

Chapter 11

자바스크립트 객체

C.ontents

01 객체의 이해

02 객체 생성

03 배열 객체와 Date 객체

04 문서 객체 모델

학습목표

- 객체 모델링의 의미를 알고 프로그램에 적용할 수 있다.
- 사용자 정의 객체를 생성할 수 있다.
- 배열 객체와 Date 객체를 활용하여 프로그램을 작성할 수 있다.
- 문서 객체를 생성하여 동적인 웹 문서를 만들 수 있다.

1. 객체 모델링

● 객체

- 세상에 존재하는 모든 것

● 자동차 객체의 모델링



객체	속성	메소드
car	car.name="Sonata" car.speed=100 car.color="white" car.door=4	car.start(){ } car.accel(){ } car.break(){ } car.transe(){ }

2. 자바스크립트 객체

● 자바스크립트 객체

- 사용자 정의 객체: 사용자가 직접 객체의 속성과 메소드를 정의하여 사용하는 객체(예: Car(), House(), Hotel())
- 내장 객체: 자바스크립트 프로그램 자체에서 정의하여 사용자에게 제공하는 객체(예: Object(), Array(), Date())

● 내장 객체의 종류

- 브라우저 객체 모델(BOM, Browser Object Model): 웹 브라우저의 각종 요소를 객체로 표현
- 문서 객체 모델(DOM, Document Object Model): 웹 문서의 각종 요소를 객체로 표현
- 전역 자바스크립트 객체(Global JavaScript Objects): 자바스크립트 프로그램 전체에서 사용하는 내장 객체

2. 자바스크립트 객체

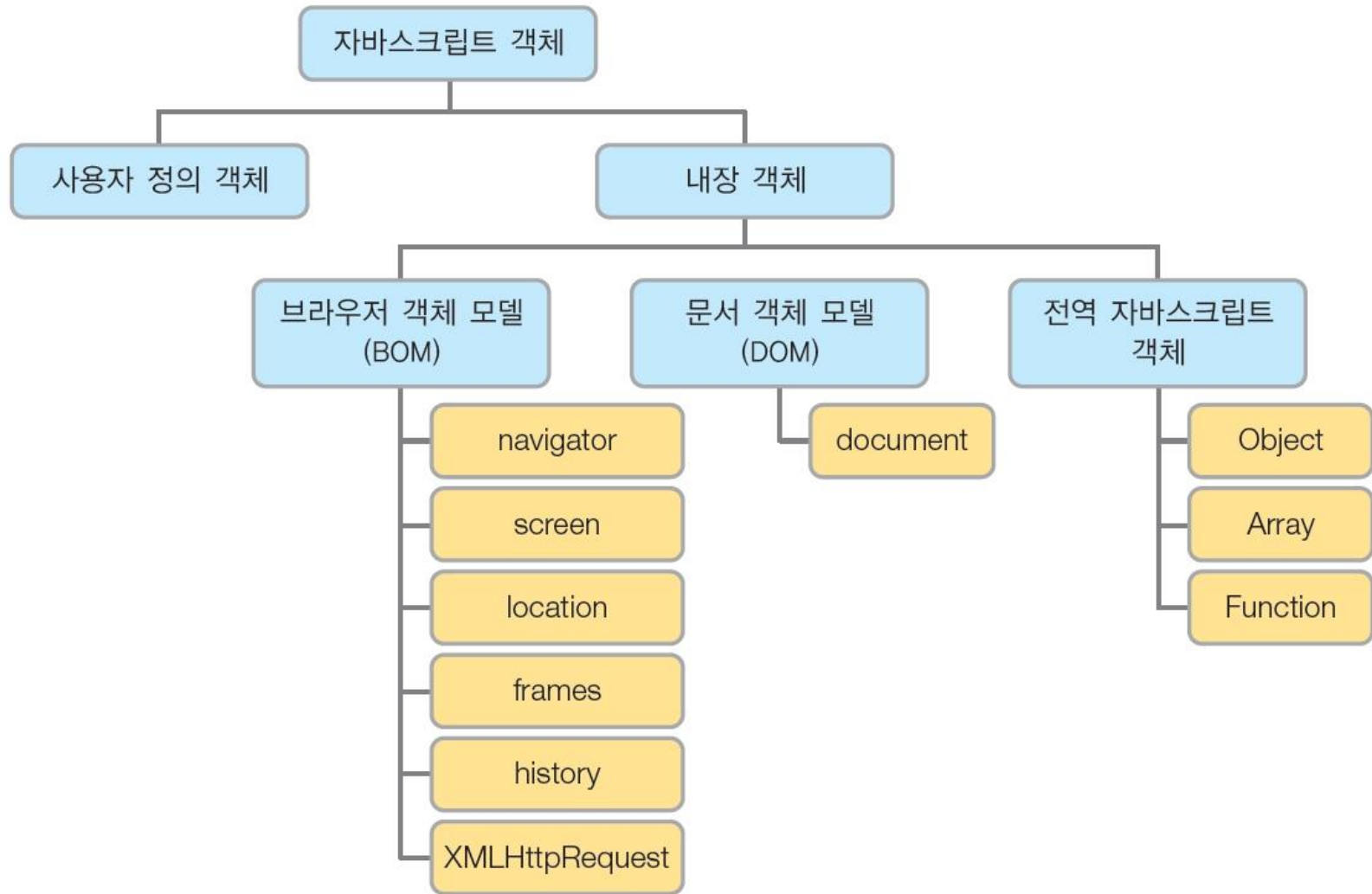


그림 11-2 자바스크립트 객체의 종류

1. 객체 변수를 이용하는 방법

● 객체 변수를 이용하여 객체 생성

```

      객체 변수   속성값
      ↓         ↓
var car = {
속성명 → name: "Sonata",
        speed: 100,
        color: "white",
        door: 4,
        메소드명 → start: function() {
                    return this.speed+10;
                }
};
    
```

속성 정의

메소드 정의
(함수 형식)

● 객체 속성 접근 방법

방법	사용 예	방법	사용 예
객체명.속성명	car.name	객체명['속성명']	car['name']
	car.speed		car['speed']
	car.color		car['color']

1. 객체 변수를 이용하는 방법

표 11-3 자바스크립트로 제어할 요소를 찾아 결과를 출력하는 방법

방법	사용 예	의미
innerHTML 속성 이용	<code>document.getElementById("carname").innerHTML;</code>	웹 문서 안에서 아이디가 "carname"인 요소를 찾아 내용을 출력한다.
textContent 속성 이용	<code>var cname=document.getElementById("carname"); cname.textContent;</code>	웹 문서 안에서 아이디가 "carname"인 요소를 찾아서 cname 변수에 반환한 후 cname 변수의 내용을 출력한다.

예제 11-1 속성만 가진 객체 만들기

ch11/01_obj.html

```
<body>
  <p id="var1"></p>
  <p id="var2"></p>
  <p id="var3"></p>
  <script>
    var car={name: 'Sonata', speed: 100, color: 'white'};
    document.getElementById("var1").innerHTML="자동차 이름 : " + car['name'];
    document.getElementById("var2").innerHTML="자동차 속도 : " + car.speed;
    document.getElementById("var3").innerHTML="자동차 색상 : " + car.color;
  </script>
</body>
```

자동차 이름 : Sonata

자동차 속도 : 100

자동차 색상 : white

1. 객체 변수를 이용하는 방법

예제 11-2 메소드를 호출하여 연산 결과 출력하기

ch11/02_obj.html

```
<body>
  <p id="msg1"></p>
  <p id="msg2"></p>
  <p id="msg3"></p>
  <script>
    var obj={
      m1: function() {
        return "Hello Sonata";
      },
      m2: function(a) {
        var result=a;
        return result;
      },
      m3: function(a, b) {
        var result=a+b;
        return result;
      }
    };
    document.getElementById( "msg1" ).innerHTML=obj.m1();
    document.getElementById( "msg2" ).innerHTML=obj.m2(100);
    document.getElementById( "msg3" ).innerHTML=obj.m3(100, 200);
  </script>
</body>
```

Hello Sonata

100

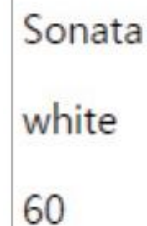
300

1. 객체 변수를 이용하는 방법

예제 11-3 자동차 객체 생성하기

ch11/03_obj.html

```
<body>
  <p id="carname"></p>
  <p id="carcolor"></p>
  <p id="carspeed"></p>
  <script>
    var car={
      name: 'Sonata',
      speed: 50,
      color: 'white',
      start: function() {
        return this.speed+10;
      }
    };
    var cname=document.getElementById("carname");
    cname.textContent=car.name;
    var colname=document.getElementById("carcolor");
    colname.textContent=car.color;
    var cspeed=document.getElementById("carspeed");
    cspeed.textContent=car.start();
  </script>
</body>
```



1. 객체 변수를 이용하는 방법

예제 11-4 자동차의 속도 조절하기

ch11/04_obj.html

```
<body>
  <p id="upspeed"></p>
  <p id="downspeed"></p>
  <script>
    var car={
      name: 'Sonata',
      speed: 50,
      color: 'white',
      speedup: function() {
        return this.speed+10;
      },
      speeddown: function() {
        var low=this.speed-10;
        return low;
      }
    };
    var upspeed=document.getElementById( "upspeed" );
    upspeed.textContent='속도 증가 : ' + car.speedup();
    var downspeed=document.getElementById( "downspeed" );
    downspeed.textContent='속도 감소 : ' + car.speeddown();
  </script>
</body>
```

속도 증가 : 60

속도 감소 : 40

1. 객체 변수를 이용하는 방법

예제 11-5 자동차의 속도 제어하기

ch11/05_obj.html

```

<body>
  <p id="upspeed"></p>
  <p id="downspeed"></p>
  <script>
    var car={
      name: 'Sonata',
      speed: 100,
      color: 'white',
      speedup: function(a) {
        var sp=this.speed+a;
        if(sp>=300) {
          sp=50;
          return sp;
        }
        else {
          return sp;
        }
      },
      speeddown: function(a) {
        var sp=this.speed-a;
        if(sp<0) {
          sp=0;
          return sp;
        }
        else {
          return sp;
        }
      }
    };
  </script>
</body>

```

```

var upspeed=document.getElementById("upspeed");
upspeed.textContent='속도 증가 : ' + car.speedup(100);
var downspeed=document.getElementById("downspeed");
downspeed.textContent='속도 감소 : ' + car.speeddown(30);
</script>
</body>

```

속도 증가 : 200

속도 감소 : 70

2. 생성자 함수를 이용하는 방법

● Object 함수 이용

```
var car=new Object();      // 객체 생성
car.name='Sonata';         // 속성 정의
car.speed=100;
car.color='blue';
car.speedup=function() {   // 메소드 정의
    return this.speed+10;
};
```

2. 생성자 함수를 이용하는 방법

예제 11-6 Object 함수를 이용하여 객체 만들기

ch11/06_obj.html

```
<body>
  <p id="carname"></p>
  <p id="carcolor"></p>
  <p id="carspeed"></p>
  <script>
    var car=new Object();
    car.name='Sonata';
    car.speed=100;
    car.color='blue';
    car.speedup=function() {
      return this.speed+10;
    };
    var cname=document.getElementById("carname");
    cname.textContent='자동차 이름 : ' + car.name;
    var colname=document.getElementById("carcolor");
    colname.textContent='자동차 색상 : ' + car.color;
    var cspeed=document.getElementById("carspeed");
    cspeed.textContent='자동차 속도 : ' + car.speedup();
  </script>
</body>
```

자동차 이름 : Sonata
자동차 색상 : blue
자동차 속도 : 110

2. 생성자 함수를 이용하는 방법

● 생성자 함수 정의

```
function Car(name, color, speed) {  
  this.name=name;  
  this.color=color;  
  this.speed=speed;  
  this.speedup=function() {  
    return this.speed+10;  
  };  
  this.speeddown=function() {  
    return this.speed-10;  
  };  
}
```

2. 생성자 함수를 이용하는 방법

예제 11-7 생성자 함수 정의 후 객체 만들기

ch11/07_obj.html

```
<body>
  <p>[Hong's Car]</p>
  <p id="carname"></p>
  <p id="carcolor"></p>
  <p id="carspeed"></p>
  <p>[Kim's Car]</p>
  <p id="carname2"></p>
  <p id="carcolor2"></p>
  <p id="carspeed2"></p>
  <script>
    function Car(name, color, speed) {
      this.name=name;
      this.color=color;
      this.speed=speed;
      this.speedup=function() {
        return this.speed+10;
      };
      this.speeddown=function() {
        return this.speed-10;
      };
    }
    var Hongcar=new Car('Sonata', 'blue', 100);
    var Kimcar=new Car('Jeep', 'red', 70);
```

```
var cname=document.getElementById("carname");
cname.textContent='자동차 이름 : ' + Hongcar.name;
var colname=document.getElementById("carcolor");
colname.textContent='자동차 색상 : ' + Hongcar.color;
var cspeed=document.getElementById("carspeed");
cspeed.textContent='자동차 속도 : ' + Hongcar.speedup();
var cname=document.getElementById("carname2");
cname.textContent='자동차 이름 : ' + Kimcar.name;
var colname=document.getElementById("carcolor2");
colname.textContent='자동차 색상 : ' + Kimcar.color;
var cspeed=document.getElementById("carspeed2");
cspeed.textContent='자동차 속도 : ' + Kimcar.speedup();
</script>
</body>
```

```
[Hong's Car]
자동차 이름 : Sonata
자동차 색상 : blue
자동차 속도 : 110

[Kim's Car]
자동차 이름 : Jeep
자동차 색상 : red
자동차 속도 : 80
```


2. 생성자 함수를 이용하는 방법

예제 11-8 이미 생성된 객체에 속성 추가 및 삭제하기

ch11/08_obj.html

```
<body>
  <p>[Hong's Car]</p>
  <p id="hong1"></p>
  <p id="hong2"></p>
  <p id="hong3"></p>
  <p>[Kim's Car]</p>
  <p id="data1"></p>
  <p id="data2"></p>
  <p id="data3"></p>
  <p id="data4"></p>
  <script>
    function Car(name, color, speed) {
      this.name=name;
      this.color=color;
      this.speed=speed;
      this.speedup=function() {
        return this.speed+10;
      };
      this.speeddown=function() {
        return this.speed-10;
      };
    }
    var Hongcar=new Car('Sonata', 'blue', 100);
    var Kimcar=new Car('Jeep', 'red', 70);
    Kimcar.price='3천만 원';
    delete Kimcar.color;
```

```
var cname=document.getElementById("hong1");
cname.textContent='자동차 이름 : ' + Hongcar.name;
var colname=document.getElementById("hong2");
colname.textContent='자동차 색상 : ' + Hongcar.color;
var cspeed=document.getElementById("hong3");
cspeed.textContent='자동차 속도 : ' + Hongcar.speedup();
var cname=document.getElementById("data1");
cname.textContent='자동차 이름 : ' + Kimcar.name;
var colname=document.getElementById("data2");
colname.textContent='자동차 색상 : ' + Kimcar.color;
var cspeed = document.getElementById("data3");
cspeed.textContent='자동차 속도 : ' + Kimcar.speedup();
var cspeed=document.getElementById("data4");
cspeed.textContent='자동차 가격 : ' + Kimcar.price;

</script>
</body>
```

[Hong's Car]

자동차 이름 : Sonata

자동차 색상 : blue

자동차 속도 : 110

[Kim's Car]

자동차 이름 : Jeep

자동차 색상 : undefined

자동차 속도 : 80

자동차 가격 : 3천만 원

1. 배열 객체

● 배열 내 객체 구조

```
var car=[
  { name:'Sonata', color:'blue', speed:100 },
  { name:'Jeep', color:'red', speed:70 },
  { name:'Passt', color:'white', speed:150 }
]
car[0].color;
car[2].speed;
car[0];
```

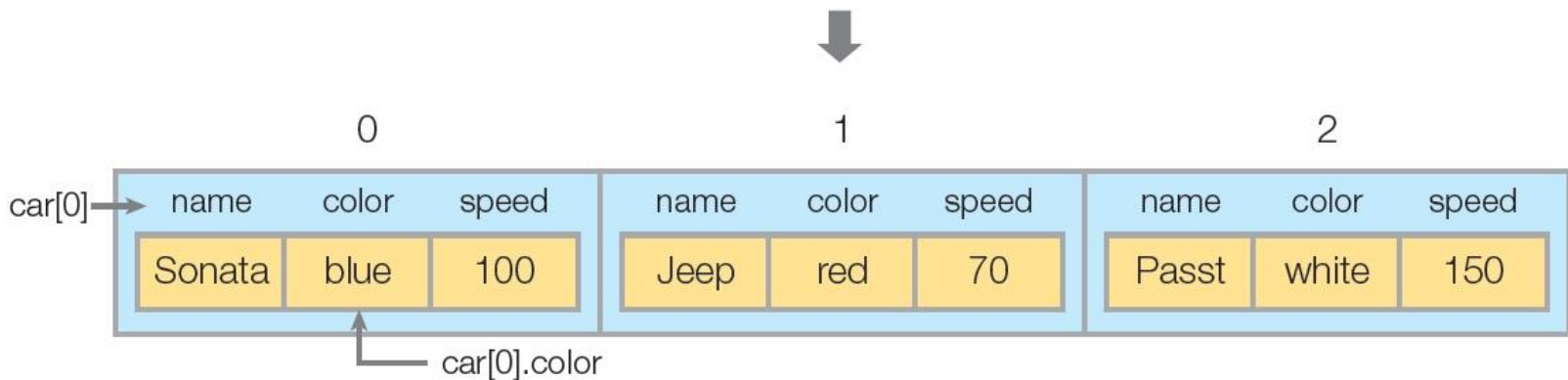


그림 11-4 배열 내 객체 구조

1. 배열 객체

예제 11-9 배열 내 객체 구조 선언하기

ch11/09_obj.html

```

<body>
  <p>[Car1]</p>
  <p id="data10"></p>
  <p id="data20"></p>
  <p id="data30"></p>
  <p>[Car2]</p>
  <p id="data11"></p>
  <p id="data21"></p>
  <p id="data31"></p>
  <p>[Car3]</p>
  <p id="data12"></p>
  <p id="data22"></p>
  <p id="data32"></p>
  <script>
    var car=[
      { name:'Sonata', color:'blue', speed:100 },
      { name:'Jeep', color:'red', speed:70 },
      { name:'Passt', color:'white', speed:150 }
    ]
    for(var i=0; i<3; i++) {
      var cname=document.getElementById("data1"+i);
      cname.textContent='자동차 이름 : ' + car[i].name;
      var colname=document.getElementById("data2"+i);
      colname.textContent='자동차 색상 : ' + car[i].color;
      var cspeed=document.getElementById("data3"+i);
      cspeed.textContent='자동차 속도 : ' + car[i].speed;
    }
  </script>
</body>

```

```

[Car1]
자동차 이름 : Sonata
자동차 색상 : blue
자동차 속도 : 100

[Car2]
자동차 이름 : Jeep
자동차 색상 : red
자동차 속도 : 70

[Car3]
자동차 이름 : Passt
자동차 색상 : white
자동차 속도 : 150

```

1. 배열 객체

● Array 생성자 배열 객체 구조

```
var car=new Array (
    ['Sonata', 'blue', 100],
    ['Jeep', 'red', 70],
    ['Passt', 'white', 150]
);
car[0][1];
car[2][2];
car[0];
```

	[0]	[1]	[2]
car[0]	Sonata	blue	100
car[1]	Jeep	red	70
car[2]	Passt	white	150

car[2][1]

그림 11-5 Array 생성자 배열 객체 구조

1. 배열 객체

예제 11-10 Array 생성자로 배열 객체 구조 선언하기

ch11/10_obj.html

```
<body>
  <p id="car"></p>
  <p>[Car1 속성]</p>
  <p id="car0"></p>
  <p>[Car2 속성]</p>
  <p id="car1"></p>
  <p>[Car3 속성]</p>
  <p id="car2"></p>
  <script>
    var car=new Array (
      ['Sonata','blue',100],
      ['Jeep','red',70],
      ['Passt','white',150]
    );
    var cname=document.getElementById("car");
    cname.textContent="car[2][1] : "+ car[2][1];
    for(var i=0; i<3; i++) {
      var cname=document.getElementById("car"+i);
      cname.textContent=car[i];
    }
  </script>
</body>
```

car[2][1] : white

[Car1 속성]

Sonata,blue,100

[Car2 속성]

Jeep,red,70

[Car3 속성]

Passt,white,150

2. Date 객체

- Date 생성자의 작성 형식

```
var d=new Date();  
var d=new Date(milliseconds);  
var d=new Date(dateString);  
var d=new Date(year, month, day, hours, minutes, seconds, milliseconds);
```

2. Date 객체

예제 11-11 Date 객체 활용하기

ch11/11_obj.html

```

<body>
  <p id="d1"></p>
  <p id="d2"></p>
  <p id="d3"></p>
  <p id="d4"></p>
  <script>
    // 기본 생성자
    document.getElementById("d1").innerHTML = new Date();
    // 1970.01.01 이후의 밀리초 계산
    document.getElementById("d2").innerHTML = new Date(1491803527400);
    // 문자열 날짜
    document.getElementById("d3").innerHTML=new Date("October 15, 2018 06:18:07");
    // 주의사항 : 월(month)은 0부터 시작
    // 날짜 지정
    document.getElementById("d4").innerHTML=new Date(2018, 11, 25, 18, 30, 29);
  </script>
</body>

```

```

Thu Jun 22 2017 17:52:12 GMT+0900 (대한민국 표준시)
Mon Apr 10 2017 14:52:07 GMT+0900 (대한민국 표준시)
Mon Oct 15 2018 06:18:07 GMT+0900 (대한민국 표준시)
Tue Dec 25 2018 18:30:29 GMT+0900 (대한민국 표준시)

```

2. Date 객체

표 11-4 Date 객체 메소드의 종류

구분	메소드	속성 정보
반환 메소드	getDate()	1~31 날짜 반환
	getDay()	0~6 요일 반환(0 : 일요일, 1 : 월요일 ...)
	getFullYear()	연도 반환
	getHours()	0~23 시간 반환
	getMillisecond()	0~999 밀리초 반환
	getMinutes()	0~59 초 반환
	getMonth()	0~11 월 반환
	getSecond()	0~59 초 반환
설정 메소드	setDate()	1~31 날짜 설정
	setDay()	0~6 요일 설정(0 : 일요일, 1 : 월요일 ...)
	setFullYear()	연도 설정
	setHours()	0~23 시간 설정(시간, 분, 초, 밀리초)
	setMillisecond()	0~999 밀리초 설정
	setMinutes()	0~59 초 설정
	setMonth()	0~11 월 설정
	setSecond()	0~59 초 설정

2. Date 객체

예제 11-12 Date 객체의 메소드 활용하기 1

ch11/12_obj.html

```
<body>
  <p id="d1"></p>
  <p id="d2"></p>
  <p id="d3"></p>
  <p id="d4"></p>
  <script>
    var today=new Date();
    document.getElementById("d1").innerHTML=today.getFullYear() + "년";
    document.getElementById("d2").innerHTML=today.getMonth()+1 + "월 " + today.getDate() + "일";
    document.getElementById("d3").innerHTML=today.getHours() + "시 " + today.getMinutes() + "분 " +
    today.getSeconds() + "초";
    document.getElementById("d4").innerHTML="1970년 1월 1일 이후 현재까지 몇 초가 지났나요?</p>" +
    today.getTime() + "ms가 지났습니다.";
  </script>
</body>
```

2017년

6월 22일

17시 52분 51초

1970년 1월 1일 이후 현재까지 몇 초가 지났나요?

1498121571861ms가 지났습니다.

2. Date 객체

예제 11-13 Date 객체의 메소드 활용하기 2

ch11/13_obj.html

```

<body>
  <p id="d1"></p>
  <p id="d2"></p>
  <p id="d3"></p>
  <p id="d4"></p>
  <p id="d5"></p>
  <p id="d6"></p>
  <p id="d7"></p>
  <p id="d8"></p>
  <p id="d9"></p>
  <script>
    var today=new Date();
    document.getElementById( "d1" ).innerHTML=today.toString();
    document.getElementById( "d2" ).innerHTML=today.toISOString();
    document.getElementById( "d3" ).innerHTML=today.toJSON();
    document.getElementById( "d4" ).innerHTML=today.toLocaleDateString();
    document.getElementById( "d5" ).innerHTML=today.toLocaleTimeString();
    document.getElementById( "d6" ).innerHTML=today.toLocaleString();
    document.getElementById( "d7" ).innerHTML=today.toString();
    document.getElementById( "d8" ).innerHTML=today.getTimeString();
    document.getElementById( "d9" ).innerHTML=today.toUTCString();
  </script>
</body>

```

Thu Jun 22 2017

2017-06-22T08:53:39.460Z

2017-06-22T08:53:39.460Z

2017. 6. 22.

오후 5:53:39

2017. 6. 22. 오후 5:53:39

Thu Jun 22 2017 17:53:39 GMT+0900 (대한민국 표준시)

17:53:39 GMT+0900 (대한민국 표준시)

Thu, 22 Jun 2017 08:53:39 GMT

2. Date 객체

예제 11-14 디지털 시계 만들기

ch11/14_obj.html

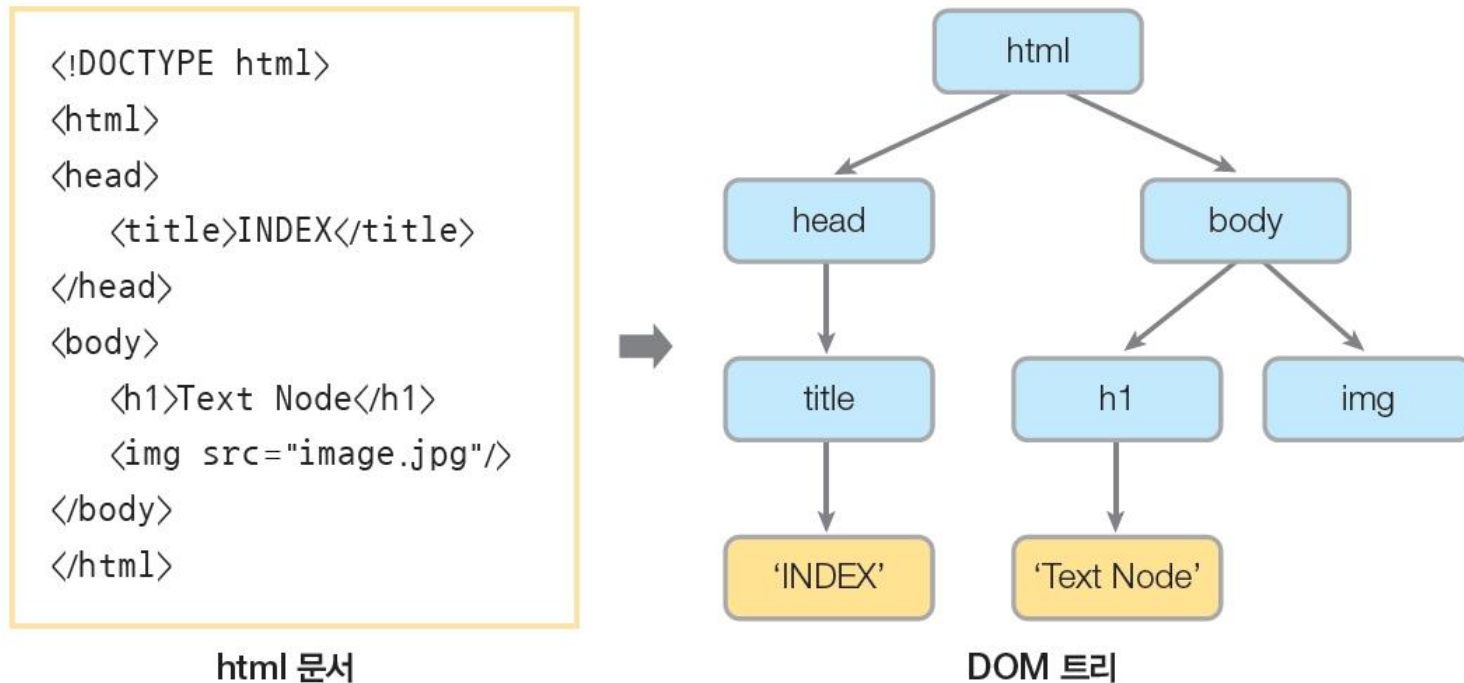
```
<body>
  <div id="digClock"></div>
  <script>
    function digClock() {
      var today=new Date();
      var day=today.getMonth()+1 + "월 " + today.getDate() + "일 ";
      var time=today.getHours() + "시 " + today.getMinutes() + "분 " + today.getSeconds() + "초";
      document.getElementById("digClock").innerHTML=day+time;
      setTimeout('digClock()', 1000);
    }
    digClock();
  </script>
</body>
```

6월 22일 17시 54분 13초

0. 개요

● 문서 객체 모델(DOM)

- 웹 문서를 메모리로 읽어 들여 트리 구조로 변환
- 문서의 각종 요소는 요소 노드로, 텍스트는 텍스트 노드로 변환
- 변환된 트리 구조를 이용하면 자바스크립트로 웹 문서를 조작할 수 있음



1. 문서 객체 생성

표 11-5 문서 객체 생성과 관련된 메소드

메소드	설명
createElement()	요소 노드를 생성한다.
createTextNode()	텍스트 노드를 생성한다.
appendChild()	요소 노드를 body 객체에 추가한다.

예제 11-15 문서 객체 생성하기

ch11/15_dom.html

```

<head>
  <script>
    function add() {
      var header=document.createElement('h3');           // 요소 노드 생성
      var textNode=document.createTextNode('내 이름은 홍길동입니다. '); // 텍스트 노드 생성
      header.appendChild(textNode);                       // 요소 노드와 텍스트 노드 연결
      document.body.appendChild(header);                  // 생성한 문서 객체를 body에 추가
    };
  </script>
</head>
<body>
  <a href="#" onclick="add()">당신의 이름은 무엇입니까?</a>
</body>

```

당신의 이름은 무엇입니까?



당신의 이름은 무엇입니까?

내 이름은 홍길동입니다.

2. 문서 객체 속성값 설정

● 문서 객체의 속성값을 설정하는 방법

- 객체 변수 사용: 객체변수.속성='값';
- 속성 메소드 사용: setAttribute(속성, 값);

예제 11-16 객체 변수를 사용해 속성값 설정하기

ch11/16_dom.html

```
<head>
  <script>
    function add() {
      var img=document.createElement('img');
      img.src='html5.jpg';
      img.width=100;
      img.height=122;
      document.body.appendChild( img );
    };
  </script>
</head>
<body>
  <a href="#" onclick="add()">[이미지 추가]</a>
</body>
```

[이미지 추가]

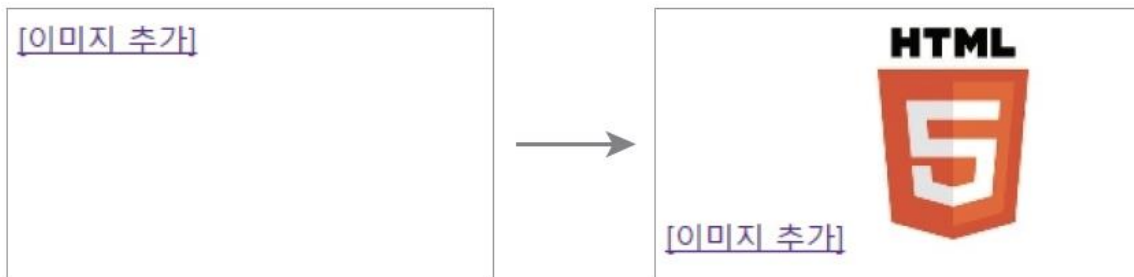


2. 문서 객체 속성값 설정

예제 11-17 setAttribute() 메소드를 사용해 속성값 설정하기

ch11/17_dom.html

```
<head>
  <script>
    function add() {
      var img=document.createElement('img');
      img.setAttribute('src', 'html5.jpg');
      img.setAttribute('width', 100);
      img.setAttribute('height', 122);
      document.body.appendChild( img);
    };
  </script>
</head>
<body>
  <a href="#" onclick="add()">[이미지 추가]</a>
</body>
```



3. 문서 객체 스타일 변경

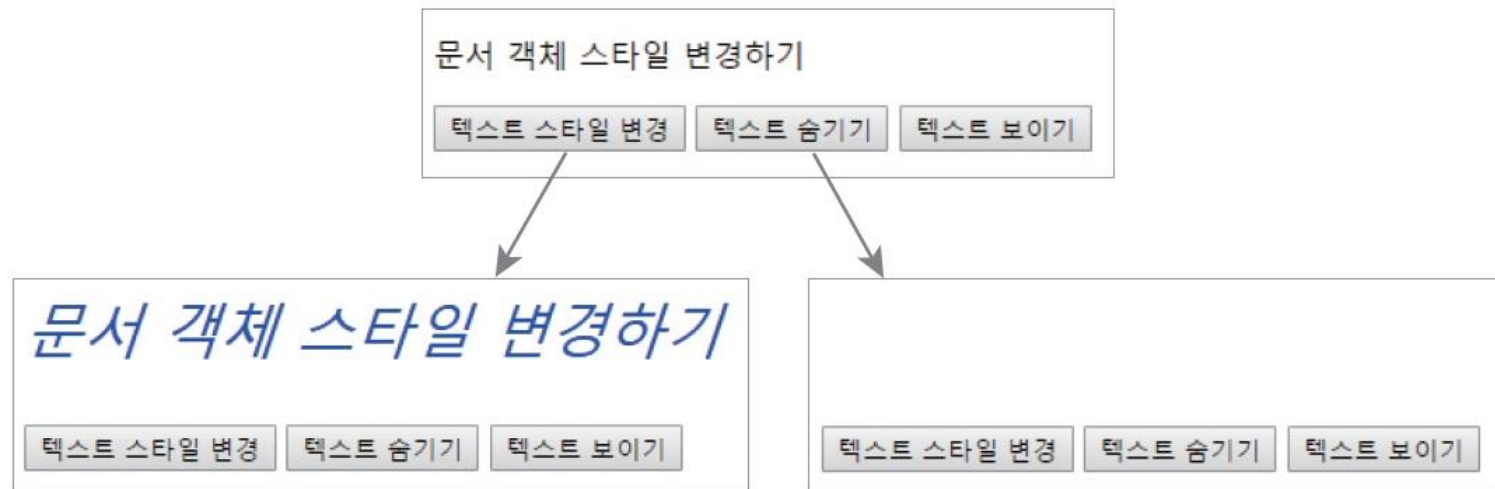
```
document.getElementById(id).style.속성명="속성값";
```

예제 11-18 문서 객체 스타일 변경하기

ch11/18_dom.html

```
<head>
  <script>
    function textstyle() {
      document.getElementById("msg").style.color="blue";
      document.getElementById("msg").style.fontSize="30px";
      document.getElementById("msg").style.fontStyle="italic";
    };
    function texthidden() {
      document.getElementById("msg").style.visibility="hidden";
    };
    function textvisible(){
      document.getElementById("msg").style.visibility="visible";
    };
  </script>
</head>
<body>
  <p id="msg">문서 객체 스타일 변경하기</p>
  <input type="button" onclick="textstyle()" value="텍스트 스타일 변경">
  <input type="button" onclick="texthidden()" value="텍스트 숨기기">
  <input type="button" onclick="textvisible()" value="텍스트 보이기">
</body>
```


3. 문서 객체 스타일 변경





Thank You
