

**** All following exams please using Javascript only ****

1.

/**

There is an array, each item has such format:

```
{firstName: 'xxx', lastName: 'xxx', customerID: 'xxx', note: 'xxx',  
profession: 'xxx'}
```

lastName, **note** can be empty, **customerID** can only be a set of digital numbers.

profession can only have 'student', 'freelancer', 'productOwner', 'engineer' or 'systemAnalytics'.

**/

/**

Q1. Please follow the principle ('firstName' + 'lastName' + 'customerID') to sort this array and print it out.

**/

```
function sortUserName(user) {
```

```
}
```

/**

Q2. Please sort by 'profession' to follow the principle.

('systemAnalytics' > 'engineer' > 'productOwner' > 'freelancer' > 'student')

**/

```
function sortByType(user) {
```

```
}
```

2.

```
/** HTML
<div class="container">
  <div class="header">5/8 外出確認表</div>
    <div class="content">
      <ol class="shop-list">
        <li class="item">麵包</li>
        <li class="item">短袖衣服</li>
        <li class="item">飲用水</li>
        <li class="item">帳篷</li>
      </ol>
      <ul class="shop-list">
        <li class="item">暈車藥</li>
        <li class="item">感冒藥</li>
        <li class="item">丹木斯</li>
        <li class="item">咳嗽糖漿</li>
      </ul>
    </div>
  <div class="footer">以上僅共參考</div>
</div>
**/

/** CSS
.container {
  font-size: 14px;
}
.container .header {
  font-size: 18px;
}
.container .shop-list {
  list-style: none;
  margin-left: -15px;
}
.container .shop-list li.item {
  color: green;
}
.container .shop-list .item {
  /* Explain why does this color not works, and how to fix make it work on
1st list */
  color: blue;
}

/* Write styling make every other line give background color to next one */
**/
```

3.

/**

Please write down a function is used to create an array of unique values.

Example:

```
let items = [1, 1, 1, 5, 2, 3, 4, 3, 3, 3, 3, 3, 3, 7, 8, 5, 4, 9, 0, 1, 3, 2, 6, 7, 5, 4, 4, 7, 8, 8, 0, 1, 2, 3, 1];
```

output: [1, 5, 2, 3, 4, 7, 8, 9, 0, 6]

*/

```
function getUniqueNumber (items) {
```

```
}
```

4.

/ Can you explain about Interface and Enum, and where will you be using, please make some examples. */**

5.

/ Can you explain the problem with the following code, and how to fix it. */**

```
class Count extends React.Component {
  constructor(props) {
    super(props);
    this.state = { count: 0 };
    this.handleAddCount = this.handleAddCount.bind(this);
  }
```

```
  handleAddCount() {
    this.setState({ count: this.state.count + 1 });
    this.setState({ count: this.state.count + 1 });
    this.setState({ count: this.state.count + 1 });
  }
```

```

render() {
  return (
    <div>
      <h2>{this.state.count}</h2>
      <button onClick={this.handleAddCount}>Add</button>
    </div>
  );
}

ReactDOM.render(
  <Count />,
  document.getElementById('root')
);

```

6.

/ Please write the sample code to debounce handleChange (Do not use any 3th party libs other than react) **/**

```

const SearchBox = () => {

  handleChange = (event) => {
    // make ajax call
  };

  return <input type="search" name="p" onChange={handleChange} />;
};

```