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

1. Find the culprit

fix.html

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

Solution: Incorrect highlight and correct solution is provided below:

```
<script>
  alert("I'm JavaScript!");
</script>
```

— —

2. 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!”);
```

Solution: JavaScript file name is incorrect, highlight the same.

— —

3. 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
```

Solution: In JavaScript, there is 3 ways to create string that can span multiple line :

- Using template literals
- Using + operator
- Using \ operator

— —

4. 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"
```

Solution: Incorrect highlight and correct solution is provided below:

```
admin = fname + " " + lname;
```

---

5. **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}' );
```

Solution: We have to use template literals here. Incorrect highlight and correct solution are provided below:

```
alert(`hello ${name}`);
```

---

6. 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);
```

Solution: Incorrect highlight and correct solution is provided below:

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

---

7. 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")
}
```

Solution: In this code a is string and if we compare the string “2” with string “12” we will get “Code is Blasted”, as here comparison is not happing. To make it work we have to both the string into number. Like below then we get “Diffused” has output.

```
var a = 2 > 12;
```

---

8. 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 this
if (a) {
 console.log( 'OMG it works for any number inc 0' );
}
else
{
 console.log( "Success" );
}
```

Solution: Incorrect highlight and correct solution is provided below:

```
let a = parseInt(prompt("Enter a number?"));
if (a <= 0) {
```

---

9. 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");
}
```

Solution: Incorrect highlight and correct solution is provided below:

```
let value = parseInt(prompt('How many runs you scored in this ball'));
```

---

10. 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);
```

Solution: Incorrect highlight and correct solution is provided below:

```
let message = (login == 'Employee') ? 'Welcome' :
```

---

11. 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);
```

Solution: Incorrect highlight, no need to declare message again and again.

---

12. 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 (lock && " " || undefined )
{
  message = "Go away";
}
else
{
 message = "welcome";
}
console.log(message);
```

Solution: Incorrect highlight. Don’t assign any value to lock variable.

---

13. 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 characters
let i = 3;
while (i) {
  console.log( --i );
}
```

Solution: Incorrect highlight and correct solution is provided below:

```
while (i > 0) {
    console.log(i);
    i--;
}
```

---

14. 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)
}
```

Solution: Incorrect highlight and correct solution is increment the loop with + 2 :

```
for (let num = 2; num <= 20; num += 2)
```

---

15. 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!');
}
```

Solution: We have to use template literals here. Incorrect highlight and correct solution are provided below:

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

---

16. 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");
  }
}
```

Solution: Incorrect highlight and correct solution is provided below:

```
            while (countdown >= 0) {
```

17.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);
```

Solution: We will get below output:

hi

Initial msg has empty value. In fist condition it assign a string value and in second condition lemeout has number datatype and it is not doing any operation/comparison, so the msg get out from it condition without adding ‘Hello’ in the msg.
