

Rapport

Namn: Xiaorui Wang

Övning 5 randomDice() & print()

För att uppnå denna uppgift skapades två funktioner som heter "randomDice ()" och "print ()".

I "randomDice ()" först skapar man en variabel som heter "arr" och tilldelas en tom array till den, som håller det värde som kommer att returneras senare. Därefter skapas en för-loop och går över 1000 gånger för att få 1000 slumpmässiga tal mellan ett och sex. Alla dessa slumpmässiga tal trycks in i "arr" -arrayen. Slutligen returnerar "arr" och lagrar den.

```
for (let i = 0; i < 1000; i++) {  
    //get a random number between 1 and 6 and  
    //push the results into the array  
    arr.push(Math.floor(Math.random() * (7 - 1)) + 1);  
}  
return arr; //return array
```

I den andra funktionen "print ()" skapas en konstant variabel som heter "data" och håller "randomDice ()" -funktionen. Därefter använder man "reduce()" -metoden för att loop över "data" och lagra värde i en variabel som heter "countOccurance". Den "reduce()" metoden fungerar så här. Det tar två parametrar: den första är en funktion som har två argument som heter "total" respektive "item". Om "item" finns i "total", inkrement förekomsten av "total[item]" en gång. Eller om det här är första gången "total [item] inträffar markera det som 1. När allt är klart, returnera "total ". "Reduce ()" -metoden tar en "initial value" som är tömt här . Efteråt, returnera "countOccurance" och lagra den.

```
const countOccurance = data.reduce(function (total, item) {  
    if (item in total) { //if item is found in total  
        total[item]++; //increament and add one [item] to  
        total  
    } else { //first time finding [item]  
        total[item] = 1; //mark it as 1  
    }  
    return total; //after looping 1000 times, return and  
    store total  
}, {}); //initial value of data.reduce() is empty  
return countOccurance; //return and store the end value
```

Utanför funktioner ska man visa resultat via console.table() och uppdatera det ett par gånger för att testa och se olika resultat.

(index)	Value
1	168
2	165
3	171
4	162
5	171
6	163

► Object

(index)	Value
1	161
2	166
3	174
4	156
5	180
6	163

► Object

Övning 6 factorialize()

Först skapa en konstant variabel "aNum" som tilldelas en "prompt ()" metod som kommer att be användaren att mata in ett slumpstal mellan 1 och 1000.

Då i en funktion som heter "factorialize()" som tar en parameter "num" och i kod block skriver man flera if/else satsningar att köras för att utvärdera ett resultat. Om input numret är mindre än 0 vilket betyder ett negativt tal returnerar satsningen -1. Om input numret är 0 returnerar satsningen 1 istället. Eller om input numret ligger utanför intervallet, vilket betyder större än 1000, kommer en alert() att "fired up" och varnar användaren "Invalid". Slutligen utförs en rekursion om ett giltigt intervallnummer har givits och utvärderats. Utanför funktionen anropar "factorialize (aNum)" för att aktivera resultatet.

```
function factorialize(num) {
  if (num < 0) return -1;
  else if (num === 0) return 1;
  else if (num > 1000) alert('Invalid');
  else {
    return (num * factorialize(num - 1)); //recursion to
    calculate the factorial num based on the given number
    aNum from prompt()
  }
}
factorialize(aNum); // call fn
```

Resultatet från funktionen "factorialize" tilldelas variabeln "text" som ett värde och visas i "document" markerat som id "text". Se skärmdumpen nedan:

This page says

Ange ett nummer mellan 1 och 1000

Cancel

OK

This page says

Ange ett nummer mellan 1 och 1000

Cancel

OK

720