133-Rendering Notes video

- create a new component under imports/ui called NoteListItem.js
- import react
- this is going to be a default export stateless functional component

```
import React from 'react'


export default (props) => {

    return (



    );

};
```

- we are going to pass props in our title, if there is no title we will pass an Untitled string
- down in the paragraph we are going to have our time stamp and we will be using moment for this, and we are going to be using the updatedAt from note.js file , we then are going to format() the way we want to

```
import React from 'react';

import moment from 'moment';



export default (props) => {

    return (

        <div>

            <h5>{ props.note.title || 'Untitled note' }</h5>

            <p>{ moment(props.note.updatedAt).format('M/DD/YY')}</p>

        </div>

    );

};
```

- so now are going to do prop types but we need to change out the current stateless component to something different

```
import React from 'react';

import moment from 'moment';
```

```
const NoteListItem = (props) => {

    return (

        <div>

            <h5>{ props.note.title || 'Untitled note' }</h5>

            <p>{ moment(props.note.updatedAt).format('M/DD/YY')}</p>

        </div>

    );

};


export default NoteListItem;
```

- **the only** reason we are doing this is to setup our propTypes

```
import React from 'react';

import moment from 'moment';

import PropTypes from 'prop-types';


const NoteListItem = (props) => {

    return (

        <div>

            <h5>{ props.note.title || 'Untitled note' }</h5>

            <p>{ moment(props.note.updatedAt).format('M/DD/YY')}</p>

        </div>

    );

};


NoteListItem.propTypes = {

    note: PropTypes.object.isRequired

};


export default NoteListItem;
```

- now go to NoteList.js and make an instance of this new component <NoteListItem/> and pass the appropriate props

- first we need to use map() to go through the array of notes and then we return each indivual note with it's key and the note prop

```javascript
import React from 'react';

import { Meteor } from 'meteor/meteor';

import { createContainer } from 'meteor/react-meteor-data';

import PropTypes from 'prop-types';

import NoteListItem from './NoteListItem';


import { Notes } from '../api/notes'

import NoteListHeader from './NoteListHeader';


export const NoteList = (props)=>{

    return (

        <div>

            NoteList { props.notes.length }

            <NoteListHeader/>

            {props.notes.map((note)=>{

                return <NoteListItem key={note._id} note={note}/>

            })}

        </div>

    );

};
```

- now we are going to start running some test cases, lets start with NoteListItem.js
- lets make a new file for the test cases in imports/ui and call it NoteListItem.test.js
- inside this file lets import what we need

```javascript
import React from 'react';

import expect from 'expect';

import  { mount } from 'enzyme';


import NoteListItem from './NoteListItem';
```

- we then want to make sure we are running in the client side

```
import React from 'react';

import expect from 'expect';

import { mount } from 'enzyme';

import { Meteor } from 'meteor/meteor';


import NoteListItem from './NoteListItem';


if(Meteor.isClient){


}
```

- **we then** are going to be describing NoteListItem

```
import React from 'react';

import expect from 'expect';

import { mount } from 'enzyme';

import { Meteor } from 'meteor/meteor';


import NoteListItem from './NoteListItem';


if(Meteor.isClient){

    describe('NoteListItem' function (){


    })
}
```

- then we are going to be defining two test cases, the first we are going to check for the title and time stamp
- for the title we can use an string, with time stamp we can use something from chrome tools
- go to chorme tools and type

```
new Date().getTime()

1510584394273
```

```
import React from 'react';
```

```
import expect from 'expect';

import  { mount } from 'enzyme';

import { Meteor } from 'meteor/meteor';


import NoteListItem from './NoteListItem';


if(Meteor.isClient){

    describe('NoteListItem', function (){

        it('should render title and time stamp', function (){



        });

    });

}
```

- and now we can set those two const

```
import React from 'react';

import expect from 'expect';

import  { mount } from 'enzyme';

import { Meteor } from 'meteor/meteor';


import NoteListItem from './NoteListItem';


if(Meteor.isClient){

    describe('NoteListItem', function (){

        it('should render title and time stamp', function (){

            const title = 'my title';

            const updateAt = 1510584394273;

        });

    });

}
```

- lets now continue and create our wrapper and set it equal to mount, and we are going to pass an object to our note prop of title and updateAt

```javascript
import React from 'react';

import expect from 'expect';

import  { mount } from 'enzyme';

import { Meteor } from 'meteor/meteor';



import NoteListItem from './NoteListItem';



if(Meteor.isClient){

    describe('NoteListItem', function (){

        it('should render title and time stamp', function (){

            const title = 'my title';

            const updateAt = 1510584394273;

            const wrapper = mount(<NoteListItem note={{title, updateAt}} />)

        });

    });

}
```

- now we can move on to the assertions, then we are going to expect the h5 text value the same as the const title, and the also with p
- the date that I sude to the wrapper.find('p') was the current date

```javascript
import React from 'react';

import expect from 'expect';

import  { mount } from 'enzyme';

import { Meteor } from 'meteor/meteor';



import NoteListItem from './NoteListItem';



if(Meteor.isClient){

    describe('NoteListItem', function (){

        it('should render title and time stamp', function (){

            const title = 'my title';

            const updateAt = 1510584394273;
```

```
            const wrapper = mount(<NoteListItem note={{title, updateAt}} />)


            expect(wrapper.find('h5').text()).toBe(title);

            expect(wrapper.find('p').text()).toBe('11/13/17');

        });

    });

}
```

- now go the terminal and switch to test mode by exiting the development mode

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

- **now lets do another test case for the title untitled**

```
import React from 'react';

import expect from 'expect';

import  { mount } from 'enzyme';

import { Meteor } from 'meteor/meteor';



import NoteListItem from './NoteListItem';



if(Meteor.isClient){

    describe('NoteListItem', function (){

        it('should render title and time stamp', function (){

            const title = 'my title';

            const updateAt = 1510584394273;

            const wrapper = mount(<NoteListItem note={{title, updateAt}} />)


            expect(wrapper.find('h5').text()).toBe(title);

            expect(wrapper.find('p').text()).toBe('11/13/17');

        });
        it('should set default title if no title set', function (){

            const title = '';

            const updateAt = 1510584394273;
```

```
        const wrapper = mount(<NoteListItem note={{title, updateAt}} />)


        expect(wrapper.find('h5').text()).toBe('Untitled note');

    });

  });

}
```

-