

Практична робота 15. Використання бібліотеки jQuerry

1. На HTML-сторінці є посилання <a>. У випадку коли, href починається на https://, потрібно додати посиланню атрибут target="_blank".

2. Для заданої сторінки знайдіть всі <h2> з класом head, зробіть для них зелений колір фону, потім серед знайдених елементів знайдіть елементи з класом inner і поставте їм розмір шрифту 35px.

```
.   <h2 class="head">header1</h2>
<h2 class="head">header2 <span class="inner">inner elem1</span></h2>
<h2>header3</h2>
<h2 class="head">header4<span>inner elem2</span></h2>
<h2>header5</h2>
```

3. Знайдіть теги <div>, які стоять безпосередньо після <h3> і перемістіть кожен <div>-елемент так, щоб він став безпосередньо над <h3>.

```
<h3>header1</h3>
<div>text1</div>
<h3>header2</h3>
<div>text2</div>
```

4. На HTML-сторінці є 6 чекбоксів. Напишіть скріпт, який після того, як користувач позначив будь-які 3 чекбокси, всі чекбокси робить неактивними.

Вихідний код:

```
let arr = ["Tom", "Sam", "Ray", "Bob"];
let /* Ваш код */ = arr;
console.log(x); // "Tom"
console.log(y); // "Sam"

console.log(z); // [Bob]
```

5. Допишіть код використовуючи деструктурування, щоб в консолі браузера з'явилися рядки, які написані в коментаражах.

6. Вихідний код:

```
let data = {
    names: ["Sam", "Tom", "Ray", "Bob"],
    ages: [20, 24, 22, 26],
};

let /* Ваш код */ = data;
console.log(name2); // "Tom"
console.log(age2); // 24
console.log(name4); // "Bob"
console.log(age4); // 26
```

Допишіть код використовуючи деструктурування, щоб в консолі браузера з'явилися рядки, які написані в коментарях.

7. Напишіть функцію `mul()`, яка приймає будь-яку кількість параметрів різного типу і повертає добуток параметрів типу `Number`.

Якщо параметри типу `Number` відсутні, повертає число 0.

```
function mul() {
    // Ваш код
}
console.log(mul(1, "str", 2, 3, true)); // 6
console.log(mul(null, "str", false, true)); // 0
```

Не можна використовувати властивість `arguments`, але в функцію `mul()` можна додати один параметр.

8.

Вихідний код:

```
let server = {
    data: 0,
    convertToString: function (callback) {
        callback((function () {
            return this.data + "";
        })).bind(this));
    }
};
let client = {
    server: server,
    result: "",
    calc: function (data) {
        this.server.data = data;
        this.server.convertToString(this.notification());
    },
    notification: function () {
        return (function (callback) {
            this.result = callback();
        }).bind(this);
    }
};
client.calc(123);
console.log(client.result); // "123"
console.log(typeof client.result); // "string"
```

Змініть код використовуючи стрілкові функції, щоб в коді не використовувалися методи `bind()`.

9. Напишіть функцію mapBuilder (keysArray, valuesArrays), яка приймає два масиви однакової довжини. Використовуючи ці масиви, функція повинна створювати об'єкт типу Map, **ключами** якого є значення з **першого масиву**, а **значеннями** Map - значення з **другого масиву**. Після цього функція повертає даний об'єкт Map.

Приклади використання функції:

```
let keys = [1, 2, 3, 4];
let values = ["div", "span", "b", "i"];
let map = mapBuilder(keys, values);
console.log(map.size); // 4
console.log(map.get(2)); // "span"
```