Assigment for day 2

Q1 What is lexical structure

Ans A programming language’s lexical structure specifies a set of some basic rules about how code should be written in it. Rules like what variable names look like, the delimiter characters for comments, and how one program statement is separated from the next. It is the lowest-level syntax of a language. for example text, comments, literals, reserved, seimicolons, unicode.

Q2 What is unicode?

Ans JavaScript is written in Unicode (It is an industry standard for the consistent encoding of written text). This means you can use Emojis as variable names, but more importantly, you can write identifiers in any language, for example, Japanese or Chinese, with some rules.

Q3 Explain all the keywords pressent in the javascript with examples.

Ans Function keyword used to define a function to execute a block of code.

Example

```
var func = function(){
return "Hello";
}
alert(func());
```

class keyword used to define a class.

Example

```
public class Employee {
public String efn = "Joseph";
public String eln = "Doe";
}
```

Return keyword:-Used to return from the function or method with or without a value.

Example

```
var func = function(){
return "Hello";
}
```

Eval keyword:-Used to evaluate a specified string. The eval use as a global function eval().

Example

```
function fun( ) {

var str1=2;

var str1=3;

var res = eval(new String(str1 + str2));

document.write(res);

}

fun();
```

For keyword:-Used to define a loop, for loop to repeatedly execute a block of code until a condition true.

Example

```
for(var a=0; a<=10; a++) {

document.write("The loop is running for " + a + " times");

}
```

If keyword:-Used to define a conditioned construct. if the statement is used to run a block of code if the condition is true.

Example

```
var date = new Date();

var day = date.getDay(); // Sunday    Saturday : 0    6

if(day==5) {

alert("This is weekend!");

} else {

alert("This is non-weekend!");
```

Break keyword:-used into a loop to break or stop the execution of the loop.

Example

```
for(var a=0; a<=10; a++) {
if(a == 5)
break;
document.write("The loop is running for " + a + " times");
}
```

Continue keyword:-Used into a loop to continue the loop and skip the following statements inside the loop.

Example

```
for(var a=0; a<=10; a++) {
if(a == 5)
continue;
document.write("The loop is running for " + a + " times");
}
```

null keyword:-Used to represent a special data type no value.

Example

```
var age = null;
alert(age);
```

New keyword:-Used to create an object.

Example

```
Employee obj = new Employee ();
```

while keyword:-Used for while loop, while loop executes the block of code until the condition is true.

Example

```
var a=1;
while(a <= 10)
{
```

```
document.write("loop is running for " + a + "times</p>");

a++;

}
```

super keyword:-Used to call function or method of a parent class.

Example

```
uper.disp(); //the disp is a method of a parent class
```

default keyword:-Used in a switch expression to specify the actions to be performed if no case

Example

```
var date = new Date();

switch(date.getDay()) {

case 6:

alert("This is weekend.");

break;

case 0:

alert("This is weekend.");

default:

alert("Looking for a weekend.");

break;

}
```

typeof keyword:-Which used to return the data type of an operand.

Example

```
typeof(“hello”) // output as string
```

Var keyword:-Used to declare a variable,

Example

```
var    fruits = ["apple", "banana", "orange"];
var age=22;
```

Const keyword:-Used to define a constant variable and that cannot be farther reassigned.

Example

```
const age=22;
```

This keyword:-Used to refer to the current object.

Example

```
class Employee extends Person {

constructor(name, eid, salary) {

super(name);

}

get incsalary() {

return this.salary * 0.2;

}

}
```

let keyword:-Used to declare a variable limited to a scope of a block of code, unlike a variable declared by the var keyword.

Example

```
let var fruits = ["apple", "banana", "orange"];
```

Q4 What are the shorthand operators

Ans for example:-

```
let x = 1,

    y = 2;

x += y;      this is called shorthand operator

console.log(x);

// 3
```

other shorthand operator

x = x + y -> x += y

x = x - y -> x -= y

x = x * y -> x *= y

x = x / y -> x /= y

x = x % y -> x %= y

Q5 What is "use strict" in javascript

Ans "use strict"; Defines that JavaScript code should be executed in "strict mode".You can use strict mode in all your programs. It helps you to write cleaner code, like preventing you from using undeclared variables.