**GUVI: Zen Class — Part 1: Find the culprits and nail them — debugging javascript**

Once you are familiar with basic syntax you can reinforce your understanding by solving these simple snippets

**Find the culprit**

fix.html

<!DOCTYPE html>  
<html>  
<body>  
 <script>  
 alert( “I’m JavaScript!’);  
 </script>  
 Whats the error in this ?  
</body>  
</html>

— — Answer - alert( "I’m JavaScript!");

— — — — — — — — — — — — — — — — — — — — — — — — — — — — -

**Find the culprit and invoke the alert**

fix.html

<!DOCTYPE html>  
<html>  
<body>  
 <script src=”script.js”></script>  
</body>  
</html>

scripts.js

alert(“I’m invoked!”);

— Answer - <script src="scripts.js"></script>

— — — — — — — — — — — — — — — — — — — — — — — — — — — — — -

**Explain the below how it works**

explain.html

<!DOCTYPE html>  
<html>  
<body>  
 <script src=”script.js”></script>  
</body>  
</html>

script.js

alert("I'm JavaScript!");  
alert('Hello') // this line is not having semicolon  
alert(`Wor  
 ld`)  
alert(3 +  
1  
+ 2); // this is multiple line code and its working

* The **alert()** method in JavaScript is used to display a virtual alert box. It is mostly used to give a warning message to the users. It displays an alert dialog box that consists of some specified message (which is optional) and an OK button. When the dialog box pops up, we have to click "OK" to proceed.
* Semicolon not mandatory because it is method of a function.

**Fix the below to alert**Guvi geek

fix.html

<!DOCTYPE html>  
<html>  
<body>  
 <script src=”script.js”></script>  
</body>  
</html>

script.js

let admin=9, fname=10.5;   
fname = "Guvi";  
lname = "geek"  
admin = fname+lname;alert( admin ); // "Guvi geek"

**Answer – To add one space “guvi ” end or “ geek” before geek**

**Fix the below to alert**hello Guvi geek

fix.html

<!DOCTYPE html>  
<html>  
<body>  
 <script src=”script.js”></script>  
</body>  
</html>

script.js

let fname=10.5;   
fname = "Guvi";  
lname = "geek"let name = fname+lname;alert( 'hello ${name}' );

**Ans – lname end of statement semicolon missing and ${name} - it is template literal this type supported within `hello ${name}`**

let fname=10.5;   
 fname = "Guvi";  
 lname = "geek";

let name = fname+lname;

alert( `hello ${name}` );

**Fix the below to alert sum of two numbers**

fix.html

<!DOCTYPE html>  
<html>  
<body>  
 <script src=”script.js”></script>  
</body>  
</html>

script.js

let a = prompt("First number?");  
let b = prompt("Second number?");  
alert(a + b);

Answer = to mention type of value like

Let a = parseInt(prompt(“FirstNumber”)  
let b = parseInt(prompt(“FirstNumber”)  
alert(a + b);

**Fix the below to alert sum of two numbers**

fix.html

<!DOCTYPE html>  
<html>  
<body>  
 <script src=”script.js”></script>  
</body>  
</html>

script.js

let a = prompt("First number?");  
let b = prompt("Second number?");  
alert(a + b);

Answer = to mention type of value like

Let a = parseInt(prompt(“FirstNumber?”);  
let b = parseInt(prompt(“FirstNumber?”);  
alert(a + b);

**If you run the below scritpt you will get “**Code is Blasted**”**

**Explain Why the Code is blasted and how to diffuse it and get “**Diffused**”.**

fix.html

<!DOCTYPE html>  
<html>  
<body>  
 <script src=”script.js”></script>  
</body>  
</html>

script.js

var a = "2" > "12";//Don't touch below this  
if (a) {  
 console.log("Code is Blasted")  
}  
else  
{  
 console.log("Diffused")   
}

Answer – Now the condition is true so output true(“code is blased”)

* Suppose the condition is false example- var a = “2”<”12”; display the output is else part (“diffused”)

**How to get the success in console.**

fix.html

<!DOCTYPE html>  
<html>  
<body>  
 <script src=”script.js”></script>  
</body>  
</html>

script.js

let a = prompt("Enter a number?");//Don't modify any code below thisif (a) {  
 console.log( 'OMG it works for any number inc 0' );  
}  
else  
{  
 console.log( "Success" );  
}

Answer – To Give empty prompt value and just click ok it will be provide (“Success”)

**How to get the correct score in console.**

fix.html

<!DOCTYPE html>  
<html>  
<body>  
 <script src=”script.js”></script>  
</body>  
</html>

script.js

let value = prompt('How many runs you scored in this ball');  
if (value === 4) {  
 console.log("You hit a Four");  
} else if (value === 6) {  
 console.log("You hit a Six");  
} else {  
 console.log("I couldn't figure out");  
}

Answer = let value = +prompt('How many runs you scored in this ball');  
if (value === 4) {  
 console.log("You hit a Four");  
} else if (value === 6) {  
 console.log("You hit a Six");  
} else {  
 console.log("I couldn't figure out");  
}

Otherwise = it is automatically taken type of string strings consist of ===

let value = prompt('How many runs you scored in this ball');  
if (value == 4)// if (value ===”4”) {  
 console.log("You hit a Four");  
} else if (value == 6) {  
 console.log("You hit a Six");  
} else {  
 console.log("I couldn't figure out");  
}

**Fix the code to welcome the Employee**

fix.html

<!DOCTYPE html>  
<html>  
<body>  
 <script src=”script.js”></script>  
</body>  
</html>

script.js

let login = 'Employee';  
let message = (login == 'Employee') ? :  
 (login == 'Director') ? 'Greetings' :  
 (login == '') ? 'No login' :  
 '';console.log(message);

Answer : let login = 'Employee';

let message = (login == 'Employee')?  'Welcome the Employee'  : (login == '') ? 'No login' : '';

  console.log(message);

**Fix the code to welcome the boss**

fix.html

<!DOCTYPE html>  
<html>  
<body>  
 <script src=”script.js”></script>  
</body>  
</html>

script.js

// You cant change the value of the msg  
let message;if (null || 2 || undefined )  
{  
 let message = "welcome boss";  
}  
else  
{  
 let message = "Go away";  
}  
 console.log(message);

Answer : var only accepted reinitialize and redeclaration

var message;

if (null || 2 || undefined )

{

 var message = "welcome boss";

}

else

{

 var message = "Go away";

}

  console.log(message);

**Fix the code to welcome the boss**

fix.html

<!DOCTYPE html>  
<html>  
<body>  
 <script src=”script.js”></script>  
</body>  
</html>

script.js

let message;  
let lock = 2;//Dont change any code below this if (null || lock || undefined )  
{  
 message = "Go away";  
}  
else  
{  
 message = "welcome";  
}  
 console.log(message);

Ans : let message;

  let lock ="";//Dont change any code below this

  if (null || lock || undefined )

  {

    message = "Go away";

  }

  else

  {

   message = "welcome";

  }

    console.log(message);

let message;

  let lock;//Dont change any code below this

  if (null || lock || undefined )

  {

    message = "Go away";

  }

  else

  {

   message = "welcome";

  }

    console.log(message);

* Let message also undefined but not through the error it is waiting for lexical scope

— — — — — — — — — — — — — — — — — — — — — — — — — — — — — — -

**Fix the code to welcome the boss**

fix.html

<!DOCTYPE html>  
<html>  
<body>  
 <script src=”script.js”></script>  
</body>  
</html>

script.js

let message;  
let lock = 2;//Dont change any code below thisif (lock && " " || undefined )  
{  
 message = "Go away";  
}  
else  
{  
 message = "welcome";  
}  
console.log(message);

Ans : let message;

  let lock ="";//Dont change any code below this

  if (null || lock || undefined )

  {

    message = "Go away";

  }

  else

  {

   message = "welcome";

  }

    console.log(message);

let message;

  let lock;//Dont change any code below this

  if (null || lock || undefined )

  {

    message = "Go away";

  }

  else

  {

   message = "welcome";

  }

    console.log(message);

* Let message also undefined but not through the error it is waiting for lexical scope

**Change the code to print**

3

2

1

fix.html

<!DOCTYPE html>  
<html>  
<body>  
 <script src=”script.js”></script>  
</body>  
</html>

script.js

//You can change only 2 characterslet i = 3;while (i) {  
 console.log( --i );  
}

Answer = let i = 4;

while (i>1) {

      console.log( --i );

}

3

2

1

— — — — — — — — — — — — — — — — — — — — — — — — — — — — — — -

**Change the code to print 1 to 10 in 4 lines**

fix.html

<!DOCTYPE html>  
<html>  
<body>  
 <script src=”script.js”></script>  
</body>  
</html>

script.js

let num = 1  
console.log(num)  
num += 1  
console.log(num)  
num += 1  
console.log(num)  
num += 1  
console.log(num)  
num += 1  
console.log(num)  
num += 1  
console.log(num)  
num += 1  
console.log(num)  
num += 1  
console.log(num)  
num += 1  
console.log(num)  
num += 1  
console.log(num)

//   output in row

let num = 10

for(let i=1; i<=num; i++)

{

      console.log(i)

}

// output in column

let num1 = " ";

for(let i=1; i<=10; i++)

{

      num1 +=" "+i ;

}console.log(num1.trim())

**Change the code to print even numbers**

fix.html

<!DOCTYPE html>  
<html>  
<body>  
 <script src=”script.js”></script>  
</body>  
</html>

script.js

//You are allowed to modify only one character for (let num = 2; num <= 20; num += 1) {  
 console.log(num)  
}

for(let num = 2; num <= 20; num += 1) {

      if(num%2==0)

      {

            console.log(num)

      }

}

let even = "";

for(let num = 2; num <= 20; num += 1) {

      if(num%2==0)

      even+=" "+num;

}

console.log(even.trim())

**Change the code to print all the gifts**

fix.html

<!DOCTYPE html>  
<html>  
<body>  
 <script src=”script.js”></script>  
</body>  
</html>

script.js

let gifts = ["teddy bear", "drone", "doll"];for (let i = 0; i < 3; i++) {  
 console.log('Wrapped ${'gifts[i]'} and added a bow!');  
}

Answer = let gifts = ["teddy bear", "drone", "doll"];

for (let i = 0; i < 3; i++) {

      console.log(`Wrapped ${gifts[i]} and added a bow!`);

    }

Output : Wrapped teddy bear and added a bow!

Wrapped drone and added a bow!

Wrapped doll and added a bow!

**Fix the code to disarm the bomb.**

fix.html

<!DOCTYPE html>  
<html>  
<body>  
 <script src=”script.js”></script>  
</body>  
</html>

script.js

let countdown = 100;while (countdown > 0) {  
 countdown--;  
 if(countdown == 0)  
 {  
 console.log("bomb triggered");  
 }  
}

ANS :

let countdown = 100;

while (countdown > 0) {

  console.log(countdown--)

}

if(countdown == 0)

  console.log("bomb triggered")

Whats the msg printed and why?

var lemein = “0”;  
var lemeout = 0;  
var msg = “”;if (lemein) {  
 msg += “hi”;  
 }if (lemeout) {  
 msg += ‘Hello’;  
}console.log(msg);

Output : hi – Because the lemein value is string

Whats the msg printed and why? Guess you answer before running it.

var lemein = “0”;  
var lemeout = 0;  
var msg = “”;if (lemein) {  
 msg += “hi”;  
 }if (lemeout) {  
 msg += ‘Hello’;  
}console.log(msg);

Output : hi – Because the lemein value is string and 0 is consist of nothing