JavaScript

- it is a scripting language
- it object oriented language
- it functional language

variable

- it is a placeholder to store a value in memory
- it is mutable
- let and var are used to declare varibale
- const is also used to declare varibale e.g :

```
let number=11;
      var firstName="rayn"
      const cars= ['i20','i10','creta']
```

pre-defined objects

```
- console
    - object that represents browser console
    - menthod:
        - log(): used to print the message on the browser console
        - info()
        - warn()
        - error()
- window
     - it represents the browser's window (UI)
     - menthods:
        - alert()
        - prompt()
        - confirm()
```

pop ups

- alert e.g: alert("this is an alert")
- confirm e.g:

```
 const result = confirm("do you want to submit")
      if(result){
        console.log("hello world")
        }
         else{
        console.log("bye bye")
        }
```

pre-defined value

- undefined
- NaN

  e.g

```
  const price = 100;

  const fistName = "rayn";

const result= price * fistName

console.log(price + fistName)

console.log("🚀 ~ file: day14.js ~ line 8 ~ result", result)
```

- infinity e.g:

```
  let a=10;
  let b=0;

  console.log(10/0)
```

data types

- number e.g:

```
```javascript
    let a=10;
    console.log("🚀 ~ file: day14.js ~ line 15 ~ a", typeof(a))
```
```

- string
- boolean

operators

- addition (+)
- division (/)
- multiplication (*)

comparison oprators

- double equal to (==)
- triple equal to (===)
- not equal to (!=)
- not equal to (!==)

function

- block of code which can be reused
- types:
  - empty function:
  - parameterless function
  - parameterized function

higher order function (HOF)

- Map

```
    const numbers=[2,5,1,8,9]

      // to multipy all numbers by 2

      // const multiply=numbers.map( function (elem)  { return  elem * 2}
)
      // const multiply=numbers.map( (elem) => {elem * 2} )
      const multiply=numbers.map( elem => elem * 2 )
      console.log("🚀 ~ file: day16_1.js ~ line 8 ~ multiply", multiply)
```

- Filter

```
   const numbers=[2,5,1,8,9]
   const greterThanFive= numbers.filter(ele => ele > 5 )
   console.log("🚀 ~ file: day16_1.js ~ line 14 ~ greterThanFive",
greterThanFive)
```

- Reduce

```
   // if we want get summ of all elements of array
   let initial=0;
   const sum= numbers.reduce((total, ele)=> total+ele, initial )
 console.log("🚀 ~ file: day16_1.js ~ line 18 ~ sum", sum)
```

Objects

- objects in javascript are property value pair
- e.g.:

```
    const car = {
                  name : "i20" ,
                  color: "gray" ,
                  price: 7.2
               }
```