Conditions and Loops

Assignment Questions

Assignment Questions

Q1. What are conditional statements? Explain conditional statements with syntax and examples.

Ans Conditional statements control behavior in JavaScript and determine whether or not pieces of code can run.

There are multiple different types of conditionals in JavaScript including:

"If" statements: where if a condition is true it is used to specify execution for a block of code.

"Else" statements: where if the same condition is false it specifies the execution for a block of code.

"Else if" statements: this specifies a new test if the first condition is false.

Now that you have the basic JavaScript conditional statement definitions, let's show you examples of each

Q2. Write a program that grades st9dents based on their marks^
  8 If grFatFr than 90 thFn A GradE
  8 If bFtwFFn 70 and 90 thFn a B gradE
  8 If bFtwFFn 50 and 70 thFn a C gradE
  8 BFlow 50 thFn an F gradF

Ans

```javascript
const grade = 80

if(grade>=90){

    console.log("A")

}
else if(grade>70 && grade<90){

    console.log("B")

}
else if(grade>70 && grade<90){

    console.log("C")

}
else{

    console.log("F")

}


// output


B
```

Q3. What are loops, and what do we need them? Explain different types of loops with their syntax and examples.

ANS

Loops are handy, if you want to run the same code over and over again, each time with a different value.

Often this is the case when working with arrays:

JavaScript supports different kinds of loops:

- `for` - loops through a block of code a number of times
- `for/in` - loops through the properties of an object
- `for/of` - loops through the values of an iterable object
- `while` - loops through a block of code while a specified condition is true
- `do/while` - also loops through a block of code while a specified condition is true

Q4. Generate n9mbers between any 2 given n9mbers.
Ex

8   onst num1 = 10;
8   onst num2 = 25;
O9tp9t: 11, 12, 13, …., 25
Ans

```
for (number1 = 10; number1 <= 20 ; number1= number1+4) {

    if(number1>10){

        console.log(number1)

    }


}


// output


14
18
```

Q5. Use the while loop to print n9mbers from 1 to 25 in ascending and descending order.

Ans

```
let number = 1


while(number<=25){


    console.log(number)


    number++


}



//output


1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
```

Full Stack Web Development