**JS Questions**

Table of Contents

[**1.** **What are the possible ways to create objects in JavaScript** 2](#_Toc124953218)

[**2.** **What is a prototype chain** 2](#_Toc124953219)

[**3.** **What is JSON** 2](#_Toc124953220)

[**4.** **compare Object and Map** 2](#_Toc124953221)

[**5.** **compare Array and Set** 2](#_Toc124953222)

[**6.** **arrow functions** 2](#_Toc124953223)

[**7.** **What is the Temporal Dead Zone** 2](#_Toc124953224)

[**8.** **differences between cookie, local storage and session storage** 3](#_Toc124953225)

[**9.** **What is the difference between null and undefined** 3](#_Toc124953226)

[**10.** **What is the difference between window and document** 3](#_Toc124953227)

[**11.** **What is event bubbling** 3](#_Toc124953228)

[**12.** **IIFE(Immediately Invoked Function Expression)** 3](#_Toc124953229)

1. **What are the possible ways to create objects in JavaScript**

* Object literal

var object = {

name: "Sudheer",

age: 34

};

* New Object()
* Object.create(null)

1. **What is a prototype chain**

* Dùng để tạo object mới dựa trên các thuộc tính của một object có sẵn. Giống inheritance (kế thừa) OOP

difference between Call, Apply and Bind

call(arg1, arg2,arg3,...): gọi function với this dc gán cho đối số đầu tiên. Các đối số thứ 2 trở đi là dc truyền vào hàm gọi call().

apply(arg1, arg2): gọi function với this dc gán cho đối số đầu tiên. đối số thứ 2 phải là array và dc truyền vào hàm gọi call().

bind(arg): gọi function với this dc gán cho đối số đầu tiên. Ko có đối số thứ 2.

1. **What is JSON**

Text-based data. Use to transmit data through network if native data is big.

Note: JSON.parse() doesn’t parse value of undefined.

1. **compare Object and Map**

|  |  |  |
| --- | --- | --- |
|  | Object | Map |
| Keys | Must be string or symbol (JavaScript automatically converts keys to a string a value.) | * Any value * ordered |
| size property | No | yes |
| Iterable | No. Use strict like Object.keys | yes |
| Prototype | Yes | No |
| (other) |  | better addition or removal of key pairs |

1. **compare Array and Set**

|  |  |  |
| --- | --- | --- |
|  | Array | Set |
| Index | Yes (in order) | no |
| Present | [1,2,3,4,5,6] | {1,2,3,4,5,6} |
| Duplicate Item | Yes [1,1,2,2,3] | No (distinct) {1,2,3} |
| QueryBy | Index | key |
| Constructor | [1,2,3,4,5], Array.from(), new Array() | new Set([1,2,3,4,5]) |
|  |  |  |

1. **arrow functions**

* shorter syntax for a function expression and does **not have its own this, arguments, super, or new.target**. So, not used to contructor or method in class or object

1. **What is the Temporal Dead Zone**

* that occurs when declaring a variable with the let and const
* The time span when that happens, between the creation of a **variable’s binding** and **its declaration**, is called the temporal dead zone

1. **differences between cookie, local storage and session storage**

|  |  |  |  |
| --- | --- | --- | --- |
|  | cookie | local storage | session storage |
| Accessed | Client and server | Client | Client |
| Lifetime | Manual | Manual | Until tab is closed |
| Data size | 4kb | 5MB | 5MB |

1. **What is the difference between null and undefined**

|  |  |  |
| --- | --- | --- |
|  | null | undefined |
| Value | Dc khai bào và dc gán giá trị là null ko chỏ đến obj nào | Dc khai báo nhưng Ko dc gán giá trị |
| Type | object | undefined |
| Operations (+ - \* /) | Convert to 0 | Convert to NaN |

1. **What is the difference between window and document**

|  |  |
| --- | --- |
| window | document |
| Root element | Direct child of window Document Object Model(DOM) |
| Mặc định, có thể gọi mà ko cần từ khoá window ở trước | Phải thông qua **window.document** or **document.** |
| Có các phương thức, thuộc tính như alert(), confirm() and conlog(), document, location | Có các phương thức như getElementById, getElementsByTagName, createElement.  Phải thêm **document** trước các phương thức này |

Note: những methods này do webAPI cung cấp, ko phải do JS

1. **What is event bubbling**

* Events trigger from the **child element** to the **parent element (in to out)**

What is event capturing

* Events trigger from the **parent element** to the **child element (out to in)**

1. **IIFE(Immediately Invoked Function Expression)**

* Private data, variables
* Use as module because we can not access variables from outside