139-The Editor Component video

- create a new component under imports/ui and call it Editor.js
- lets import React and create a named export Editor as an ES6 class component

```
import React from 'react';


export class Editor extends React.Component{

    render(){

        return (

            <div>Editor</div>

        );

    }

}
```

- lets instintiate this new component onto Dashboard.js

```
import React from 'react';


import PrivateHeader from './PrivateHeader';

import NoteList from './NoteList';

import { Editor } from './Editor'




export default () => {

    return (

        <div>

            <PrivateHeader title="Dashboard"/>

            <div className="page-content">

                <NoteList/>

                <Editor/>

            </div>

        </div>

    );
}
```

- lets import createContainer and make this our named default export

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

export class Editor extends React.Component{
    render(){
        return (
            <div>Editor</div>
        );
    }
}

export default createContainer(()=>{

})
```

- **and now in Dashboard.js lets change our named export to default export**

```
import React from 'react';

import PrivateHeader from './PrivateHeader';
import NoteList from './NoteList';
import  Editor  from './Editor'


export default () => {
    return (
        <div>
            <PrivateHeader title="Dashboard"/>
            <div className="page-content">
                <NoteList/>
                <Editor/>

            </div>

        </div>

    );

}
```

- back in Editor.js import Session, create a const selectedNoteId with the value of Session.get
- and that is going to return an object which will get the selectedNoteId and the note coming from Notes api, and we are going to find the selectedNoteId

```
import React from 'react';

import { Session } from 'meteor/session';

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

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


export class Editor extends React.Component{

    render(){

        return (

            <div>Editor</div>

        );

    }

}


export default createContainer(()=>{

    const selectedNoteId = Session.get('selectedNoteId');


    return {

        selectedNoteId,

        note: Notes.findOne(selectedNoteid)

    };

}, Editor)
```

- lets define two propTypes that will not be required

```
import React from 'react';
import { Session } from 'meteor/session';
import { createContainer } from 'meteor/react-meteor-data';
import { Notes } from '../api/notes';
import PropTypes from 'prop-types'

export class Editor extends React.Component{
    render(){
        return (
            <div>Editor</div>
        );
    }
};

Editor.propTypes = {
    note : PropTypes.object,
    selectedNoteId : PropTypes.string
}

export default createContainer(()=>{
    const selectedNoteId = Session.get('selectedNoteId');

    return {
        selectedNoteId,
        note: Notes.findOne(selectedNoteid)
    };
}, Editor)
```

- in Editor.js lets remove the return block and substitue with an if statement to check wheater there is a not or isn't

```
import React from 'react';
import { createContainer } from 'meteor/react-meteor-data';
import { Session } from 'meteor/session';
import { Notes } from '../api/notes';

import PropTypes from 'prop-types'


export class Editor extends React.Component{

    render(){

        if(this.props.note){

            return (

                <p>We got the note!</p>

            );

        }else if(this.props.selectedNoteId){

            return (

                <p>Note not found</p>

            );

        }else{

            return (

                <p>Pick or create a note to get started</p>

            );

        }

    }

};
```

- lets simplify our if statement, we are going to be using a turnary operator instead - along with one if statement

```
import React from 'react';

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

import { Session } from 'meteor/session';

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

import PropTypes from 'prop-types'


export class Editor extends React.Component{

    render(){
        if(this.props.note){
            return (
                <p>We got the note!</p>
            );
        }else{
            return (
                <p>
                    {this.props.selectedNoteId ? 'Note not found' : 'Pick or create a note to get started'}
                </p>
            );
        }
    }
};
```

-