Document

Write a code to get like below using

JOIN METHOD:

1)i/p: var fruits= [“Apple”,” banana”,” orange”]

o/p:” Apple, banana, orange”;

CODE: var fruits = ["Apple", "banana", "orange"];

var result = fruits. join (", ");

console.log(result);

EXPLANATION: The code defines an array `fruits` containing three fruit names. It then uses the `join` method to concatenate the elements into a single string, separated by a comma and a space. Finally, it prints the resulting string, which is `"Apple, banana, orange"` to the console.

2) i/p: var number= [“1,2,3,4,5”]

o/p:” 1-2-3-4-5”;

CODE: var number = ["1,2,3,4,5"];

var result = split Numbers. join ("-");

console.log(result);

EXPLANATION: The code starts with an array containing a single string of numbers separated by commas. It splits this string into an array using the `split` method and then joins the elements with a hyphen using the `join` method. Finally, it prints the resulting string, which is `"1-2-3-4-5"` to the console.

3)i/p: var chars= [‘h’, ’e’, ’l’,’ l’, ’o’];

o/p:” hello”

CODE: var chars = ['h', 'e', 'l', 'l', 'o'];

var result = chars. join ('');

console.log(result);

EXPLANATION: The code initializes an array `chars` containing individual characters of the word "hello." It uses the `join (‘‘) ` method to concatenate these characters into a single string without any separators. Finally, it prints the resulting string, which is `"hello"`, to the console.

SLICE METHOD:

1)i/p: var text=” JavaScript is awesome”

o/p:” JavaScript”

CODE: var text = "JavaScript is awesome";

var result = text. slice (0, 10);

console.log(result);

EXPLANATION: The code defines a string `text` containing "JavaScript is awesome." It uses the `slice (0, 10) ` method to extract the substring "JavaScript" from the beginning of the string up to index 10. Finally, it prints the result to the console.

2)i/p: i/p: var sentence=" I love learning JavaScript"

o/p:" learning"

CODE: var sentence = "I love learning JavaScript";

var result = sentence. Slice (7, 16);

console.log(result);

EXPLANATION: The code defines a string `sentence` containing "I love learning JavaScript." It uses the `slice (7, 16) ` method to extract the substring "learning" from the specified indices. Finally, it prints the result to the console.

3)i/p: var text=" frontend development" ;(using negative index)

o/p:" development"

CODE: var text = "frontend development";

var result = text. Slice (-11);

console.log(result);

EXPLANATION: The code defines a string `text` containing "frontend development." It uses the `slice (-11) ` method, which starts slicing from the 11th character from the end, extracting "development." Finally, it prints the result to the console.

SPLIT METHOD:

1)var date="2024-10-21"

o/p: ["2024","10","21"]

CODE: var date = "2024-10-21";

var result = date. Split ("-");

console.log(result);

EXPLANATION: The code defines a string `date` containing "2024-10-21." It uses the `split ("-") ` method to divide the string into an array based on the hyphen delimiter, resulting in ` ["2024", "10", "21"]. ` Finally, it prints the resulting array to the console.

2)var text='I love JavaScript programming';

o/p: ["I", "love"]

CODE: var text = 'I love JavaScript programming'

var words = text. Split ("

var result = words. Slice (0, 2);

console.log(result);

EXPLANATION: The code defines a string `text` containing "I love JavaScript programming." It uses the `split (" “) ` method to separate the string into an array of words based on spaces. Then, it extracts the first two words using `slice (0, 2) ` and prints the result, which is ` ["I", "love”] `, to the console.

3)let URL=’https://www.example.com/path/page.html’;

o/p: [“https:”,””,www.example.com,” path”,” page.html”]

CODE: let URL = 'https://www.example.com/path/page.html';

let result = URL. Split (/ [: \/] +|\. |\//);

console.log(result);

EXPLANATION: The code defines a string `URL` containing a web address. It uses the `split` method with a regular expression to separate the URL into components based on colons, slashes, and periods. Finally, it prints the resulting array, which includes `["https:", "", "www.example.com", "path", "page.html"]`, to the console.