117-Using an Assertion Library [video]

- the assertion library we are going to be using is expectjs and you find it here https://github.com/mjackson/expect
- go to the terminal

```
joses-MacBook-Pro:notes mendoza$ meteor npm install expect@1.20.2 --save-dev
```

- **you should be** able to see this on you package.json under devDependencies
- lets import it in users.test.js

```
import expect from 'expect';
```

- now we use expect with the value we want to check not the final value, so in our code we would use the const res and using the .toBe() method that is where we use our final value, lets also delete the if statement below it

```
import expect from 'expect';


const add = (a, b)=> {
    if(typeof b !== 'number'){
        return a + a;
    }
    return a + b;
};


const square = (a) => a * a;


describe('add', function(){
    it('should add two numbers', function(){
        const res = add(11, 9);
        expect(res).toBe(20);
    });


    it('should double a single number', function(){
        const res = add(44);
        if(res !== 88){
            throw new Error('Number was not doubled');
```

```
        }
    });
});
```

- in our terminal lets start our test suite

```
joses-MacBook-Pro:notes mendoza$ npm test
```

- **now lets on purpose break our code by using the toBe value 21, that should give us an error**
- **and it does, this is the error that throws on the browser**

```
Error: Expected 20 to be 21
```

- go back to you code and change it back to 20
- now lets add a expect() to the it() below

```
import expect from 'expect';

const add = (a, b)=> {
    if(typeof b !== 'number'){
        return a + a;
    }
    return a + b;
};

const square = (a) => a * a;

describe('add', function(){
    it('should add two numbers', function(){
        const res = add(11, 9);
        expect(res).toBe(20);
    });

    it('should double a single number', function(){
        const res = add(44);

        expect(res).toBe(88);
    });
```

```
});
```

- now change the last one so they all have the expect function

```
import expect from 'expect';

const add = (a, b)=> {
    if(typeof b !== 'number'){
        return a + a;
    }
    return a + b;
};

const square = (a) => a * a;

describe('add', function(){
    it('should add two numbers', function(){
        const res = add(11, 9);
        expect(res).toBe(20);
    });

    it('should double a single number', function(){
        const res = add(44);

        expect(res).toBe(88);
    });
});

describe('test', function(){
    it('should square a number', function(){
        const res = square(9);

        expect(res).toBe(81);
    });
});
```