1. What is image map in HTML? Explain with an example.

Answer:An image map is that you should be able to perform different actions depending on where in the image you click.To create an image map you need an image,and some HTML code that describes the clickable areas.
Example:

```
<img src="workplace.jpg" alt="Workplace" usemap="#workmap">
<map name="workmap">
  <area shape="rect" coords="34,44,270,350" alt="Computer" href="computer.htm">
  <area shape="rect" coords="290,172,333,250" alt="Phone" href="phone.htm">
  <area shape="circle" coords="337,300,44" alt="Coffee" href="coffee.htm">
</map>
```

2. Roman numerals are represented by seven different symbols: I, V, X, L, C, D and M. Symbol Value I 1 V 5 X 10 L 50 C 100 D 500 M 1000 For example, 2 is written as II in Roman numeral, just two ones added together. 12 is written as XII, which is simply X + II. The number 27 is written as XXVII, which is XX + V + II. Roman numerals are usually written largest to smallest from left to right. However, the numeral for four is not IIII. Instead, the number four is written as IV. Because the one is before the five we subtract it making four. The same principle applies to the number nine, which is written as IX. There are six instances where subtraction is used: I can be placed before V (5) and X (10) to make 4 and 9. X can be placed before L (50) and C (100) to make 40 and 90. C can be placed before D (500) and M (1000) to make 400 and 900. Given a roman numeral, convert it to an integer.

Answer:

```
var romanToInt = function(s) {

    const maps = {
        'I' : 1,
        'V' : 5,
        'X' : 10,
        'L' : 50,
        'C' : 100,
        'D' : 500,
        'M' : 1000
    }

    let roman_list = s.split('')

    console.log("roman_list", roman_list)

    let eng_obj = {}

    roman_list.map((r, i) => {
        return eng_obj[i] = maps[r]
    })

    console.log("eng_obj", eng_obj)
```

```
    let integer_value = 0
    let skip = ''

    for (const key in eng_obj) {
        console.log('integer_value = ', integer_value)
        console.log('eng_obj = ', eng_obj[key])


        let next = (Number(key) + 1).toString()
        console.log("next key = ", next)
        if (eng_obj[next] > eng_obj[key]) {
            skip = next
            integer_value += (eng_obj[next] - eng_obj[key])
        } else {
            if(key !== skip) {
                integer_value = integer_value + eng_obj[key]
            }
        }

    }


    return integer_value
};

const result = romanToInt('IX')
console.log(result)
```

3. You are climbing a staircase. It takes n steps to reach the top. Each time you can either climb 1 or 2 steps. In how many distinct ways can you climb to the top? Example 1: Input: n = 2 Output: 2 Explanation: There are two ways to climb to the top. 1. 1 step + 1 step 2. 2 steps Example 2: Input: n = 3 Output: 3 Explanation: There are three ways to climb to the top. 1. 1 step + 1 step + 1 step 2. 1 step + 2 steps 3. 2 steps + 1 step

Answer:

```
let climbStairs = (n, count = 0, memory = []) => {
    //if step is greater than stairs then return 0
    if(count > n) {
        return 0;
    }
```

```
    //if step is eqaul to the stairs then return 1
    if(count === n) {
        return 1;
    }

    //if the value is present for the given step then return it
    if(memory[count] > 0) {
        return memory[count];
    }

    //compute the value for the given step and save it
    memory[count] = climbStairs(n, count + 1, memory) + climbStairs(n,
count + 2, memory);

    return memory[count];
 };
console.log(climbStairs(5))
```

4. What are the benefits of CSS Sprites?

Answer: The concept of CSS Sprites is used to reduce the loading time for a web page because it combines the various small images into one image.It reduces the number of http requests and hence the loading time.

5. What are the limitations of CSS?

Answer:CSS has various limitations as a programming language thats are as follows:

- CSS cannot perform any logical operations like if/else or for/while or +/-.
- We can not read any files using CSS.
- It can not interact with databases.
- CSS can not request a web page.

6. Explain passed by value and passed by reference.

Answer:
"Passing by value" means that you pass the actual value of the variable into the function. So, in your example, it would pass the value 9.

"Passing by reference" means that you pass the variable itself into the function (not just the value). So, in your example, it would pass an integer object with the value of 9.

7. What is an Immediately Invoked Function in JavaScript?

Answer:An Immediately Invoked Function Expression is a JavaScript function that runs as soon as it is defined. example:() => {} ()

8. What do you mean by strict mode in javascript and characteristics of javascript strict-mode?

Answer:Strict mode changes previously accepted "bad syntax" into real errors. As an example, in normal JavaScript, mistyping a variable name creates a new global variable. In strict mode, this will throw an error, making it impossible to accidentally create a global variable.

9. Explain Higher Order Functions in javascript

Answer:In Javascript, functions can be assigned to variables in the same way that strings or arrays can. They can be passed into other functions as parameters or returned from them as well. A “higher-order function” is a function that accepts functions as parameters and/or returns a function.

10. List some features of WordPress with examples.

Answer:Easily Manage Your Website Content.Unlimited Pages, Posts, Products & More. Integrated News & Blog Posting. Flexible Post Scheduling. Restore Deleted Pages & Rollback Versioning.Website Portability & Vendor Lock-in Avoidance. Easily Add Additional Website Editors. Managed Website Security.

11. What is a plugin in WordPress? List plugin that comes with WordPress.

Answer:A WordPress plugin is **a piece of software that “plugs into” your WordPress site**. Plugins can add new functionality or extend existing functionality on your site, allowing you to create virtually any kind of website, from ecommerce stores to portfolios to directory sites.

WordPress plugin list:

- WPForms. Contact form is essential for every website as it is the channel for your viewers to be in touch with you. ...
- Yoast SEO. Search Engine Optimization plays a vital role in boosting visitors to your website. ...
- Akismet. ...
- WP Rocket. ...
- UpdraftPlus. ...
- Migrate Guru. ...
- WooCommerce. ...
- Redirection.

12. What are the types of hooks in WP and what are their functions?

Answer:There are two types of hooks 1) Action hooks 2) Filter hooks
Hooks allow a user to create WordPress theme or plugin with shortcode without changing the original files. Action hooks allow you to insert an additional code from an outside resource, whereas, Filter hooks will only allow you to add content or text at the end of the post.

13. How do you enable debug mode in WP?

Answer:For enabling debug mode in WordPress, developers have to edit the `wp-config.php` file and modify the `WP_DEBUG` constant value to true condition. For enabling debug mode in WordPress, developers have to edit the `wp-config.php` file and modify the `WP_DEBUG` constant value to true condition.

14. What is the difference between "echo" and "print" in PHP?

Answer:`echo` and `print` are more or less the same. They are both used to output data to the screen.

The differences are small: `echo` has no return value while `print` has a return value of 1 so it can be used in expressions. `echo` can take multiple parameters (although such usage is rare) while `print` can take one argument. `echo` is marginally faster than `print`.

15. What is the difference between "$message" and "$$message"?

Answer:$message is used to store variable data. $$message can be used to store variable of a variable. Data stored in $message is fixed while data stored in $$message can be changed dynamically.

16. What are magic constants in PHP?

Answer:Magic constants are **the predefined constants in PHP which get changed on the basis of their use**. They start with double underscore (__) and ends with double underscore. They are similar to other predefined constants but as they change their values with the context, they are called magic constants.

17. What are the various types of relationships in Database? Define them.

Answer:There are three types of database relationships:

1.one-to-one.

2.one-to-many.

3.Many-to-many.

One-to-one: when one row in a table may be linked with only one row in another table and vice versa. For example: Suppose we have two tables, Users and Students. One user’s record is only related to only one student and also one student's data is only related to a single user record.

One-to-many: a one-to-many relationship occurs when a parent record in one table can potentially reference several child records in another table. For example: Suppose we have two tables, Users and Orders. One user can have multiple order records in the orders table.

Many-to-many: A many-to-many relationship occurs when multiple records in a table are associated with multiple records in another table. For example: Suppose we have two tables Teachers and Classes. One teacher can have multiple class records and also one class can have multiple teacher records.

18. Explain Normalization and De-Normalization with examples.

Answer: Normalization is used to remove redundant data from the database and to store non-redundant and consistent data into it. Denormalization is used to combine multiple table data into one so that it can be queried quickly.

19. What do you understand by Data Independence? What are its two types?

Answer:The ability to modify schema definition in one level without affecting schema of that definition in the next higher level is called data independence. There are two levels of data independence, they are Physical data independence and Logical data independence.

20. What are Python namespaces?

Answer:Since every name introduced has a place where it lives and can be hooked for, in Python. This is referred as namespace. Namespace is like a box where a variable name is mapped to the object placed. Therefore whenever the variable is searched out, this box will be searched, to get corresponding object.

21. What are Dict and List comprehensions?

Answer: They are syntax constructions to ease the creation of a Dictionary or List based on existing iterable

22. What is slicing in Python?

Answer:It is a single expression anonymous function often used as an inline function.

23. How is memory managed in Python?

Answer:Steps in managing memory in Python are -

- The Python memory is primarily managed by Python private heap space.
- All Python objects and data structures are located in a private heap.
- The programmer does not have access to this private heap and interpreter takes care of this Python private heap.
- The allocation of Python heap space for Python objects is done by Python memory manager.
- The core API gives access to some tools for the programmer to code.
- Python has an inbuilt garbage collector, that recycles all the unused memory and frees the memory and makes it available to the heap space

24. What is the difference between Python Arrays and lists?

Answer:Arrays can hold only a single data type element whereas lists can hold any data type elements

25. A peak element is an element that is strictly greater than its neighbors.

Given a 0-indexed integer array nums, find a peak element, and return its index. If the array contains multiple peaks, return the index to any of the peaks. You may imagine that nums[-1] = nums[n] = -∞. In other words, an element is always considered to be strictly greater than a neighbor that is outside the array. You must write an algorithm that runs in O(log n) time. Example: Input: nums =

[1,2,3,1] Output: 2 Explanation: 3 is a peak element and your function should return the index number 2.

Answer:

```
  const findPeakElement = function(nums) {

   let lowValue = 0, highValue = nums.length-1, midValue;

   while(lowValue < highValue) {
      midValue = Math.floor((lowValue + highValue)/2);
      if(nums[midValue] > nums[midValue+1]) highValue = midValue;
      else lowValue = midValue+1;
   }

   return lowValue;
};

console.log(findPeakElement([1,2,9,3,2,4])
```