

# OneDegree Quiz

---

## Frontend

1. List

- (a) `function1` is private method. `function2` is public method.
- (b) memory inefficient

2. List

- (a) `[1, 10, 3]`
- (b) `aArray[0] = undefined; aArray[1] = undefined; aArray[2] = undefined;`
- (c) `const cArray = [...aArray]`

3. List

```
const mul = (a) => (b) => (c) => a * b * c;
```

Diff 

4. List

- (a) `10`
- (b) `1`
- (c) `{}`
- (d) `{}`
- (e) `[1, undefined, 3]`

5. List

For mounting a component, `componentDidMount` → `render`

For updating a component, `componentWillUpdate` → `render` → `componentDidUpdate`

For unmounting a component, `componentWillUnmount`

## 6. List

(a)

- The component may have performance issue (frequently re-render)
- The state is unused. The whole component can be rewrite with function component.

(b)

Option 1. Replace `Component` with `PureComponent`Option 2. Implement `shouldComponentUpdate`

```
shouldComponentUpdate(prevProps) { return prevProps.name !==
this.props.name; }
```

Diff ▾

(c) `componentDidMount`

## 7. PureComponent only do shallow comparison between previous props and new props.

## 8. List

(a) State management

(b) action, reducer (and optional action creator)

## 9. List

```
const reducer = (state = { city: 'Taipei' }, action) => { switch
(action.type) { case 'UPDATE': { return { ...state, city:
action.payload.city, }; } default: return state; } }
```

JavaScript ▾

## 10. Immutability in js means we cannot update variable by reference. Everytime we want to update a variable, we create a new reference.

It prevents us from accidentally modify the value of variable. Moreover, in react, PureComponent compares the reference to a variable, so immutability helps to re-render PureComponent and improve rendering performance.

## Backend

## 1. List

```
new_fruits = {} for k in fruits: new_fruits[fruits[k]] = k fruits = new_fruits
```

Python ▾

## 2. List

```
import math def is_prime(n): if n == 1: return False for factor in range(2, int(math.sqrt(n)) + 1): if n % factor == 0: return False return True
```

Python ▾

## 3. List

4. When client sends requests to different domain, port and protocol, server will set HTTP response header `Access-Control-Allow-Origin` and browser will check whether the response header is matched with client's domain.

## 5. List

## 6. List

```
def find_single(arr): result = arr[0] for i in range(1, len(arr)): result ^= arr[i] return result
```

Python ▾

## 7. List

```
SELECT Name FROM Customers OUTER JOIN Orders ON Customers.Id = Orders.CustomerId WHERE Orders.CustomerId == NULL
```

SQL ▾

## 8. List

```
def min_sum(arr_2d, depth=0, cursor=0): current_row = arr_2d[depth] if depth == len(arr_2d) - 1: return current_row[cursor] else: left_min_sum = min_sum(arr_2d, depth + 1, cursor) right_min_sum = min_sum(arr_2d, depth + 1, cursor + 1) if left_min_sum < right_min_sum: return current_row[cursor] + left_min_sum else: return current_row[cursor] + right_min_sum
```

Python ▾

## Bonus

```
A0 = {'a': 1, 'b': 2, 'c': 3, 'd': 4, 'e': 5} A1 = [0,1,2,3,4,5,6,7,8,9]
A2 = [1,2,3,4,5] A3 = [1,2,3,4,5] A4 = [1,2,3,4,5] A5 = { 0: 0, 1: 1, 2:
4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64, 9: 81, } A6 = [ [0, 0], [1,
1], [2, 4], [3, 9], [4, 16], [5, 25], [6, 36], [7, 49], [8, 64], [9,
81], ]
```

Python ▾