a. What is the eval() method?

In JavaScript, eval() is a global function that allows you to execute code dynamically by taking a string as an argument and interpreting it as JavaScript code. Here's how it works:

```
var x = 10;
var y = 20;
var code = "var result = x + y; result * 2;"; // This is a string containing JavaScript code.
var dynamicResult = eval(code); // The eval() function interprets the code string.
console.log(dynamicResult); // This will output the result of the evaluated code. In this example, eval() takes the code variable, which is a string containing JavaScript code, and executes it.
```

In the end, dynamicResult contains the result of the evaluated code.
However, it's essential to be cautious when using eval() because it can be a security risk. When executing code from user input or external sources, it can potentially introduce security vulnerabilities if not used carefully. In many cases, there are safer alternatives to achieve dynamic code execution without using eval(), such as using functions or JSON.parse() for structured data.

b. What is the purpose of prev() and next() methods in jQuery?

In jQuery, the `prev()` and `next()` methods are used to navigate to and manipulate the siblings of an HTML element. They allow you to select and work with the previous and next sibling elements of the currently selected element.

1. `prev()`: The `prev()` method selects the immediately preceding sibling of the selected element. It is used to access the element that comes just before the currently selected element in the DOM tree. For example:

```javascript
$("#myElement").prev().addClass("highlight"); // Adds a class to the previous sibling
```

2. `next()`: The `next()` method selects the immediately following sibling of the selected element. It allows you to access the element that comes just after the currently selected element in the DOM tree. For example:

```javascript
$("#myElement").next().addClass("highlight"); // Adds a class to the next sibling
```

You can use these methods to traverse and manipulate elements in your page's structure. They are especially useful when you have a series of elements and you want to perform operations on adjacent siblings, such as highlighting them, changing their content, or applying other modifications.

c. How to delete all rows in a table except first row?

```
$("#myTable tr:gt(0)").remove();
$("#myTable tr:not(:first)").remove();
```