**GUVI**

**Zen class: Full stack development**

**Red- Action items**

**Batch-28:** Classes starts on Sep 11 and joined on Sep 18.

1st and 2nd class not attended and have to see on it. Task updated in portal

First 2 class mentor is Venkat

3rd session: Sai Mohan(Mentor):

Topics covered:

* Hoisting
* Const and Let are block of scope/ function scope, var Global scope
* Objects and JSON:  
  Objects key does not have double quotes but JSON does have double quotes in it(notation). JSON used for API,
* Touch based:

Array(declare, index and length)

Function

Slice method.

Method and function both are same per my understanding

Task submitted.

Scribbles/ works which done during the session:

/\* Array length check

var a =[1,2];

[10]=91;

console.log(a.length);

\*/

/\* Write a JavaScript function to get the first element of an array. Passing a parameter 'n' will return the first 'n' elements of the array. – Program will work on it/ attach Mohan’s program- Done please see below

var first1= [1,2,3,4];

console.log(first1.slice(0,2));

\*/

**Program solving:**

/\*1. Writes a JS function to get the last element of an array. Passing a parameter 'n' will return the first 'n' elements of the array. \*/- Program done by Mohan.

function compute(arr,n){

if(n ==null){return arr[0];}

if(n<0){return [];}return arr.slice(0,n);}

console.log(compute([0,6,4,5],3));

**4th session: Sai Mohan(Mentor):**

**Topics covered:**

* XHR(XML Http request)
* Rest countries API
* Array of objects
* New keyword it create a object(specific class)-Explained in XML
* Get method , onload, response, send and open- Explained in XML
* -For of application for objects , array but not for simple key value pair. However, For in will do for simple objects as well.
* For in and for of are both forward and unidirectional. And for loop is forward and backward directional

Task uploaded.

Program related to JSON XML requested done in VS code .

Path: D:\Full stack development\VS\_Code files\XML HTTP request

Scribbles/ works which done during the session:

Below done in Guvi ide

var sample= [

{

"id": 1,

"title": "sunt aut facere repellat provident occaecati excepturi optio reprehenderit",

"body": "quia et suscipit\nsuscipit recusandae consequuntur expedita et cum\nreprehenderit molestiae ut ut quas totam\nnostrum rerum est autem sunt rem eveniet architecto"

},

{

"id": 2,

"title": "qui est esse",

"body": "est rerum tempore vitae\nsequi sint nihil reprehenderit dolor beatae ea dolores neque\nfugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis\nqui aperiam non debitis possimus qui neque nisi nulla"

},

{

"id": 3,

"title": "ea molestias quasi exercitationem repellat qui ipsa sit aut",

"body": "et iusto sed quo iure\nvoluptatem occaecati omnis eligendi aut ad\nvoluptatem doloribus vel accusantium quis pariatur\nmolestiae porro eius odio et labore et velit aut"

},

{

"id": 4,

"title": "eum et est occaecati",

"body": "ullam et saepe reiciendis voluptatem adipisci\nsit amet autem assumenda provident rerum culpa\nquis hic commodi nesciunt rem tenetur doloremque ipsam iure\nquis sunt voluptatem rerum illo velit"

},

{

"id": 5,

"title": "nesciunt quas odio",

"body": "repudiandae veniam quaerat sunt sed\nalias aut fugiat sit autem sed est\nvoluptatem omnis possimus esse voluptatibus quis\nest aut tenetur dolor neque"

},

{

"id": 6,

"title": "dolorem eum magni eos aperiam quia",

"body": "ut aspernatur corporis harum nihil quis provident sequi\nmollitia nobis aliquid molestiae\nperspiciatis et ea nemo ab reprehenderit accusantium quas\nvoluptate dolores velit et doloremque molestiae"

},

{

"id": 7,

"title": "magnam facilis autem",

"body": "dolore placeat quibusdam ea quo vitae\nmagni quis enim qui quis quo nemo aut saepe\nquidem repellat excepturi ut quia\nsunt ut sequi eos ea sed quas"

},

{

"id": 8,

"title": "dolorem dolore est ipsam",

"body": "dignissimos aperiam dolorem qui eum\nfacilis quibusdam animi sint suscipit qui sint possimus cum\nquaerat magni maiores excepturi\nipsam ut commodi dolor voluptatum modi aut vitae"

},

{

"id": 9,

"title": "nesciunt iure omnis dolorem tempora et accusantium",

"body": "consectetur animi nesciunt iure dolore\nenim quia ad\nveniam autem ut quam aut nobis\net est aut quod aut provident voluptas autem voluptas"

},

{

"id": 10,

"title": "optio molestias id quia eum",

"body": "quo et expedita modi cum officia vel magni\ndoloribus qui repudiandae\nvero nisi sit\nquos veniam quod sed accusamus veritatis error"

}]

console.log(sample[sample.length-1].id+" "+sample[sample.length-1].title+" "+sample[sample.length-1].body);

iterate things to get the all the values details.

Program solving:

/\*2. Writes a JS function to get the last element of an array. Passing a parameter 'n' will return the last 'n' elements of the array. \*/- Have to try it.

/\*function last(arr,n)

{

if(n===null)

{

return arr.length-1;

}

}

console.log(last([0,1,2,3],0));\*/

**5th session:**

Python tutor we can use to debug to see the code in clear way.

**Topics discussed:**

* Rest countries and country layer site
* Anonym’s function
* Arrow function
* IIFE

**Open weather:**open weather API key: 0b0d22a710f53652f08facfa70b0e2d1

Pre read: Template literals for tomorrow session.- Done

Task has to be uploaded/submitted.

Scribbles/ works which done during the session:

/\*Write a JS func to find the most frequent item in an array using normal and anonymous function\*/- Have to try it.

**6th session:**

**Topics discussed:**-Template literals(Kind of easy concat)

-Exception handling(try, catch and throw keyword)

-Destructing array (spread operator).

-rest parameter or spread operator, return an array

-ES 6 expressions

Scribbles/ works which done during the session: None

Task has to be submitted

**7th session:**

**Topics discussed:**

-