Q1: const numbers = [1, 5, 18, 2, 77, 108]; print the odd numbers. You're not allowed to use for, while, do...while, for..of, forEach loop.

```
function printOdd (...arr){

    arr.filter(num => num % 2 != 0)
    .map((item)=>console.log(item));
}
```

Q2: Create a function using function declaration named sum with one parameter of Array type, the returned result is the sum of all elements which are greater than 20.
sum([10, 20, 50, 30, 8]);

```
function sum (...arr){

 return arr.filter (num => num > 20)
.reduce ((num , currentValue) => num + currentValue, 0);

}
```

Q3: Create a function using function expression named getNewArray with one parameter of String Array, return a new array which contains all string, length is greater than and equal to 5, and contains letter 'a'.
getNewArray(["Hello", "Wonderful", "Happy", "People", "Have a great day"]);

```
getNewArray = function (...arr){

    return arr.filter (str => str.length >= 5 && str.includes('a'));

}
console.log(getNewArray("Hello", "Wonderful", "Happy", "People", "Have a great day"));
```

Q4:

```
var a = 2;
let b = 3;
function outer() {
    let c = 5;
    var d = 7;
    return function inner() {
        b = 8;
        let c = 9;

        console.log(a);
        console.log(b);
        console.log(c);
        console.log(d);
    }
}
outer()();
```

What will be the output?

Based on the code above,

1. What's the LE of global EC after creation phase finished before execution phase starts?

LE : { outer: null, a: undefined, Outer: fn} TZD: {b}

2. What's the LE of global EC after execution phase finished?

LE : { outer: null, a: 2, Outer: fn, b = 3}

3. What's the LE of function outer EC after creation phase finished before execution phase starts?

LE : { outer: GlobalEC, d: undefined, Inner: fn, arguments:{length: 0}} TZD: {c}

4. What's the LE of function outer EC after execution phase finished?

LE : { outer: GlobalEC, d: 7, Inner: fn, c = 9, arguments:{length: 0}}

5. What's the LE of function inner EC after creation phase finished before execution phase starts?

LE : { outer: OuterEC, arguments:{length: 0}} TZD: {c}

6. What's the LE of function inner EC after execution phase finished?

LE : { outer: OuterEC, arguments:{length: 0}, c: 9, b=8}

Ouput : A => 2 B => 8 C => 9 D => 7

Q5: When the HTML is

```
<body>
  <p style="color:red">First</p>
  <div class="central">
    <p class="item">Second</p>
    <ul>
      <li id="item">Third</li>
    </ul>
  </div>
</body>
```

And the CSS is:

```
body {
  background-color: yellow;
  color: blue;
}
p { color: orange; }
```

```
.central, .item {
  color: green;
}
#item {
  background-color: white;
}
ul{
 color: purple;
 background-color: beige;
}
```

Specify what colors will show on the screen for the:

	Background-color	color
First	**yellow**	**red**
Second	**yellow**	**green**
Third	**white**	**purple**

Q6: Write a regular expression that matches a string containing a date in the format mm/dd/yyyy.

^(0[1-9]|1[0-2])\/(0[1-9]|[12][0-9]|3[01])\/\d{4}$

Q7: Write a JavaScript function that takes an object with a firstName and lastName property, and returns a new object with a fullName property that combines the two names. Use a regular function to define the fullName property and the this keyword to refer to the firstName and lastName properties.

```
var myObject = {
    firstName : 'ivan',
    lastName : 'zimbe',
function fullName (myObject){

    return {

        fullName : this.firstName + " " + this.lastName,
    }
console.log(fullName.call(myObject, myObject));
```

Q8: Write a JavaScript function that takes an object with a firstName and lastName property, and a callback function that logs a message using the this keyword. Bind the callback function to the object using the bind method, and call the bound function with no arguments.

```
var newObject = {

    firstName : 'ivan',
```

```
    lastName : 'zimbe',
}
function logMessage (){

    console.log (this.firstName + " "+ this.lastName);

function main (object, callBack) {

    let bound = callBack.bind(object);
    bound();

main(newObject, logMessage);
```

Student ID: _____
Name: _____