React Concepts

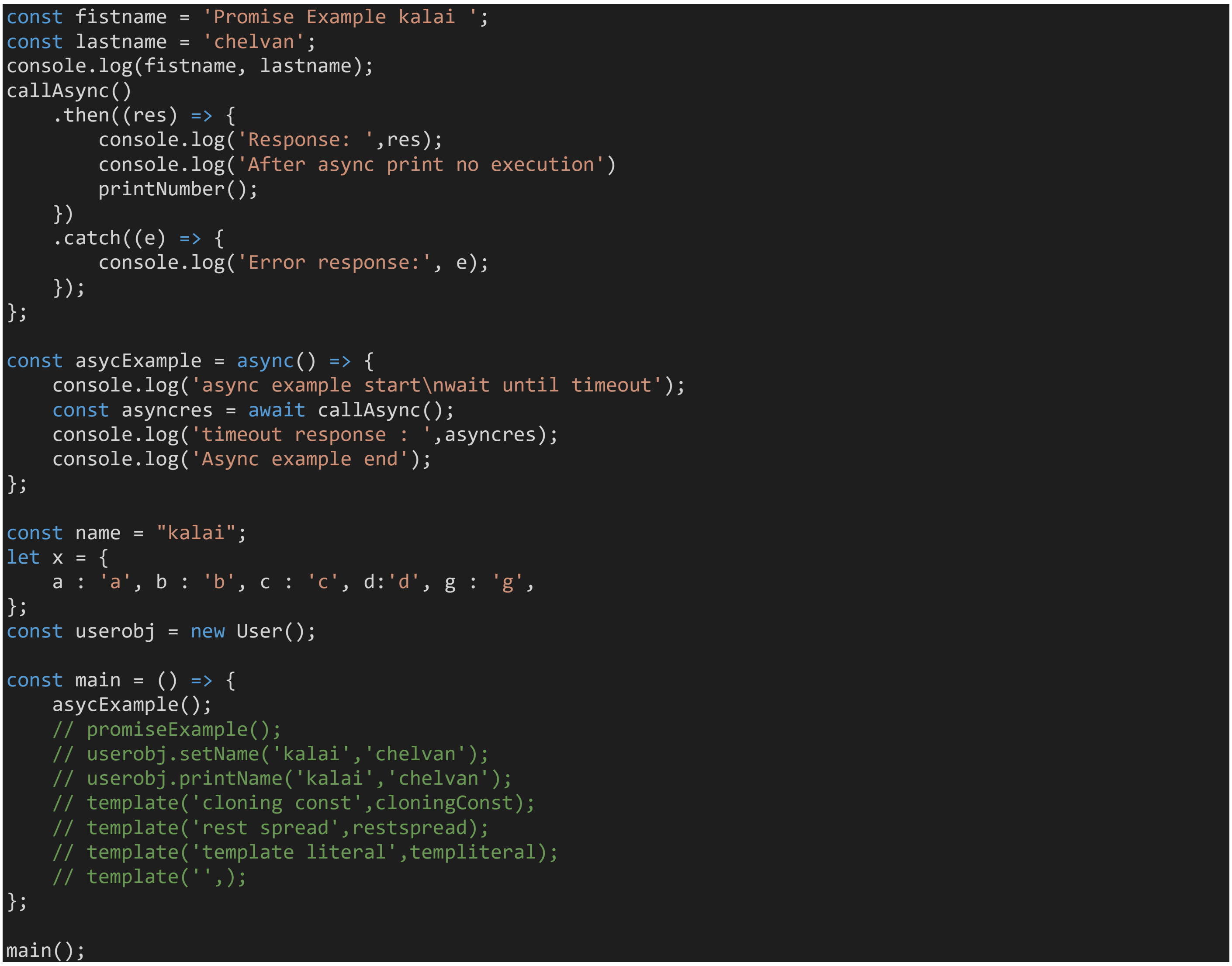
Spring AU’21

Kalai Chelvan

kalai.chelvan@accolitedigital.com

Codes:

```
const template = (name, fn) => {
    console.log(`********${name}**********`);
    fn();
    console.log(`*************************`);
};

const templiteral = () => {
    console.log('template listerals');
    console.log(`Hai ${name}`);
};

const restspread = () => {
    console.log('rest spread');
    console.log('x ->',x);
    const { a, g, ...remain } = x;
    console.log(a,g);
    console.log(remain);
    const t = { ...remain, g: 'g', h:'h' };
    console.log(t);
};

const cloningConst = () => {
    console.log("x->",x);
    const deepcopyjson = JSON.parse(JSON.stringify(x));
    const deepcopyObject = Object.assign({},x);
    const shallowCopyrest = {...x};
    shallowCopyrest.a = 'xx';
    console.log('shallowcopy altered after assigning a -> xx');
    console.log('deepcopyjson',deepcopyjson);
    console.log('deepcopyObject',deepcopyObject);
    console.log('shallowCopyrest',shallowCopyrest);
};

function User(){
    this.userData = { firstName : "", lastName : "" };
    this.logclass = () => {
        console.log(this);
    };
    this.setName = (fn, ln) => {
        this.userData.firstName = fn;
        this.userData.lastName = ln;
        this.logclass();
    };
    this.printName = () =>{
        console.log(`Name: ${this.userData.firstName} ${this.userData.lastName}`);
        this.logclass();
    };
}

const callAsync = () => {
    return new Promise((res, rej) => {
      setTimeout(() => {
        res('test');
      }, 1000);
    });
  };


const printNumber = () => {
    for (let i = 0; i < 5; i++) {
        console.log(i);
    }
};

const promiseExample = () => {
```



```
const fistname = 'Promise Example kalai ';
const lastname = 'chelvan';
console.log(fistname, lastname);
callAsync()
    .then((res) => {
        console.log('Response: ',res);
        console.log('After async print no execution')
        printNumber();
    })
    .catch((e) => {
        console.log('Error response:', e);
    });
};

const asycExample = async() => {
    console.log('async example start\nwait until timeout');
    const asyncres = await callAsync();
    console.log('timeout response : ',asyncres);
    console.log('Async example end');
};

const name = "kalai";
let x = {
    a : 'a', b : 'b', c : 'c', d:'d', g : 'g',
};
const userobj = new User();

const main = () => {
    asycExample();
    // promiseExample();
    // userobj.setName('kalai','chelvan');
    // userobj.printName('kalai','chelvan');
    // template('cloning const',cloningConst);
    // template('rest spread',restspread);
    // template('template literal',templiteral);
    // template('',);
};

main();
```