



Pre Assessment Software Engineer - Back End Ultra Voucher

Terimakasih sudah tertarik menjadi bagian dari Team Software Engineer Ultra Voucher. Berikut adalah 2 problem yang kita harapkan kamu selesaikan maksimal 1x24 Jam, Solusi atas problem ini dapat *di-submit* melalui GitHub Gists dan dikirimkan tautannya kembali ke hrd@ultravoucher.co.id

Update terhadap result kamu akan di email maksimal 2x24 Jam oleh Ultra Voucher.

Good Luck & Happy coding!

1. Logic Test

Terdapat sebuah *array of strings* sebagai berikut:

```
['cook', 'save', 'taste', 'aves', 'vase', 'state', 'map']
```

Buatlah sebuah fungsi yang mengelompokkan sebuah *array of strings* di atas menjadi kumpulan anagram, dengan *expected result* sebagai berikut:

```
[  
  ['cook'],  
  ['save', 'aves', 'vase'],  
  ['taste', 'state'],  
  ['map']  
]
```

Catatan: dilarang menggunakan built in function seperti `array.map`, `array.reduce`, dan `array.filter`.

2. Query Test

Terdapat tabel sebagai berikut:

id	name	parent_id
1	Zaki	2
2	Ilham	<i>NULL</i>
3	Irwan	2
4	Arka	3

Buatlah query SQL yang menghasilkan data sebagai berikut:

id	name	parent_name
1	Zaki	Ilham
2	Ilham	<i>NULL</i>
3	Irwan	Ilham
4	Arka	Irwan

“Becoming the Leader Market in Voucher Industry”

Answer Pre Assessment

1. Logic Tes

```
const data = ['cook', 'save', 'taste', 'aves', 'vase', 'state', 'map']
let res = []
```

```
function sortString(word) {
```

```
    var sorted = word.split("");
```

```
    for (let i=0; i<sorted.length; i++){
        for (let j = 0; j < sorted.length; j++) {
            if (sorted[j] > sorted[i]) {
                temp = sorted[i];
                sorted[i] = sorted[j];
                sorted[j] = temp;
            }
        }
    }
}
```

```
    return sorted.join("")
}
```

```
for (let i=0; i<data.length; i++) {
    if (res.length == 0) {
        res[0] = [data[i]]
    }else{

        let same = false
        for (let x=0; x<res.length; x++){
            res[x][0]
            if(sortString(res[x][0]) == sortString(data[i])){
                res[x][res[x].length] = data[i]
                same = true
            }
        }
        if(!same) {
            res[res.length] = [data[i]]
        }
    }
}
```

```
// hasilnya
console.log(res)
```

Answer Pre Assessment

2. Query Tes

- Membuat table pengguna

```
CREATE TABLE pengguna (  
    id INT,  
    name VARCHAR(50),  
    parent_id INT  
);
```

```
INSERT INTO user (id, name, parent_id) VALUES  
(1, 'Zaki', 2),  
(1, 'Ilham', NULL),  
(1, 'Irwan', 2),  
(1, 'Arka', 3);
```

```
SELECT * FROM pengguna
```

Query Query History

```
1 CREATE TABLE pengguna (  
2     id INT,  
3     name VARCHAR(50),  
4     parent_id INT  
5  
6 );  
7  
8 INSERT INTO pengguna (id, name, parent_id) VALUES  
9 (1, 'Zaki', 2),  
10 (2, 'Ilham', NULL),  
11 (3, 'Irwan', 2),  
12 (4, 'Arka', 3);  
13  
14 SELECT * FROM pengguna
```

Data Output Messages Notifications

	id integer	name character varying (50)	parent_id integer
1	1	Zaki	2
2	2	Ilham	[null]
3	3	Irwan	2
4	4	Arka	3

Total rows: 4 of 4 Query complete 00:00:00.164

- Memasukan perintah untuk data yang diminta

```
SELECT t1.id, t1.name, t2.name AS parent_name  
FROM pengguna t1  
LEFT JOIN pengguna t2 ON t1.parent_id = t2.id;
```

Query		Query History	
16	SELECT	t1.id, t1.name, t2.name AS parent_name	
17	FROM	pengguna t1	
18	LEFT JOIN	pengguna t2 ON t1.parent_id = t2.id;	
Data Output		Messages	Notifications
<div><div><div><div></div><div></div><div></div></div><div><div></div><div></div><div></div></div><div><div></div><div></div><div></div></div><div><div></div><div></div><div></div></div><div><div></div><div></div><div></div></div><div><div></div><div></div><div></div></div><div><div></div><div></div><div></div></div><div><div></div><div></div><div></div></div></div></div>			
	id integer	name character varying (50)	parent_name character varying (50)
1	1	Zaki	Ilham
2	2	Ilham	[null]
3	3	Irwan	Ilham
4	4	Arka	Irwan