

# Assessment-2

07 July 2024 10:01

**Ans:- 1**

**(b)** option is correct

**Ans:- 2**

**(b)** option is correct

**Ans:- 3**

**(b)** option is correct

**Ans:- 4**

**(a)** option is correct

**Ans:- 5**

10000000000

**Ans:- 6**

**A) Output array: ['John', 'Jane', 'Bob']**

We can use the map function to extract the names:

Output:-

```
const names = people.map(person => person.name);
console.log(names); // ['John', 'Jane', 'Bob']
```

**B) Output array:**

```
const peopleWithSalary = people.map(person => ({
  person,
  salary: 50000
}));
console.log(peopleWithSalary);
```

**C) Create an array of objects for people who are above 30 years old, and you only need their names and ages (no id property)**

```
const peopleAbove30 = people
  .filter(person => person.age > 30)
  .map(person => ({ name: person.name, age: person.age }));
```

```
console.log(peopleAbove30);
// [{ "name": "Bob", "age": 40 }]
```

**Ans:- 7**

// Define cb1 to add two numbers

```
const cb1 = (a, b) => a + b;
```

// Define cb2 to give the difference of two numbers

```
const cb2 = (a, b) => a - b;
```

// Define the main function that takes two callback functions and two numbers as parameters

```
function main(cb1, cb2, x, y) {
  const sum = cb1(x, y);
  const difference = cb2(x, y);
  console.log(`Sum: ${sum}`);
  console.log(`Difference: ${difference}`);
  return { sum, difference };
}
```

// Example usage

```
const x = 11;
```

```
const y = 4;
```

```
const result = main(cb1, cb2, x, y);
```

```
console.log(result);
```

**Ans:- 8**

( c ) option is correct

```
var outputArray = users.map(user => `${user.firstName} ${user.lastName}`); console.log(outputArray);  
// Output: ["Susan Steward", "Daniel Longbottom", "Jacob Black"]
```

**Ans:- 10**

( c ) [] :- option is correct

**Ans:- 11**

( b ) option is correct

display: none removes the element from the dom while visibility: hidden just hides the element.

**Ans:- 12**

(A) 4 10 18. option is correct

**Ans:- 13**

(A) [1, 2, 3, 4]. option is correct

Submitted by:- Vikash Verma  
Univ. Roll :- 2215001957

Submitted to :- Shubham Sir

## Assessment-2

07 July 2024 10:01

**Ans:- 1**

**(b)** option is correct

**Ans:- 2**

**(b)** option is correct

**Ans:- 3**

**(b)** option is correct

**Ans:- 4**

**(a)** option is correct

**Ans:- 5**

10000000000

**Ans:- 6**

**A) Output array: ['John', 'Jane', 'Bob']**

We can use the map function to extract the names:

Output:-

```
const names = people.map(person => person.name);
console.log(names); // ['John', 'Jane', 'Bob']
```

**B) Output array:**

```
const peopleWithSalary = people.map(person => ({
  person,
  salary: 50000
}));
console.log(peopleWithSalary);
```

**C) Create an array of objects for people who are above 30 years old, and you only need their names and ages (no id property)**

```
const peopleAbove30 = people
  .filter(person => person.age > 30)
  .map(person => ({ name: person.name, age: person.age }));

console.log(peopleAbove30);
// [{ "name": "Bob", "age": 40 }]
```

**Ans:- 7**

// Define cb1 to add two numbers

```
const cb1 = (a, b) => a + b;
```

// Define cb2 to give the difference of two numbers

```
const cb2 = (a, b) => a - b;
```

// Define the main function that takes two callback functions and two numbers as parameters

```
function main(cb1, cb2, x, y) {
  const sum = cb1(x, y);
  const difference = cb2(x, y);
}
```

```
    console.log(`Sum: ${sum}`);  
    console.log(`Difference: ${difference}`);  
    return { sum, difference };  
}
```

// Example usage

```
const x = 11;
```

```
const y = 4;
```

```
const result = main(cb1, cb2, x, y);
```

```
console.log(result);
```

**Ans:- 8**

**( c )** option is correct

```
var outputArray = users.map(user => `${user.firstName} ${user.lastName}`); console.log(outputArray);
```

```
// Output: ["Susan Steward", "Daniel Longbottom", "Jacob Black"]
```

**Ans:- 10**

**( c ) []** :- option is correct

**Ans:- 11**

**( b )** option is correct

display: none removes the element from the dom while visibility: hidden just hides the element.

**Ans:- 12**

**(A) 4 10 18.** option is correct

**Ans:- 13**

**(A) [1, 2, 3, 4].** option is correct

Submitted by:- Vikash Verma

Shubham Sir

Univ. Roll :- 2215001957

Submitted to : -