프론트엔드개발 구현시험

## 압축 파일 이름

예) 홍길동-프런트엔드.zip

## exam1.js

|  |
| --- |
| function printType(value) {    // 이 함수를 구현하라  }  printType(123);  printType("hello");  printType(true);  printType([1, 2, 3]); |

출력

|  |
| --- |
| number  string  boolean  object |

파라미터로 전달된 value의 타입이 출력되도록

printType 함수를 구현하라.

## exam2.js

|  |
| --- |
| function printSum(a, b) {    // 이 함수를 구현하라.  }  printSum(1, 2);  printSum(3, 4);  printSum(12, 34); |

출력

|  |
| --- |
| 12  34  1234 |

파라미터로 전달된 숫자를, 문자열처럼 더해서 출력되도록

printSum 함수를 구현하라.

## exam3.js

|  |
| --- |
| function createArray(count) {    // 이 함수를 구현하라  }  console.log(createArray(2));  console.log(createArray(3));  console.log(createArray(4));  console.log(createArray(5)); |

1 이상 5 이하의 정수가 count 개 들어있는 배열을 생성하여 리턴하도록

createArray 함수를 구현하라.

## exam4.js

|  |
| --- |
| function createArray(count) {    // 이 함수를 구현하라  }  console.log(createArray(2));  console.log(createArray(3));  console.log(createArray(4));  console.log(createArray(5)); |

0 이상 5 미만의 실수가 count 개 들어있는 배열을 생성하여 리턴하도록

createArray 함수를 구현하라.

## exam5.js

|  |
| --- |
| function rotate(a) {    // 이 함수를 구현하라  }  let a = [1, 2, 3, 4];  rotate(a);  console.log(a);  rotate(a);  console.log(a); |

출력

|  |
| --- |
| [ 4, 1, 2, 3 ]  [ 3, 4, 1, 2 ] |

rotate 함수를 호출할 때 마다 a 배열이 오른쪽으로 한 칸 회전해야 한다.

배열의 오른쪽 끝 값은 배열의 왼쪽 첫 칸으로 이동하고,

나머지 같들은 한 칸씩 왼쪽으로 이동한다.

## exam6.js

|  |
| --- |
| function insertZero(a, index) {    // 이 함수를 구현하라  }  let a = [1, 2, 3];  insertZero(a, 2);  console.log(a);  insertZero(a, 1);  console.log(a); |

출력

|  |
| --- |
| [ 1, 2, 0, 3 ]  [ 1, 0, 2, 0, 3 ] |

a 배열의 index 위치에 0 값을 삽입하도록

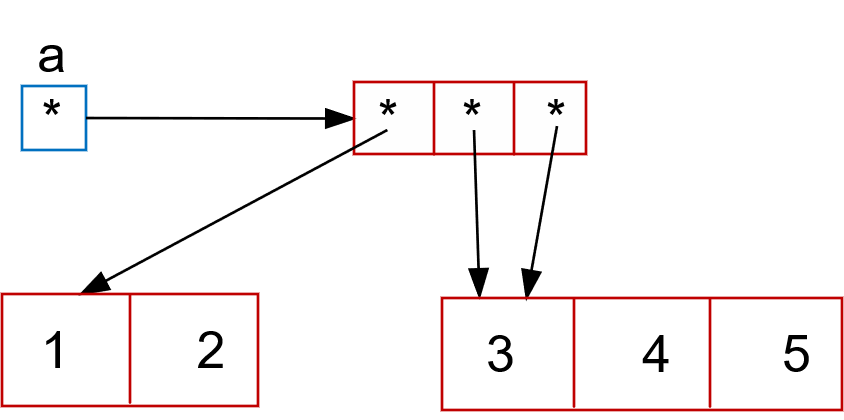
insertZero 함수를 구현하라.

## exam7.js

|  |
| --- |
| function createArray() {    // 이 함수를 구현하라  }  console.log(createArray()); |

출력

|  |
| --- |
| [ [ 1, 2 ], [ 3, 4, 5 ], [ 3, 4, 5 ] ] |



메모리 구조가 위와 같은 배열을 생성하여 리턴하도록

createArray 함수를 구현하라.

## exam8.js

|  |
| --- |
| function countSpace(s) {    // 이 함수를 구현하라  }  console.log(countSpace("aa bb ccc"));  console.log(countSpace("a b cc ddd  e")); |

출력

|  |
| --- |
| 2  5 |

문자열 s에 들어있는 공백 문자의 갯수를 리턴하도록

countSpace 함수를 구현하라.

## exam9.js

|  |
| --- |
| function removeSpace(s) {    // 이 함수를 구현하라  }  console.log(removeSpace("aa bb ccc"));  console.log(removeSpace("a b cc ddd  e")); |

출력

|  |
| --- |
| aabbccc  abccddde |

문자열 s에 들어있는 공백 문자들이 전부 제거된 새 문자열을 리턴하도록

removeSpace 함수를 구현하라.

## exam10.js

|  |
| --- |
| function testCallback(f, a, b) {    // 이 함수를 구현하라  }  testCallback((a, b) => a + b, 3, 4);  testCallback((a, b) => a \* b, 3, 4); |

출력

|  |
| --- |
| 7  12 |

a, b를 파라미터로 전달하며 f 콜백 함수를 호출하고,

f 콜백 함수가 리턴하는 값을 출력하도록

testCallback 함수를 구현하라.

## exam11.js

|  |
| --- |
| function createObj(value) {    // 이 함수를 구현하라  }  console.log(createObj(3));  console.log(createObj("hello")); |

출력

|  |
| --- |
| { value: 3 }  { value: 'hello' } |

파라미터로 전달된 value 값이

value 속성으로 들어있는 객체를 생성하여 리턴하도록

createObj 함수를 구현하라.