JavaScript assignment 1

1.difference statically language in JavaScript and dynamically in JavaScript.

|  |  |
| --- | --- |
| Statically typed language | Dynamically typed language |
| A language is statically typed if the type of a variable is known at compile time. For some languages this means that you as the programmer must specify what type each variable is (e.g.: Java, C, C++); other languages offer some form of type inference, the capability of the type system to deduce the type of a variable (e.g.: OCaml, Haskell, Scala, Kotlin) | A language is dynamically typed if the type is associated with run-time values, and not named variables/fields/etc. This means that you as a programmer can write a little quicker because you do not have to specify types every time (unless using a statically-typed language with type inference).  Examples: Perl, Ruby, Python, PHP, JavaScript |
| The main advantage here is that all kinds of checking can be done by the compiler, and therefore a lot of trivial bugs are caught at a very early stage. | Most scripting languages have this feature as there is no compiler to do static type-checking anyway, but you may find yourself searching for a bug that is due to the interpreter misinterpreting the type of a variable. Luckily, scripts tend to be small so bugs have not so many places to hide. |
| Examples: C, C++, Java, Rust, Go, Scala | Most dynamically typed languages do allow you to provide type information, but do not require it. One language that is currently being developed, [Rascal](http://www.rascal-mpl.org/), takes a hybrid approach allowing dynamic typing within functions but enforcing static typing for the function signature. |
| Simply put it this way: in a **statically typed language** variables' types are static, meaning once you set a variable to a type, you cannot change it. That is because typing is associated with the variable rather than the value it refers to. | Where on the other hand: in a **dynamically typed language** variables' types are dynamic, meaning after you set a variable to a type, you CAN change it. That is because typing is associated with the value it assumes rather than the variable itself. |

2.difference between let and var and const keywords in JavaScript.

var and let are both used for variable declaration in JavaScript but the difference between them is that var is function scoped and let is block scoped.  
It can be said that a variable declared with var is defined throughout the program as compared to let.  
An example will clarify the difference even better  
**Example of var:**

**Input:**

console.log(x);

var x=5;

console.log(x);

**Output:**

undefined

5

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| KEYWORD | Scope | Can be Redeclared | Can be Reassigned | Hoisting Behavior |
| Var | Function & global | Yes | Yes | Initialized and undefined |
| Let | Block | No | Yes | Uninitialized |
| Const | Block | No | no | Uninitialized |

3.caluclator

<html lang="en">

<head>

    <meta charset="utf-8">

    <title>JavaSctipt Calculator</title>

    <meta name="viewport" content="width=device-width, initial-scale=1" />

    <link rel="stylesheet" href="style.css" />

    <style>

        body {

    background: #1e74b6;

}

.container {

    width: 300px;

    background: #1f2326;

    border-radius: 5px;

    box-shadow: 0 15px 50px 0 rgba(0,0,0,0.4), 0 15px 20px 0 rgba(0,0,0,0.2);

    position: fixed;

    top: 50%;

    left: 50%;

    transform: translate(-50%, -50%);

}

.heading {

    text-align: left;

    font-size: 22px;

    color: #000000;

    padding: 10px 0 0 10px;

}

.resultscreen {

    height: 110px;

    width: 100%;

    position: relative;

    border-top-left-radius: 5px;

    border-top-right-radius: 5px;

    background: #f8f6f6;

}

input[type='text'] {

    width: 300px;

    padding-right: 10px;

    position: absolute;

    bottom: 0;

    border: none;

    text-align: right;

    font-size: 40px;

    color: #000000;

    background: none;

}

.line {

    width: 300px;

}

input[type='button'] {

    float: left;

    font-size: 24px;

    line-height: 70px;

    margin: 1;

    width: 73px;

    height: 70px;

    border: none;

    color: #fafafa;

    cursor: pointer;

    outline: none;

    background: #2d2d2d;

}

input[type='button'][value='0'] {

    border-bottom-left-radius: 5px;

}

input[type='button'][value='='] {

    background: #ec2629;

    color: #ffffff;

    border-bottom-right-radius: 5px;

}

    </style>

</head>

<body>

    <div class="container">

        <form name="calculatorForm">

            <div class="resultscreen">

                <div class="heading">Calculator</div>

                <input type="text" name="evalresult" />

            </div>

            <div class="buttons">

                <div class="line">

                    <input type="button" name="btn7" value="7" onClick="valuecal(btn7.value)" />

                    <input type="button" name="btn8" value="8" onClick="valuecal(btn8.value)" />

                    <input type="button" name="btn9" value="9" onClick="valuecal(btn9.value)" />

                    <input type="button" name="btndivision" value="/" onClick="valuecal(btndivision.value)" />

                </div>

                <div class="line">

                    <input type="button" name="btn4" value="4" onClick="valuecal(btn4.value)" />

                    <input type="button" name="btn5" value="5" onClick="valuecal(btn5.value)" />

                    <input type="button" name="btn6" value="6" onClick="valuecal(btn6.value)" />

                    <input type="button" name="btnmultiply" value="\*" onClick="valuecal(btnmultiply.value)" />

                </div>

                <div class="line">

                    <input type="button" name="btn1" value="1" onClick="valuecal(btn1.value)" />

                    <input type="button" name="btn2" value="2" onClick="valuecal(btn2.value)" />

                    <input type="button" name="btn3" value="3" onClick="valuecal(btn3.value)" />

                    <input type="button" name="btnsubtract" value="-" onClick="valuecal(btnsubtract.value)" />

                </div>

                <div class="line">

                    <input type="button" name="btn0" value="0" onClick="valuecal(btn0.value)" />

                    <input type="button" name="btnDot" value="." onClick="valuecal(btnDot.value)" />

                    <input type="button" name="btnAddition" value="+" onClick="valuecal(btnAddition.value)" />

                    <input type="button" name="result" value="=" onClick="evalresult.value=eval(evalresult.value)" />

                </div>

            </div>

        </form>

    </div>

    <script>

        function valuecal(result) {

                calculatorForm.evalresult.value = calculatorForm.evalresult.value + result;

            }

    </script>

</body>

</html>

4.storing details

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

</head>

<body>

    <form action="">

<label for="name">enter name:</label>

<input type="text" placeholder="enter your name" id="title"/><br>

<label for="age">enter age:</label>

<input type="number" placeholder="enter your age" id="age" /><br>

<button id="btn">add</button>

<div id="msg">

    <pre></pre>

</div>

    </form>

    <script>

let data=[];

const addData = (ev)=>{

    ev.preventDefault();

    let data1={

        id:Date.now(),

        title: document.getElementById("title").value,

        age:document.getElementById("age").value

    }

data.push(data1);

document.forms[0].reset();

console.warn('added',{data});

let pre=document.querySelector('#msg pre');

pre.textContent='\n'+JSON.stringify(data,'\t',2);

//saving local storage

localStorage.setItem('mydata',JSON.stringify(data))

}

document.addEventListener('DOMContentLoaded',()=>{

    document.getElementById("btn").addEventListener('click',addData);

})

    </script>

</body>

</html>

4 another form validation and storing program

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

    <style>

section{

    border: 1px solid black;

    width: 50%;

    margin: 0 auto;

    height: 40vh;

}

h1{

    text-align: center;

    text-decoration: underline;

    font-weight: 600;

}

#in3{margin-left: 10px;

}#in4{

    margin-left: 5%;

}

    </style>

</head>

<body>

<section>

    <h1>FORM</h1>

<br><hr>

    <form >

      &nbsp;  First Name : <input id="in1" type="text" name="firstname" /><span id="firstnameerror"></span><br><br>

       &nbsp; Last Name : <input id="in2" type="text" name="lastname" /><span id="lastnameerror"></span><br><br>

       &nbsp; Your age &nbsp;  : <input id="in3" type="number" name="age" /><span id="ageerror"></span><br><br>

   &nbsp; Enter your mail:<input type="email" placeholder="enter your email" required><br><br>

       <input id="in4" type="submit" onclick="return validate()" value="submit your details">

        <br>

    </form>

<div id="msg1">

    <pre></pre>

</div>

</section>

<script>

function validate(){

var f1=document.forms[0];

var ferror=f1.firstname;

var firstnameerrorstatus;

    var e1 = document.getElementById("firstnameerror");

if(firstnameerrorstatus=(ferror.value.length == 0)){

    e1.style.color="red";

    e1.innerHTML="enter your firstname ";

    }else{

        e1.innerHTML="";

    }

    var ferror2 = f1.lastname;

    var lastnameerrorstatus;

    var e2 = document.getElementById("lastnameerror");

    if (lastnameerrorstatus = (ferror2.value.length == 0)) {

        e2.style.color = "red";

        e2.innerHTML = "enter your lastname ";

    } else {

        e2.innerHTML = "";

    }

    var ferror3 = f1.age;

    var agenameerrorstatus;

     var e3 = document.getElementById("ageerror");

    if (agenameerrorstatus = (ferror3.value.length ==0)) {

        e3.style.color = "red";

        e3.innerHTML = "enter your age ";

    } else {

        e3.innerHTML = "";

    }

    if (firstnameerrorstatus && lastnameerrorstatus && agenameerrorstatus) {

    }

  if(firstnameerrorstatus|| lastnameerrorstatus|| agenameerrorstatus){

return false;

  }

}

//storing

    if (validate() == false) {

            let data = [];

            const addData = (ev) => {

                ev.preventDefault();

                let data1 = {

                    id: Date.now(),

                    in1: document.getElementById("in1").value,

                    in2: document.getElementById("in2").value,

                    in3: document.getElementById("in3").value

                }

                data.push(data1);

                document.forms[0].reset();

                console.warn('added', { data });

                let pre = document.querySelector('#msg1 pre');

                pre.textContent = '\n' + JSON.stringify(data, '\t', 3);

                //saving local storage

                localStorage.setItem('mydata2', JSON.stringify(data))

            }

            document.addEventListener('DOMContentLoaded', () => {

                document.getElementById("in4").addEventListener('click', addData);

            })

        }

</script>

</body>

</html>