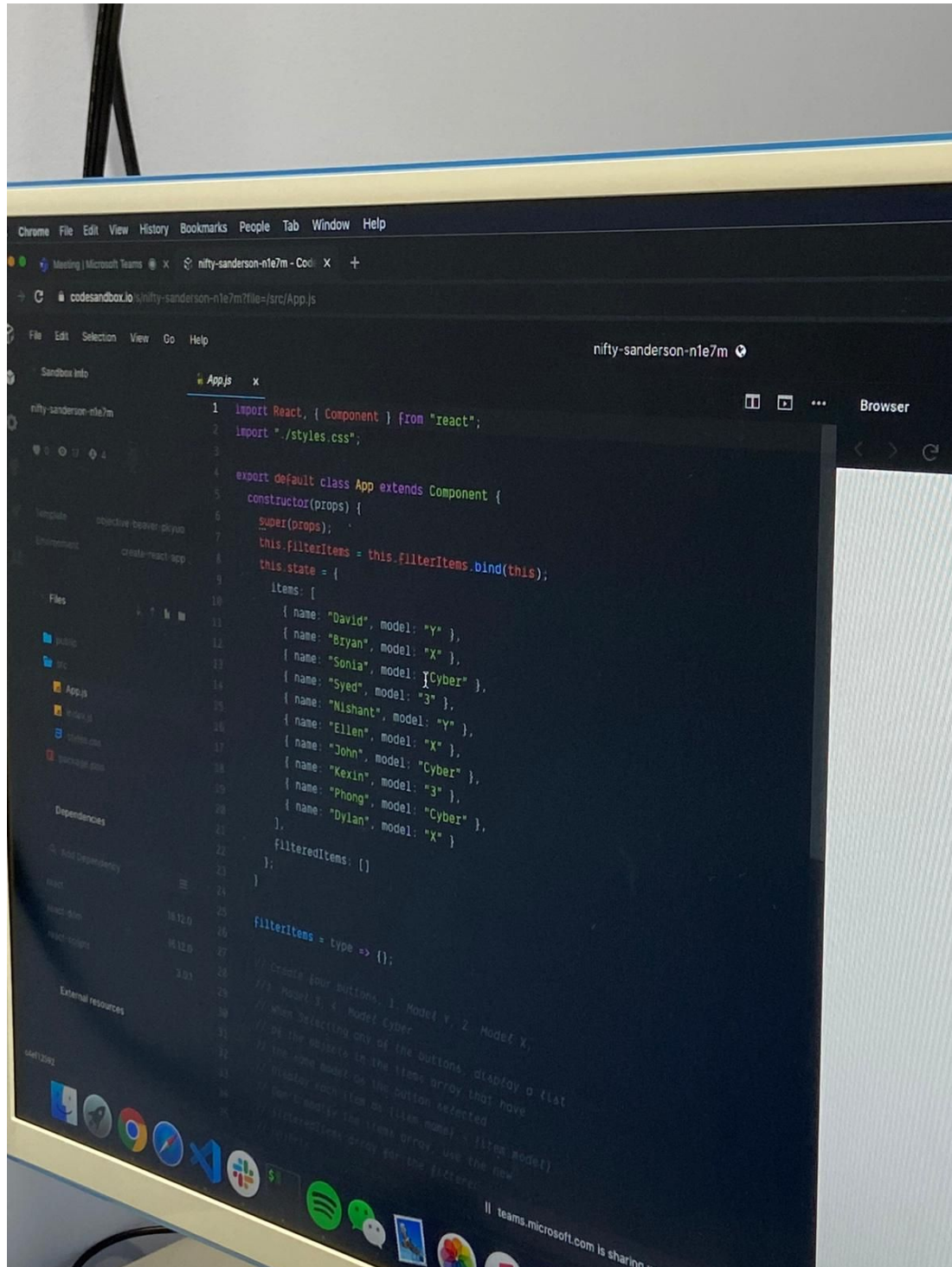


Its started

About self intro ---- interviewer is so new I feel (haha)

Coding questions started

What are those questions



Its a react code.

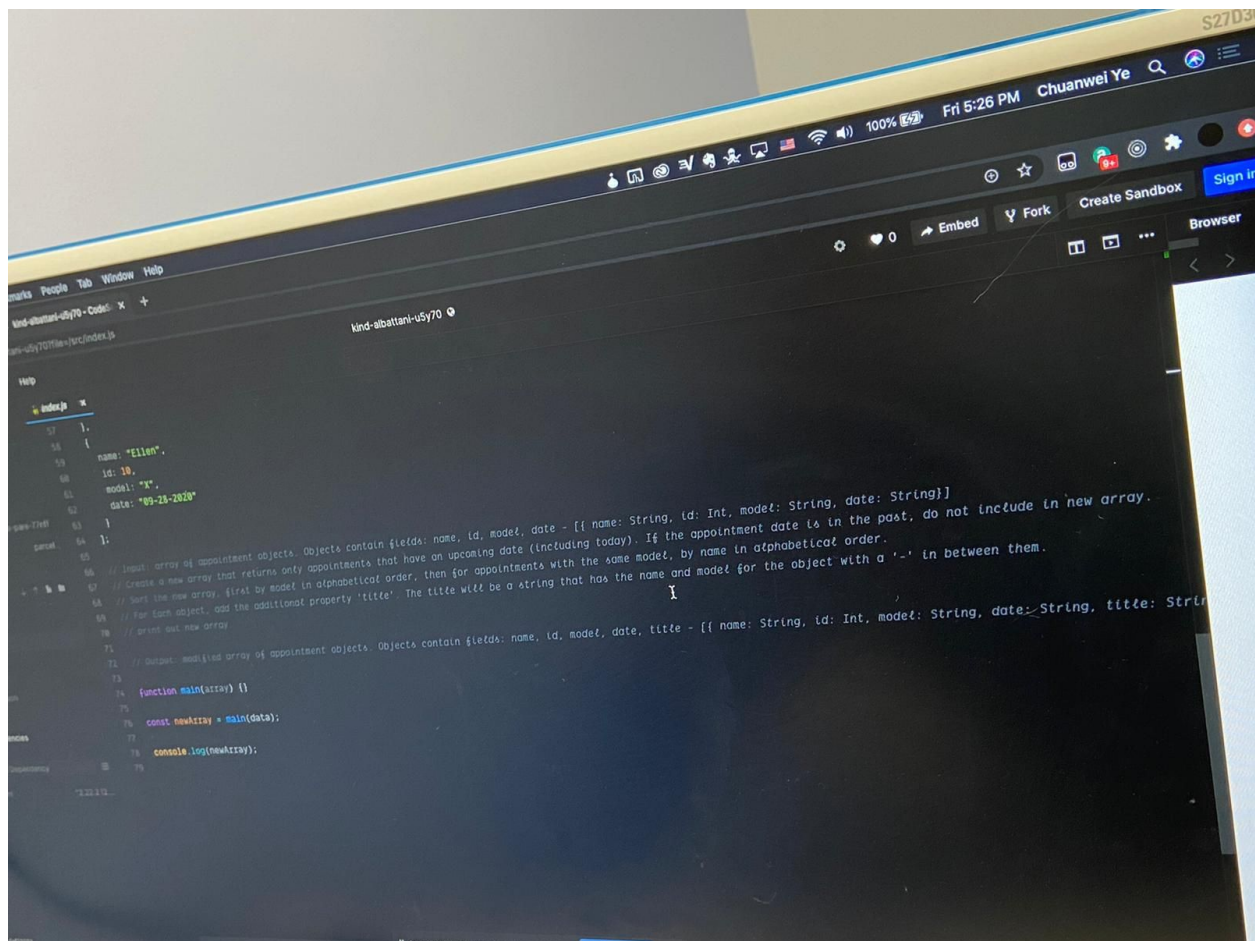
```
11     { name: "Bryan", model: "X" },
12     { name: "Sonia", model: "Cyber" },
13     { name: "Syed", model: "3" },
14     { name: "Nishant", model: "Y" },
15     { name: "Ellen", model: "X" },
16     { name: "John", model: "Cyber" },
17     { name: "Kexin", model: "3" },
18     { name: "Phong", model: "Cyber" },
19     { name: "Dylan", model: "X" }
20   ],
21   filteredItems: []
22 };
23 }
24
25 filterItems = type => {};
26
27 // Create four buttons, 1. Model Y, 2. Model X,
28 // 3. Model 3, 4. Model Cyber
29 // When Selecting any of the buttons, display a list
30 // of the objects in the items array that have
31 // the same model as the button selected.
32 // Display each item as {item.name} - {item.model}
33 // Don't modify the items array, use the new
34 // filteredItems array for the filtered list of
35 // objects
36
37 render() {
38   return <div>{`Code Here`}</div>;
39 }
40
41
```

Create four buttons
Give margin to the buttons give style to the buttons

Next question?

Give margin to the buttons give style to the buttons

Now second one java script algorithm I guess



Can you guys understand the question?

Interview done

Only two coding questions

Niceeeeeeeeeeeeeee

```
function main(array) {
```

```
  //const curDate = new Date();
```

```
  let newArray = array.filter((item) => {
```

```
    const curMoment = moment();
```

```
    const itemMoment = moment(item.date, 'DD/MM/YYYY');
```

```

    if (curMoment.diff(itemMoment) >= 0) {

        return true;

    }

    return false;

console.log(newArray)

// if (new Date(item.date).getTime() >= curDate.getTime()) {

//   return true;

// }else{

//   return false

// }

});

// it is right

// check the sort requiement

newArray.sort((itemOne, itemTwo) => {

    if (itemOne.model > itemTwo.model) {

        return 1;

    }

    else if (itemOne.model < itemTwo.model) {

        return -1;

    } else {

        Const item1Date = new Date(item1.date).getTime();

        Const item2Date = new Date(item2.date).getTime();

        if(item1Date > item2Date) {

```

```

        return 1;

    } else if (item1Date < item2Date) {

        Return -1

    } else {

        if(item1.name > item2.name) {

            Return 1;

        } else if(item1.name < item2.name) {

            Return -1

        } else {

            Return 0;

        }

    }

}

});

// test it save itt . check model in the object console!!!!!!

```

```

const result = newArray.map((item) => {

    return {

        ...item,

        title: item.name + '-' + item.model,

    };

});

```

```

return result;

}

```

```
class App extends React.Component {  
  
  modelSeleted(model) {  
  
    switch (model) {  
  
      case 'Y':  
  
        this.setState({  
  
          filteredItems: this.state.items.filter((item) => item.model === 'Y'),  
  
        });  
  
        break;  
  
      case 'X':  
  
        this.setState({  
  
          filteredItems: this.state.items.filter((item) => item.model === 'X'),  
  
        });  
  
        break;  
  
      case '3':  
  
        this.setState({  
  
          filteredItems: this.state.items.filter((item) => item.model === '3'),  
  
        });  
  
        break;  
  
      case 'Cybeer':  
  
        this.setState({  
  
          filteredItems: this.state.items.filter(  
  
            (item) => item.model === 'Cyber'  
  
          ),  
  
        },  
  
      ),  
  
    }  
  
  }  
  
}
```

```
    });  
  
    break;  
  
    default:  
  
        break;  
  
    }  
  
}
```

```
render() {  
  
    const content = this.state.filteredItems.map((item) => (  
  
        <tr>  
  
            <td>  
  
                { ' '  
  
                {item.name} - {item.model}  
  
            </td>  
  
        </tr>  
  
    ));  
  
    return (  
  
        <section>  
  
            <div className="btn-container">  
  
                <button className="btn" onClick={() => this.modelSeleted('Y')}>  
  
                    Model Y  
  
                </button>  
  
                <button className="btn" onClick={() => this.modelSeleted('X')}>  
  
                    Model X
```

</button>

<button className="btn" onClick={() => this.modelSeleted('3')}>

Model 3

</button>

<button className="btn" onClick={() => this.handleClick('Cybeer')}>

Model Cybeer

</button>

</div>

<table>{content}</table>

</section>

);

}

}