

# PASSEXAM 問題集

更に上のクオリティ 更に上のサービス



1年で無料進級することに提供する  
<http://www.passexam.jp>

**Exam** : **JSA-41-01**

**Title** : JSA - Certified Associate  
JavaScript Programmer

**Version** : DEMO

1. Which of the following methods will correctly perform a deep clone of an object in JavaScript?

- A. `original.clone()`
- B. `JSON.parse(JSON.stringify(original))`
- C. `Object.create(original)`
- D. `Object.assign({}, original)`

**Answer: B**

2. Consider the following JavaScript code snippet:

- ```
1. const book = {  
2. title: 'The Great Gatsby',  
3. author: 'F. Scott Fitzgerald',  
4. getSummary: function() {  
5. return `${this.title} by ${this.author}`;  
6. }  
7. };  
8.  
9. const getBookSummary = book.getSummary;  
10. console.log(getBookSummary()); // What will this log to the console?
```

- A. ReferenceError: title is not defined
- B. null
- C. The Great Gatsby by F. Scott Fitzgerald
- D. undefined by undefined

**Answer: D**

3. Consider the following JavaScript code snippet:

- ```
1. const obj = {  
2. prop: 42,  
3. func: function() {  
4. return this.prop;  
5. }  
6. };  
7.  
8. const extractedFunc = obj.func;  
9. console.log(extractedFunc()); // What will this log to the console?
```

- A. 42
- B. TypeError: extractedFunc is not a function
- C. undefined
- D. null

**Answer: C**

4. You have an array of tasks, where each task is an object with properties `id`, `title`, and `completed`. Write a JavaScript function that returns a new array containing only the titles of the tasks that are not completed, sorted alphabetically.

Which of the following functions correctly implements this?

- A. `function getIncompleteTaskTitles(tasks) { return tasks.sort((a, b) => a.title.localeCompare(b.title)) .filter(task => !task.completed) .map(task => task.title);}`
- B. `function getIncompleteTaskTitles(tasks) { return tasks.filter(task => task.completed) .map(task => task.title) .sort((a, b) => a.localeCompare(b));}`
- C. `function getIncompleteTaskTitles(tasks) { return tasks.map(task => task.title) .filter(task => !task.completed) .sort((a, b) => a.localeCompare(b));}`
- D. `function getIncompleteTaskTitles(tasks) { return tasks.filter(task => !task.completed) .sort((a, b) => a.title.localeCompare(b.title)) .map(task => task.title);}`

**Answer: D**

5. You have the following JavaScript code:

- 1. `let config = {`
- 2.  `server: "localhost",`
- 3.  `port: 8080,`
- 4.  `options: {`
- 5.  `secure: true,`
- 6.  `timeout: 5000`
- 7.  `},`
- 8.  `features: {`
- 9.  `feature1: "enabled",`
- 10.  `feature2: "disabled"`
- 11.  `}`
- 12. `};`
- 13.
- 14. `let key = "features";`
- 15. `let subKey = "feature1";`
- 16. `let dynamicKey = "timeout";`

Which two of the following expressions correctly access the value 5000 from the config object?

- A. `config["options"]["timeout"]`
- B. `config[key][subKey]`
- C. `config.options.timeout`
- D. `config[key].options[dynamicKey]`

**Answer: A, C**