

□ 개념 확인

(1) 괄호 안을 채워 넣으시오

- ① 자바 스크립트 객체는 키와 값으로 구성된 (프로퍼티)들의 집합이다
- ② 자바 스크립트 객체의 프로퍼티 값이 함수일 경우 일반 함수와 구분하기 위해 (메소드)라고 부른다
- ③ 자바 스크립트 객체의 프로퍼티 키는 빈 문자열을 포함하는 모든 (문자열)또는 심볼값을 사용한다
- ④ 프로퍼티 또는 메소드명 앞에 작성하는 (this)는 생성자 함수가 생성할 인스턴스를 의미한다
- ⑤ 생성자 함수를 사용한 객체 생성시 (new) 키워드를 사용한다
- ⑥ 프로퍼티 값을 읽기 위해 대괄호 표기법을 사용할 경우 대괄호 내에 들어가는 프로퍼티 키는 반드시 (문자열) 이어야 한다
- ⑦ 생성자 함수 프로토타입을 사용할 경우 내부에는 (프로퍼티)만 존재한다.
- ⑧ 클래스에서 인스턴스 프로퍼티는 반드시 (constructor)에 정의되어야 한다
- ⑨ 객체 내에 특정 프로퍼티 존재 여부를 확인하려면 (in)연산자를 사용한다
- ⑩ (클래스)로 객체를 생성할 경우 반드시 new 연산자가 있어야 한다

(2) 리터럴 표기법으로 book 객체를 생성하는 문장을 선택하시오

- ① `let book={title:'js', price:30000}`
- ② `let book={title='js', price=3000}`
- ③ `let book={title='js'; price=3000}`
- ④ `let book=[title:'js', price:30000]`

(3) 2번에서 생성된 book 객체에 접근하는 방법을 모두 선택하시오

- ① `book[title]`
- ② `book.title`
- ③ `book->title`
- ④ `book['title']`

(4) 생성자 함수를 사용하여 객체를 정의하는 문장을 선택하시오

- ①

```
let Book = function(title, price){
  this.title=title;  this.price=price;
}
```
- ②

```
function Book(title, price){
  this.title=title;  this.price=price;
}
```
- ③

```
let Book = (title, price) => {
  this.title=title;    this.price=price;
}
```
- ④

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

```
}
Book.prototype.total=title;
```

- (5) 4번의 생성자 함수를 사용하여 객체를 생성하는 문장을 제시하시오. 단, 매개값은 임의로 정할 것

==풀이==

```
let book = new Book("누가 내머리에 동쌌어?",12000)
```

- (6) 생성자 함수와 클래스로 객체를 생성하는 경우 차이점은 무엇인가?

==풀이==

생성자 함수는 new키워드를 사용해야 객체를 생성하고 아닐경우 함수를 호출하지만, 클래스로 객체를 만들 때는 new키워드가 없을 경우 동작이 하지않는다.

클래스로 객체를 만들 때는 prototype을 사용하여 메소드를 선언할 수 있다.

- (7) 질문에 답하시오

- ① Object 생성자 함수를 사용하여 빈 객체를 생성하는 문장을 제시하시오. 단 객체명은 obj1

==풀이==

```
let obj1 = Object.create(null);
```

- ② 1에서 생성된 객체에 다음과 같은 프로퍼티를 추가하고 임의의 값으로 초기화 한다.

```
time(자료타입 number), message(자료타입 string)
```

==풀이==

```
obj1.time = 100;
```

```
obj1.message = "하하";
```

- ③ console.log(age in obj1); 실행 결과를 제시하시오.

==풀이==

```
✖ Uncaught ReferenceError: age is not defined
at task.html:14:21
```

Age가 정의되지 않아서 출력이 되지 않는다.

- (8) 객체 생성과 메소드 호출을 참고하여 Book class를 작성하시오

```
const book = new Book('흑산', '김훈');
```

```
book.bwrite(); //객체 프로퍼티 값을 웹브라우저로 출력
```

==풀이==

```
class Book {
  constructor(bookName, author) {
    this.bookName = bookName;
    this.author = author;
  }
  bwrite() {
    document.write(`책 이름 : ` + this.bookName + `

```

(9) 8에서 생성된 객체의 모든 프로퍼티를 순회하면서 출력하는 문장을 작성하시오. 힌트)for~in

==풀이==

```
for(a in book){
    document.write(`<br>` + a + " : " + book[a]);
}
```

□ 개념 활용 응용 프로그래밍

(1) 다음과 같은 속성과 메소드로 구성되는 객체를 제시된 방법으로 생성하고 결과를 확인하세요

- 속성 : 가수 이름, 곡명, 재생시간
- 메소드 : play(cnt) – cnt 횟수만큼 반복 재생
- 객체 생성 방법
 - 객체 리터럴

```
가수 : 이소라, 제목: 바람이 분다, 재생시간 : 3.5 => 1 번째 재생
가수 : 이소라, 제목: 바람이 분다, 재생시간 : 7 => 2 번째 재생
가수 : 이소라, 제목: 바람이 분다, 재생시간 : 10.5 => 3 번째 재생
가수 : 이소라, 제목: 바람이 분다, 재생시간 : 14 => 4 번째 재생
가수 : 이소라, 제목: 바람이 분다, 재생시간 : 17.5 => 5 번째 재생
```

[소스]

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <script>
    let disk = {
      name: "이소라",
      song: "바람이 분다",
      time: 3.5,

      play(cnt) {
        for (let i = 0; i < cnt; i++) {
          document.write(`가수 : ` + disk.name + `,제목 : ` + disk.song + `,
재생시간 : ` + (disk.time*i) + `=> ` + (i+1) + `번째 재생<br>`);
        }
      }
    }
    disk.play(5);
  </script>
</head>

<body>

</body>
```

</html>

[실행 결과]

가수 : 이소라,제목 : 바람이 분다, 재생시간 : 0=> 1번째 재생
가수 : 이소라,제목 : 바람이 분다, 재생시간 : 3.5=> 2번째 재생
가수 : 이소라,제목 : 바람이 분다, 재생시간 : 7=> 3번째 재생
가수 : 이소라,제목 : 바람이 분다, 재생시간 : 10.5=> 4번째 재생
가수 : 이소라,제목 : 바람이 분다, 재생시간 : 14=> 5번째 재생

(2) 다음과 같은 속성과 메소드로 구성되는 객체를 생성하는 프로그램을 생성자 함수 프로토타입을 사용하여 구현한 후 제시된 결과처럼 동작할 수 있도록 프로그램을 작성하시오

- 속성 : 차량번호, 주행거리
- 메소드 : 주행거리를 dist 만큼 증가시키는 addMileage(dist) 메소드, 반환값 없음
차량번호와 주행거리를 문자열로 반환하는 toString()

127.0.0.1:5501 내용:
차량 번호와 주행거리를 입력하세요.
더 이상 없으면 '완료'를 입력하세요

50서1234 150

확인 취소

결과를 출력합니다

차량번호 : 50서1234 주행거리 : 150
차량번호 : 45머1345 주행거리 : 2000

힌트1) 데이터 입력은 prompt()함수를 사용하고 차량번호와 주행거리는 공백으로 구분한다

힌트2) 입력된 데이터는 split() 함수를 사용하여 구분한 후 객체 초기화에 사용한다

힌트3) 초기화된 객체는 Array에 저장한다.

[소스]

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <script>
    function car(carNum, distance) {
      this.carNum = carNum;
      this.distance = distance;
    }
    car.prototype.addMileage = function (dist) {
      this.distance += dist;
    }
    car.prototype.toString = function () {
      return `차량번호 : ` + this.carNum + ` 주행거리 : ` + this.distance;
    }
    let arr = new Array();

    while (true) {
```

```

        let po = prompt("차량번호와 주행거리를 입력하세요.\n 더이상 없으면 '완료'를
입력하세요");
        if (po == "완료") {
            break;
        }
        let a = po.split(" ");
        let c = new car(a[0],a[1]);
        arr.push(c);
    }
    for(let i = 0 ; i < arr.length ; i++){
        document.write(arr[i].toString() + `<br>`);
    }
</script>
</head>

<body>

</body>

</html>

```

[실행 결과]

차량번호 : 50서 1234 주행거리 : 900
 차량번호 : 12구 7451 주행거리 : 12500

(3) 다음과 같은 속성과 메소드로 구성되는 클래스 Account를 만들고 제시된 결과처럼 실행되는 프로그램을 작성하세요.

- 속성 : 예금주, 잔액
- 메소드
 - 매개변수로 받은 값 만큼 잔액을 증가하는 deposit(매개변수) 메소드, 반환값 없음
 - 매개변수로 받은 값 만큼 잔액을 감소하는 withdraw(매개변수) 메소드, 반환값 없으며 잔액이 적으면

“잔액부족” 출력

- 예금주와 잔액을 출력하는 display() 메소드, 매개변수 없음

```

현재 상태 입니다
예금주 : 스크립트
현재 잔액 : 50000

50000 예금 후 상태 입니다
예금주 : 스크립트
현재 잔액 : 100000

1000000을 인출하려고 합니다
잔액 부족 : 900000
    
```

[소스]

```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">

```

```

<title>Document</title>
<script>
    class Account{
        constructor(name,rest){
            this.name = name;
            this.rest = rest;
            document.write(`현재 상태 입니다<br>`);
        }
        deposit(money){
            document.write(money + ` 예금 후 상태 입니다<br>`);
            this.rest += money;
        }
        withdraw(money){
            let a;
            if(this.rest < money){
                a = "잔액 부족 : " + (money -this.rest);
            }
            else{
                this.rest -= money;
                a = this.display();
            }
            document.write(money + `을 인출하려고 합니다<br>` + a);
        }
        display(){
            document.write(`예금주 : ` + this.name + `<br>현재 잔액 : ` + this.rest +
`<br><br>`);
        }
    }
    const ac = new Account("스크립트" , 50000);
    ac.display();
    ac.deposit(50000);
    ac.display();
    ac.withdraw(1000000);
</script>
</head>

<body>

</body>

</html>

```

[실행 결과]

현재 잔액 : 50000

현재 잔액 : 100000

잔액 부족 : 900000

- | | | |
|-------------|---------------------|--------------|
| 백신종류 : 화미자, | 연락처 : 010-2312-8723 | 접종현황 : 미 접종 |
| 백신종류 : 화미자, | 연락처 : 010-2312-8723 | 접종현황 : 추가 1회 |
| 연락처 변경 후 출력 | | |
| 백신종류 : 화미자, | 연락처 : 010-6543-7968 | 접종현황 : 추가 1회 |

속성	값
백신	모더나, 화이자
접종 횟수	0
연락처	010-2193-5234
동작	내용
isFinished()	접종 횟수가 2이면 '접종 완료', 1이면 '추가 1회', 0이면 '미 접종' 반환
addShot()	접종 회수를 +1 증가, 만약 접종 회수가 2이면 증가 없음
changeTel(value)	연락처를 value값으로 변경

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
    <script>
        class Vaccine{
            constructor(type,count,phone){
                this.type = type;
                this.count = count;
                this.phone = phone;
            }
        }
    </script>

```

```

    isFinished(){
        if(this.count == 0){
            return "미 접종";
        }
        else if(this.count == 1){
            return "추가 1회";
        }
        else if(this.count == 2){
            return "접종 완료";
        }
    }
    addShot(){
        if(this.count < 2){
            this.count++;
        }
    }
    changeTel(value){
        console.log("연락처 변경 후 출력");
        this.phone = value;
    }
}
let vc = new Vaccine("화이자",0,"010-2193-5234");
console.log("백신 종류 : " + vc.type + ", 연락처 : " + vc.phone + " 접종현황 : "
+ vc.isFinished() + "\n");
vc.addShot();
console.log("백신 종류 : " + vc.type + ", 연락처 : " + vc.phone + " 접종현황 : "
+ vc.isFinished() + "\n");
vc.changeTel("010-6543-7968");
console.log("백신 종류 : " + vc.type + ", 연락처 : " + vc.phone + " 접종현황 : "
+ vc.isFinished() + "\n");
</script>
</head>

<body>

</body>

</html>

```

[실행 결과]

백신 종류 : 화이자, 연락처 : 010-2193-5234	접종현황 : 미 접종	task.html:38
백신 종류 : 화이자, 연락처 : 010-2193-5234	접종현황 : 추가 1회	task.html:40
연락처 변경 후 출력		task.html:33
백신 종류 : 화이자, 연락처 : 010-6543-7968	접종현황 : 추가 1회	task.html:42

> |

(5) 2학년 조카의 구구단 학습 도우미 프로그램을 제시된 결과처럼 실행되도록 프로그램하세요.

- 1~9사이에 생성된 난수를 입력창에 제시된 결과처럼 출력하고, 답을 입력 받는다(10번 반복)

127.0.0.1:5500 내용:

1) 8*8 = ?

16

확인 취소

- 맞춘 회수에 10을 곱하여 점수를 계산한다.
- 계산된 점수가 90이상이면 '친구와 놀아도 됩니다', 80 이상이면 '한번 더 연습하세요', 70 이상이면 '두번 더 연습하세요', 70미만이면 '친구와 놀 수 없습니다'를 알림창으로 출력

127.0.0.1:5500 내용:
점수 : 60 -> 친구와 놀 수 없습니다

확인

- Gugudan 클래스를 정의하여 사용하도록 한다.

[소스]

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <script>
    class Gugudan{
      constructor(){
        this.grade = 0;
      }
      gugu(){
        for(let i = 0 ; i < 10 ; i++){
          const a = Math.floor(Math.random()*9) + 1;
          const b = Math.floor(Math.random()*9) + 1;
          let po = prompt((i+1) + " ] " + a + "*" + b + " = ?");
          if(po == a*b){
            this.grade += 10;
          }
        }
        if(this.grade >= 90){
          alert("점수 : " + this.grade + " -> 친구와 놀아도 됩니다");
        }
        else if(this.grade >= 80){
          alert("점수 : " + this.grade + " -> 한번 더 연습하세요");
        }
        else if(this.grade >= 70){
          alert("점수 : " + this.grade + " -> 두번 더 연습하세요");
        }
        else{
          alert("점수 : " + this.grade + " -> 친구와 놀 수 없습니다");
        }
      }
    }
    let gu = new Gugudan();
    gu.gugu();
  </script>
</head>

<body>
```

```
</body>
```

```
</html>
```

[실행 결과]

36

127.0.0.1:5500 내용:

2] 9*9 = ?

127.0.0.1:5500 내용:

점수 : 100 -> 친구와 놀아도 됩니다