 때는 반드시 this를 사용해야 함


```
<!DOCTYPE html>
<html><head><title>new Object()로 사용자 객체 만들기</title>
<script>
function totalPrice(quantity) {
    this.price *= quantity;
}
var book = new Object();
book.title = "network";
book.author = "jin";
book.pages = 350;
book.price = 24000;
book.totalPrice = totalPrice;
</script></head>
<body>
<h3>new Object()로 사용자 객체 만들기</h3>
<hr>
<script>
    // 객체 활용
    document.write("book : ");
    document.write(book.title + ", ");
    document.write(book.author + ", ");
    document.write(book.pages + ", ");
    document.write(book.price + "<br>");
    book.totalPrice(2);
    document.write("총도서 금액은 " + book.price + "원 입니다.<br>");
</script>
</body></html>
```

new Object()로 사용자 객체 만들기

book : network, jin, 350, 24000
총도서 금액은 48000원 입니다.