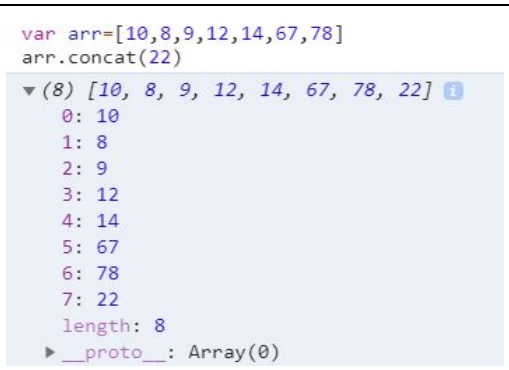
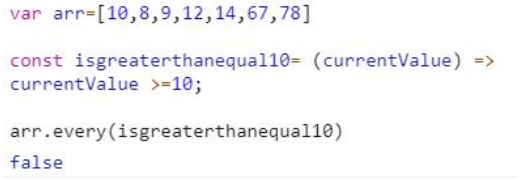
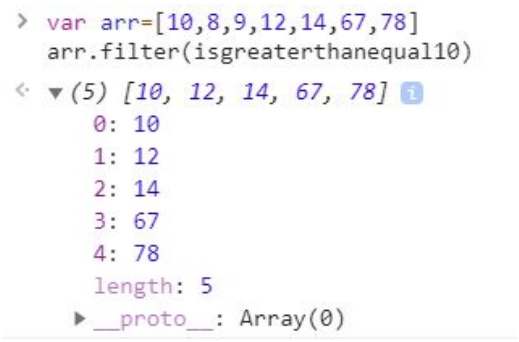
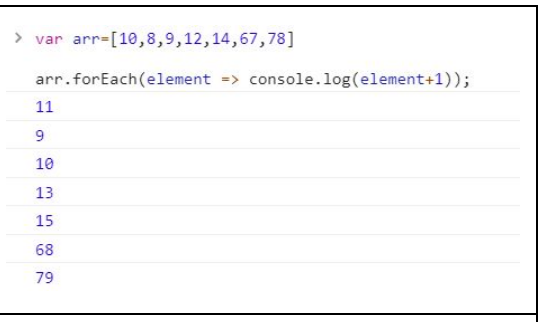
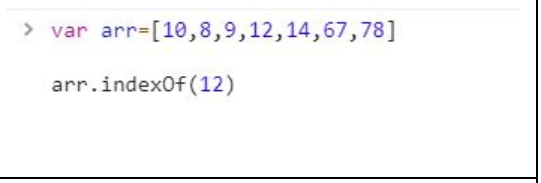
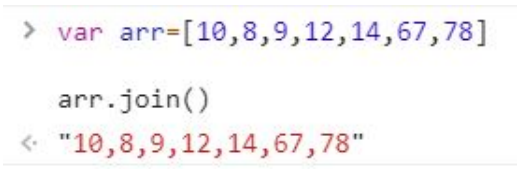
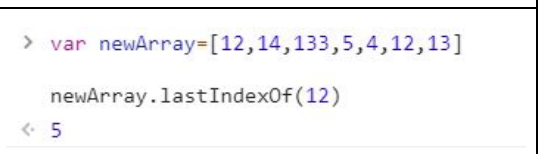
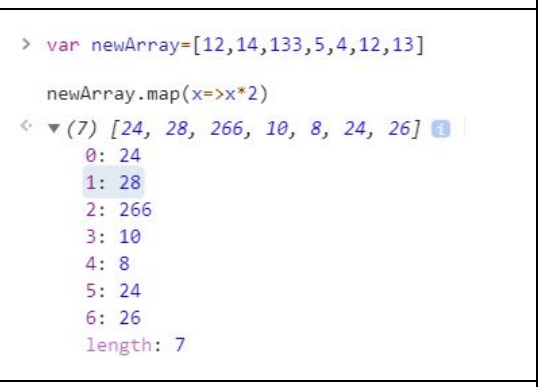
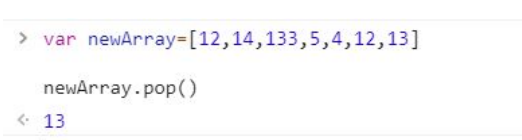
JAVASCRIPT ASSIGNMENT (Trainer: Rahul Pareek)

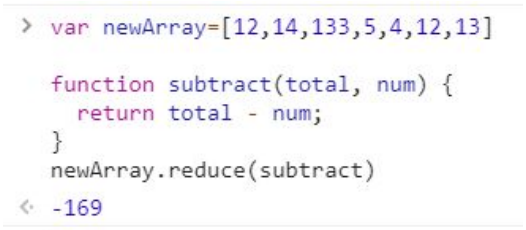
- **Submitted by (Kavita Goodwani INT 438)**

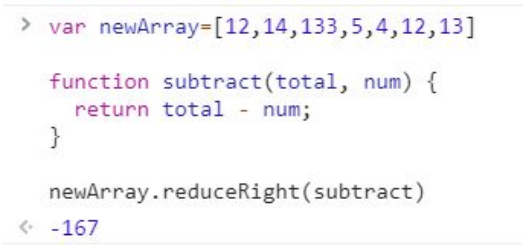
JAVASCRIPT ASSIGNMENT

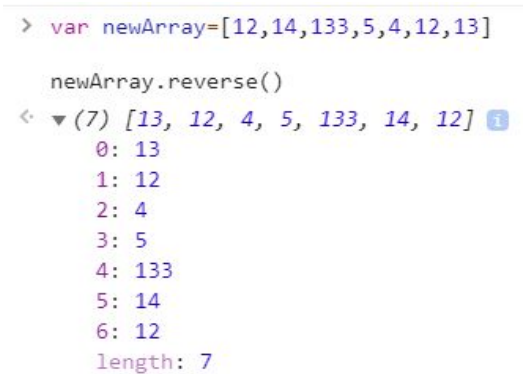
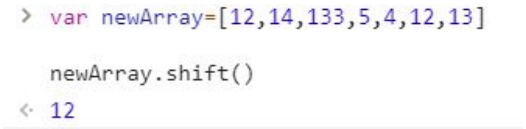
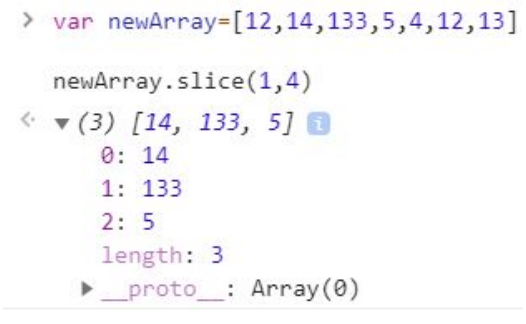
Q1. Explain Array Functions with examples.

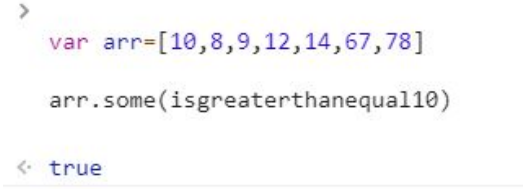
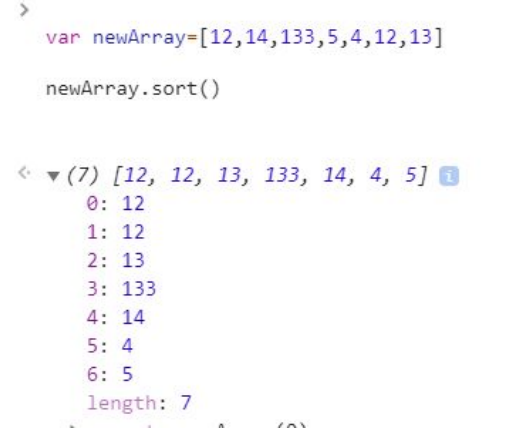
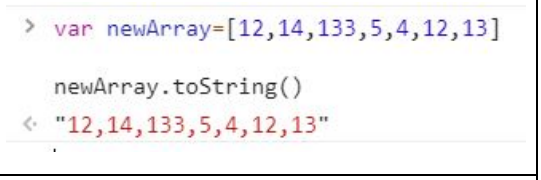
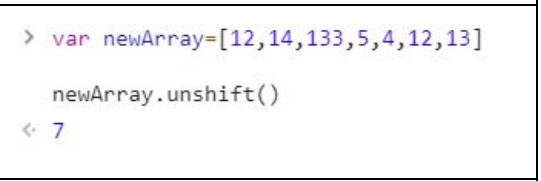
Function name	Command	Output
concat()	var arr=[10,8,9,12,14,67,78] arr.concat(22)	
every()	var arr=[10,8,9,12,14,67,78] const isgreaterthanequal10= (currentValue) => currentValue >=10; arr.every(isgreaterthanequal10)	
filter()	var arr=[10,8,9,12,14,67,78] const isgreaterthanequal10= (currentValue) => currentValue >=10; arr.filter(isgreaterthanequal10)	

forEach()	var arr=[10,8,9,12,14,67,78] arr.forEach(element => console.log(element+1));	
indexOf()	var arr=[10,8,9,12,14,67,78] arr.indexOf(12)	
join()	var arr=[10,8,9,12,14,67,78] arr.join()	
lastIndexOf	var newArray=[12,14,133,5,4,12,13] newArray.lastIndexOf(12)	
map()	var newArray=[12,14,133,5,4,12,13] newArray.map(x=>x*2)	
pop()	var newArray=[12,14,133,5,4,12,13] newArray.pop()	
push()	var newArray=[12,14,133,5,4,12,13] newArray.push(2)	

reduce()	var newArray=[12,14,133,5,4,12,13]
	
	function subtract(total, num) {
	  return total - num;
	}
	newArray.reduce(subtract)
		

reduceRight()	var newArray=[12,14,133,5,4,12,13]
	
	function subtract(total, num) {
	  return total - num;
	}
	
	newArray.reduceRight(subtract)
		

reverse()	var newArray=[12,14,133,5,4,12,13] newArray.reverse()	
shift()	var newArray=[12,14,133,5,4,12,13] newArray.shift()	
slice()	var newArray=[12,14,133,5,4,12,13] newArray.slice(1,4)	

some()	var arr=[10,8,9,12,14,67,78] const isgreaterthanequal10= (currentValue) => currentValue >=10; arr.some(isgreaterthanequal10)	
sort()	var newArray=[12,14,133,5,4,12,13] sort():newArray.sort()	
toString()	var newArray=[12,14,133,5,4,12,13] newArray.toString()	
unshift()	var newArray=[12,14,133,5,4,12,13] newArray.unshift()	

splice()	var month = ['Jan', 'March', 'April', 'June'];
	
	month.splice(1, 0, 'Feb');
	
	month
		

Q2. What will be the output of given code. Explain your answer.

```
var add = ( function(){ var counter=0; return function() { return counter+=1;}})() ;
add();
add();
add();
```

```
> var add = ( function(){ var counter=0; return function() { return counter+=1;}})() ;
  add();
  add();
  add();
< 3
> var add = ( function(){ var counter=0; return function() { return counter+=1;}})() ;
  add();
< 1
> add()
< 2
> add();
< 3
```

EXPLANATION:

The above is an example of self-invoking function in javascript, wherein the outer function returns the reference of outer function and the the initialisation is only called one time, therefore everytime add() is called it increments counter by one.

The self invoking function return as above by a developer is converted to a simple
ƒ () { return counter+=1;}
// an incrementer function

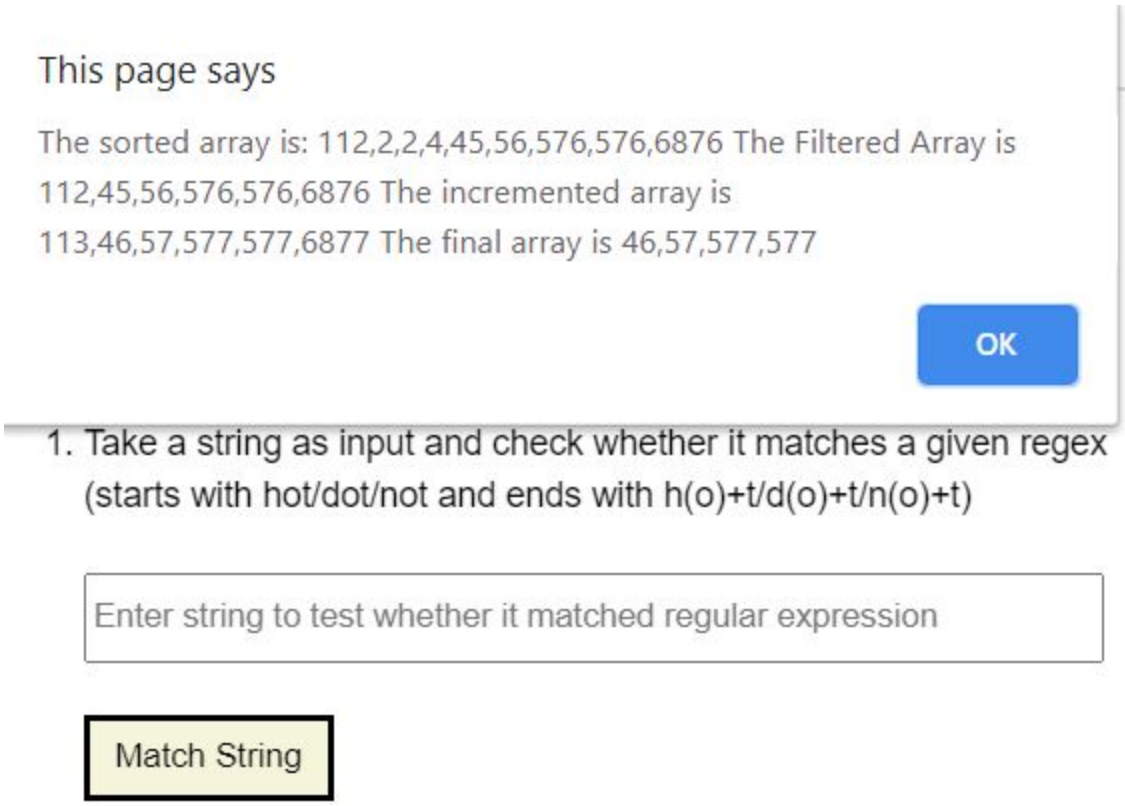
```
> console.log(add)
  ƒ () { return counter+=1;}
```

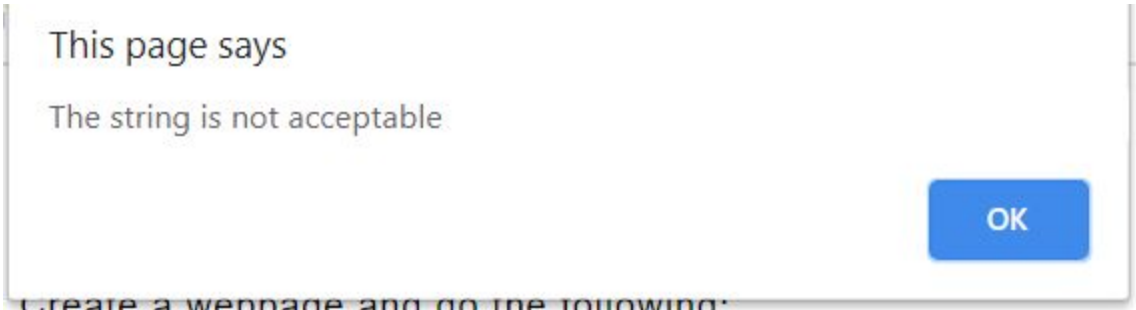
Q3. What is difference between /n and /r?

Ans /n refers to New line and /r is Carriage return : /n is used to point the cursor to the next line whereas /r is used to point to the start of the page i.e. to extreme left of page.

In Javascript we generally use only /n for next line.
Operating systems have different ideologies of next line and use both /n and /r together that is to place cursor to the extreme left of next line.

Q4. Create a webpage and do the following:
a. Take a string as input and check whether it matches a given regex (starts with hot/dot/not and ends with h(o)+t/d(o)+t/n(o)+t)
b. Take and array as input and do the following:
i. Sort the array
ii. Filter the array (Elements in array >=10)
iii. Add 1 to each element in the array
iv. Return Array elements from index 1 to 4





Create a webpage and do the following:

1. Take a string as input and check whether it matches a given regex (starts with hot/dot/not and ends with h(o)+t/d(o)+t/n(o)+t)

Hot is not good

Match String

2. Take and array as input and do the following:
   i. Sort the array
   ii. Filter the array (Elements in array >=10)
   iii. Add 1 to each element in the array
   iv. Return Array elements from index 1 to 4

Enter numbers separated by comma

Perform task