Seoul woman up

스타트업 개발자와 함께 공부하는

Node.js

01. 자바스크립트

강의 내용은 강사가 별도로 명시하지 않는 한 비공개로 간주합니다. 녹음이나 사진 촬영를 허락하지 않으며 콘텐츠를 블로그, SNS 등에 게시하거나 공개적으로 공유하지 마세요.

콘텐츠 공유 가능 여부에 대해 궁금한 점이 있는 경우 강사에게 문의하시기 바랍니다.

목차

Seoul woman up

자바스크립트 기본

자바스크립트 기본

❑ 출력

```
1 console.log("Hello world");
2
3 console.log('hello world');
4
5 console.log(`hello ${"world"}`);
6
7
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_01.js]

자바스크립트 기본

❑ 기본 자료형

```
1  //숫자 사칙연산
2  console.log(23);
3  console.log(23 + 45);
4  console.log(23 * 45);
5  console.log(2 + 3 * 5);
6  console.log(10 % 5);
7  console.log(10 / 5);
8  //문자자료형 및 이스케이프
9  console.log('안녕하세요')
10 console.log('this is "string"')
11 console.log("this is \"string\" ");
12 console.log("this is \n  string ");
13 console.log("this is \t string");
14 console.log("this is \\ string");
15 //문자열 더하기
16 console.log("hello" + " " + "javascript")
17 console.log("hello"[1]);
18 //템플릿 문자열
19 console.log(`52 + 45 = ${52 + 45}`);
20 //비교 연산자
21 console.log( 52 > 52 );
22 console.log( 45 == 52 );
23 console.log( 45 != 52 );
24 console.log( 52 >= 52 );
25 //논리 연산자
26 console.log( 52 >= 52 && 52 > 52 );
27 console.log( 52 >= 52 || 52 > 52 );
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_02.js]

자바스크립트 기본

❑ 변수

```
let pi;
console.loog(pi);
pi = 3.141592
console.loog(pi);

let pi2 = 3.141592
console.log(pi2);

let radius = 12;

console.log(`넓이 : ${pi * radius * radius}`);
console.log(`둘레 : ${pi * 2 * radius}`);

```

❑ [C]-[nodejs]-[project]-[01]-[ch01_03.js]

자바스크립트 기본

❑ 복합 대입 연산자

```
1  radius += 2
2  console.log(radius);
3
4  let output = "hello";
5  output += "world";
6  console.log(output);
7
8  let num = 0;
9  num++;
10 console.log(`num++ : ${num}`);
11 num--;
12 console.log(`num-- : ${num}`);
13
14
15 console.log(`num++ : ${num++}`);
16 console.log(`num-- : ${num--}`);
17
18 console.log(`++num : ${++num}`);
19 console.log(`--num : ${--num}`);
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_04.js]

자바스크립트 기본

❑ 자료형 검사

```
1  console.log( typeof(10) );
2
3  console.log( typeof("hello") );
4
5  console.log( typeof(true) );
6
7  console.log( typeof(function() {} ) );
8
9  console.log( typeof( {}  ) );
10
11 let beta
12 console.log(typeof(beta));
13
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_05.js]

자바스크립트 기본

❑ 자료형 변환

```
1  console.log(String(52));
2  console.log(typeof(52));
3  console.log(typeof(String(52)));
4
5  console.log(Number("45"));
6  console.log(Number(true));
7  console.log(Number("hello"));
8
9  console.log(isNaN(Number("hello")));
10
11 console.log(Boolean(0));
12 console.log(Boolean(NaN));
13 console.log(Boolean(1));
14 console.log(Boolean("Hello"));
15
16 console.log(25 + 125);
17 console.log("25" + 125);
18 console.log("25" - 125);
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_06.js]

자바스크립트 기본

❑ 일치 연산자

```
1 console.log(`50 == "50" : ${50 == "50" }`);
2 console.log(`50 === "50" : ${50 === "50" }`);
3
4 console.log(`0 == "" : ${0 == ""}`);
5 console.log(`0 === "" : ${0 === ""}`);
6
```

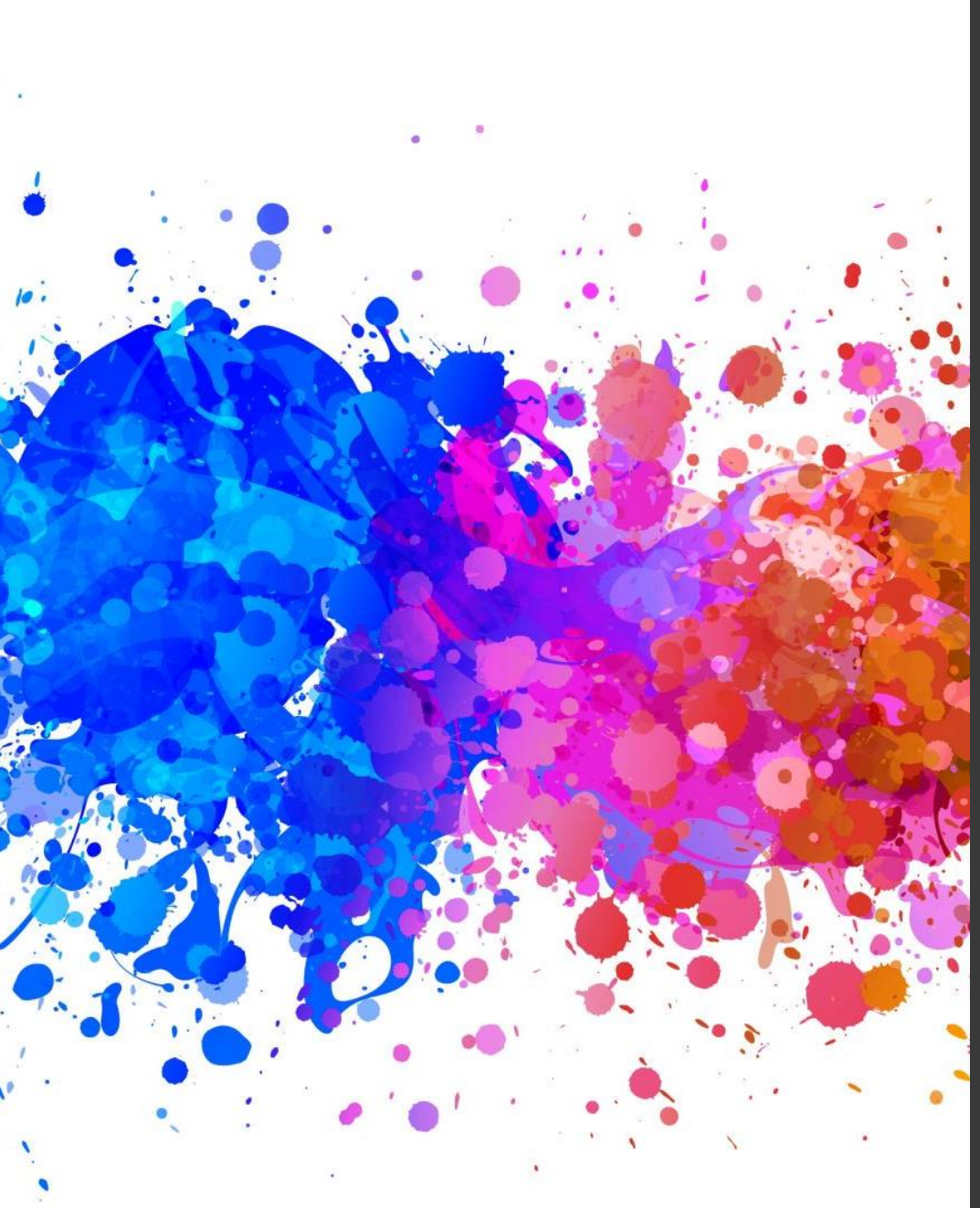
❑ [C]-[nodejs]-[project]-[01]-[ch01_07.js]

자바스크립트 기본

❑ 상수

```
1 const test = "변경 불가";
2 console.log(`before : ${test}`);
3 test = "hello"
4 console.log(`after : ${test}`);
5
6 const abc = "abc"
7 // const def;
8
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_08.js]



Seoul woman up

조건문

조건문

❑ 기본 조건문

```
1 // 기본 조건문
2 let date = new Date();
3
4 if (date.getHours() < 12 ) {
5     console.log(`오전입니다 : ${date.getHours()}`);
6 }else if(date.getHours() >= 12 ){
7     console.log(`오후입니다 : ${date.getHours()}`);
8 }
9
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_09.js]

조건문

❑ 중첩 조건문

```
1  // 중첩 조건문
2  let date = new Date();
3  let hours = date.getHours();
4
5  if(hours < 11) {
6      console.log(`아침 먹을 시간 입니다`);
7  }else {
8      if(hours < 15) {
9          console.log(`점심 먹을 시간 입니다`);
10     }else{
11         console.log(`저녁 먹을 시간 입니다`);
12     }
13 }
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_10.js]

조건문

❑ switch 문

```
let input = 12;

switch(input % 2) {
    case 0:
        console.log(`even`);
        break;
    case 1:
        console.log(`odd`);
        break;
}

```

❑ [C]-[nodejs]-[project]-[01]-[ch01_11.js]

조건문

❑ 삼항 연산자

```
1 let test
2
3 test = typeof(test) != 'undefined' ? test: 'initial';
4 console.log(test);
5
6
7
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_12.js]

Seoul woman up

반복문

반복문

❑ 배열

```
1 // array
2 let arr = [5, 23, 'hello', true, 'world', -9]
3 console.log(arr)
4 console.log(arr[2]);
5
6
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_13.js]

반복문

❑ while

```
1 // while
2 let arr = [5, 23, 'hello', true, 'world', -9]
3 let i = 0;
4 while (i < arr.length) {
5     console.log(` ${i} is ${arr[i]}`);
6     i++;
7 }
8
9
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_14.js]

반복문

❑ for

```
1 // for
2 let arr = [5, 23, 'hello', true, 'world', -9]
3 for(let i=0;i<arr.length;i++){
4     console.log(` ${i} is ${arr[i]}`);
5 }
6
7
8
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_15.js]

반복문

❑ for in

```
1 // for in
2 let arr = [5, 23, 'hello', true, 'world', -9]
3 for(i in arr){
4     console.log(` ${i} is ${arr[i]}`);
5 }
6
7
8
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_16.js]

반복문

❑ for of

```
1 // for of
2 let arr = [5, 23, 'hello', true, 'world', -9]
3 for(e of arr){
4     console.log(` ${e}`);
5 }
6
7
8
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_17.js]

반복문

❑ 중첩 반복문

```
let output = '';

for(let i=0;i<10;i++){
    for(let j=0;j< 10 - i;j++){
        output += ' ';
    }
    for(let j=0;j< i + 1; j++){
        output += '*';
    }
    output += '\n';
}

console.log(output);

```

❑ [C]-[nodejs]-[project]-[01]-[ch01_18.js]

반복문

❑ break

```
1  let arr = [5, 23, 'hello', true, 'world', -9]
2
3  for(i in arr) {
4      if(typeof(arr[i]) == 'string'){
5          break;
6      }
7      console.log(arr[i]);
8  }
9
10
```

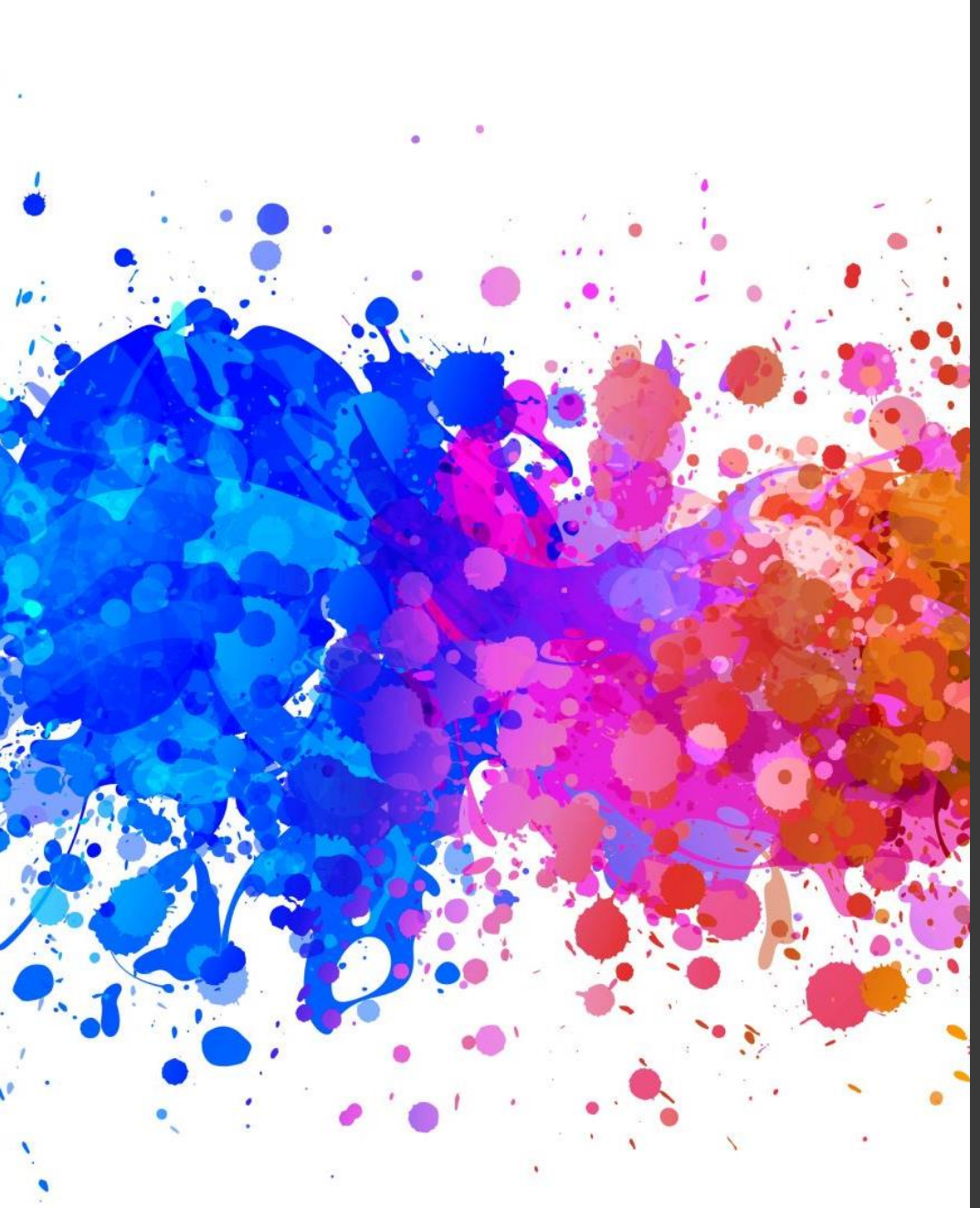
❑ [C]-[nodejs]-[project]-[01]-[ch01_19.js]

반복문

❑ continue

```
1  let arr = [5, 23, 'hello', true, 'world', -9]
2
3  for(i in arr) {
4      if(typeof(arr[i]) == 'string'){
5          console.log(arr[i]);
6          continue;
7      }
8  }
9
10
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_20.js]



Seoul woman up

함수

함수

❑ 익명함수, 선언함수, 화살표함수

```
1  // 익명함수
2  let add1 = function(x, y) {
3      return x + y;
4  }
5
6  // 선언 함수
7  function add2(x, y) {
8      return x + y;
9  }
10
11 // 화살표 함수
12 let add3 = (x, y) => x + y;
13
14 console.log(add1(1, 4));
15 console.log(add2(1, 4));
16 console.log(add3(1, 4));
17
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_21.js]

함수

❑ 콜백함수

```
1  // callback
2  function tentimes(callback) {
3      for(let i=0;i<10;i++){
4          callback();
5      }
6  }
7
8  tentimes(function() {
9      console.log('call function');
10 });
11
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_22.js]

함수

❑ 콜백함수

```
1 // parseInt
2 console.log(`parseInt("52") => ${parseInt("52")}`);
3 console.log(`parseInt("3.14") => ${parseInt("3.14")}`);
4 console.log(`parseInt("1209동") => ${parseInt("1209동")}`);
5 // parseFloat
6 console.log(`parseFloat("51.128") => ${parseFloat("51.128")}`);
7 console.log(`parseFloat("3.14") => ${parseFloat("3.14")}`);
8 console.log(`parseFloat("1209.2동") => ${parseFloat("1209.2동")}`);
9
```

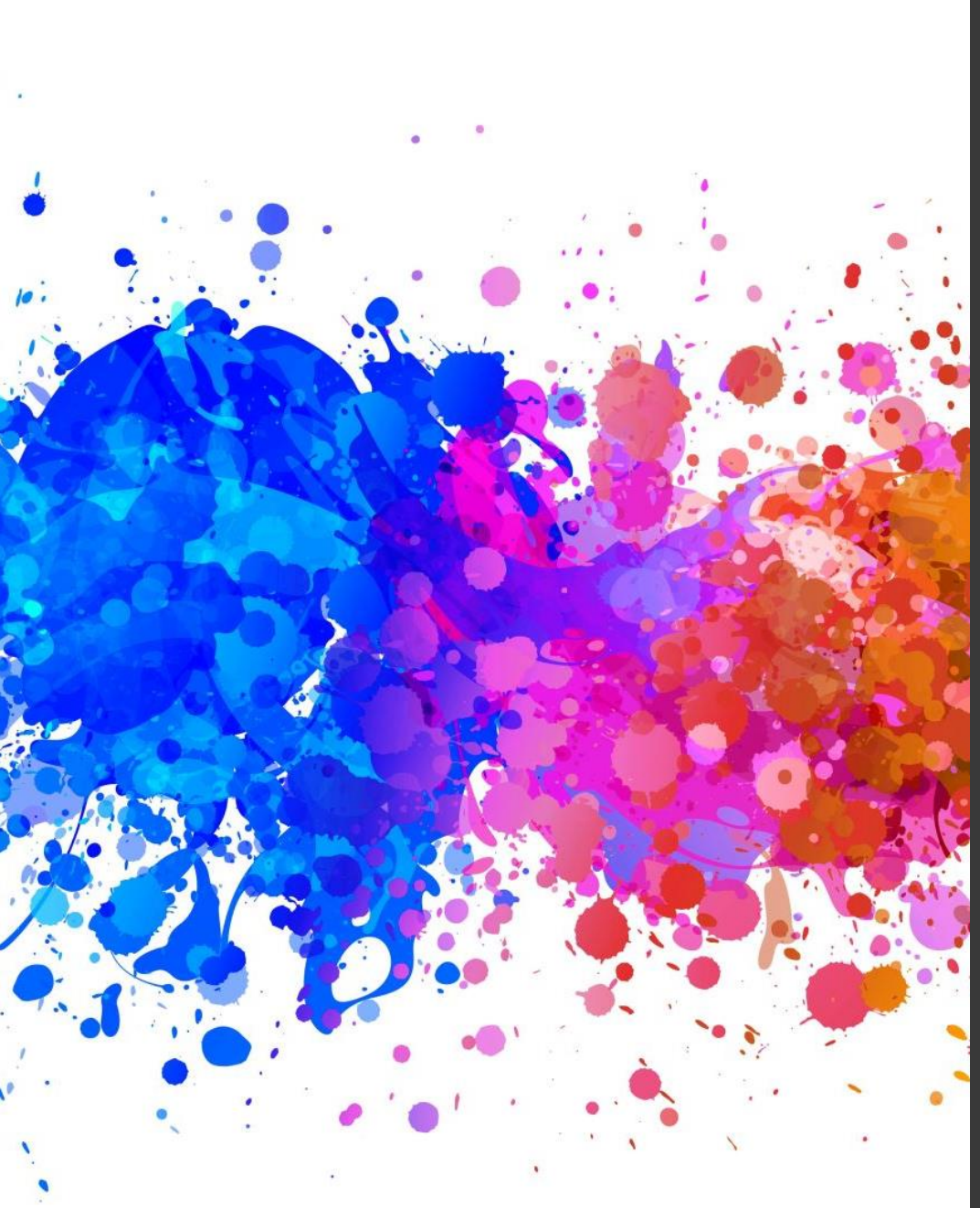
❑ [C]-[nodejs]-[project]-[01]-[ch01_23.js]

함수

❑ 시간 함수

```
1  //setTimeout
2  setTimeout(function() {
3      console.log(`1초 후 호출`);
4  }, 1000);
5
6  //setInterval
7  setInterval(function() {
8      console.log(`1초 마다 호출`);
9  }, 1000);
10
11
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_24.js]



Seoul woman up

객체

함수

❑ 객체

```
1  // 객체
2  let personInfo = {
3      'name': 'lee',
4      'age': 15,
5      'address': 'seoul',
6      'hobby': ['fishing','rc','star']
7  }
8
9  console.log(personInfo['name']);
10 console.log(personInfo['address']);
11 console.log(personInfo['hobby']);
12 console.log(personInfo.age);
13
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_25.js]

함수

❑ 객체와 for문

```
1  // for 문과 객체
2  let personInfo = {
3      name: 'lee',
4      age: 15,
5      address: 'seoul',
6      hobby: ['fishing','rc','star']
7  }
8
9  for(let key in personInfo){
10     console.log(`${key} : ${personInfo[key]}`);
11 }
12
13
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_26.js]

함수

❑ 객체와 메서드

```
1  // 객체와 메서드
2  let personInfo = {
3      name: 'lee',
4      age: 15,
5      address: 'seoul',
6      hobby: ['fishing','rc','star'],
7      addAge: function() {
8          this.age = this.age + 1;
9      }
10 }
11
12 console.log(`before age : ${personInfo.age}`);
13 personInfo.addAge();
14 console.log(`after age : ${personInfo.age}`);
15
16
```

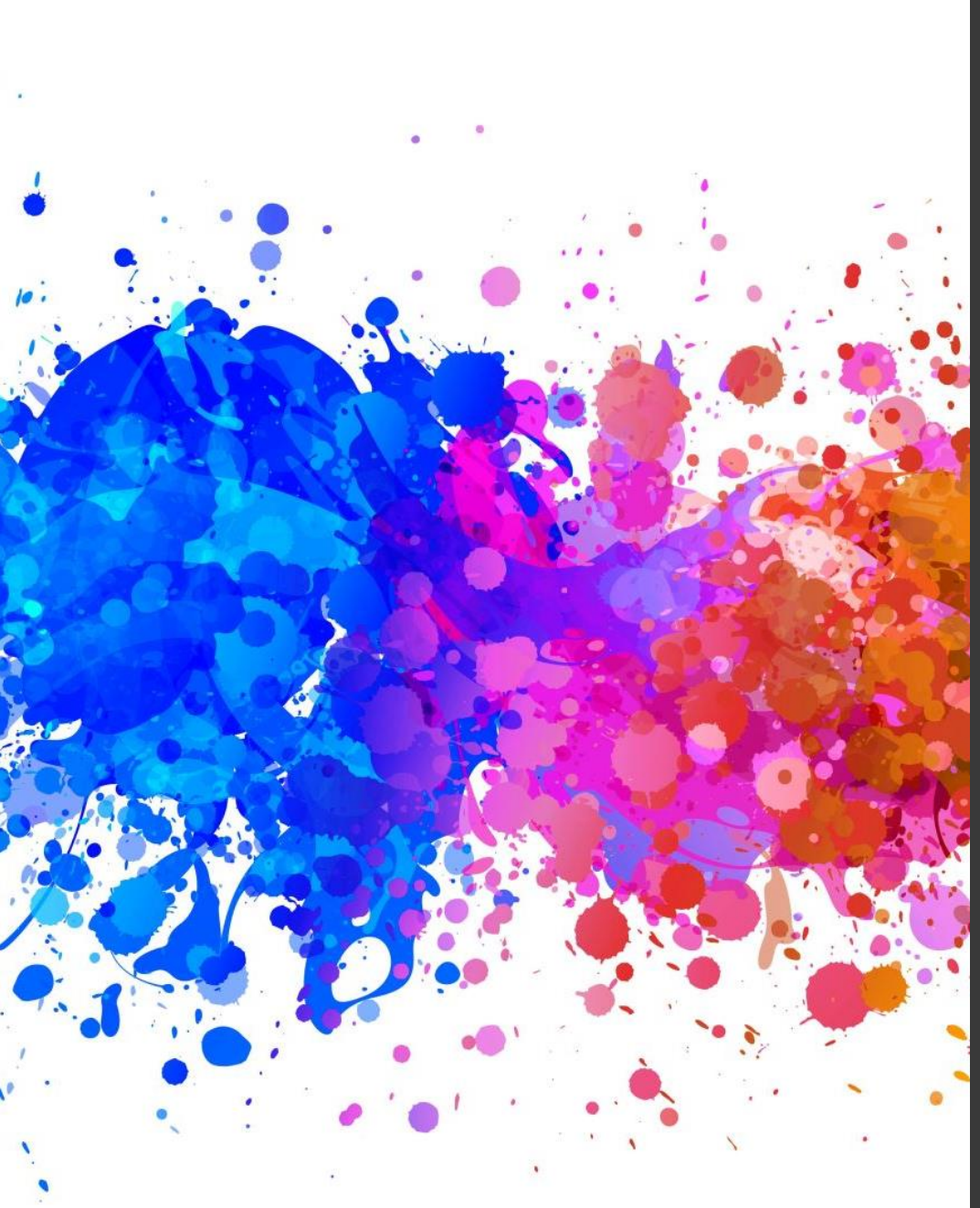
❑ [C]-[nodejs]-[project]-[01]-[ch01_27.js]

함수

❑ 클래스

```
1  // 클래스
2  class PersonInfo {
3      constructor(name, age, address, hobby) {
4          this.name = name;
5          this.age = age;
6          this.address = address;
7          this.hobby = hobby
8      }
9
10     addAge(age) {
11         this.age = this.age + age;
12     }
13
14     getAge() {
15         return this.age;
16     }
17 }
18
19 let p = new PersonInfo('lee', 12, 'seoul', ['rc','star']);
20 console.log(`before age : ${p.getAge()}`);
21 p.addAge(10);
22 console.log(`after age : ${p.getAge()}`);
23
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_28.js]



Seoul woman up

예외처리

함수

❑ 예외처리

```
1 try{
2     const arr = new Array(-1);
3 }catch(e) {
4     console.log(`exception occured : ${e}`);
5 }finally {
6     console.log(`finally called`);
7 }
8
9
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_29.js]

함수

❑ 예외처리

```
1  try{
2      const err = new Error('This is Error');
3      err.name = 'My Error';
4      err.message = 'My Error Message'
5
6      throw err;
7  }catch(e) {
8      console.log(`exception occured =>
9          name:${e.name}, message: ${e.message}`);
10 }finally {
11     console.log(`finally called`);
12 }
13
14
```

❑ [C]-[nodejs]-[project]-[01]-[ch01_30.js]

Seoul woman up

백문이 불여일타