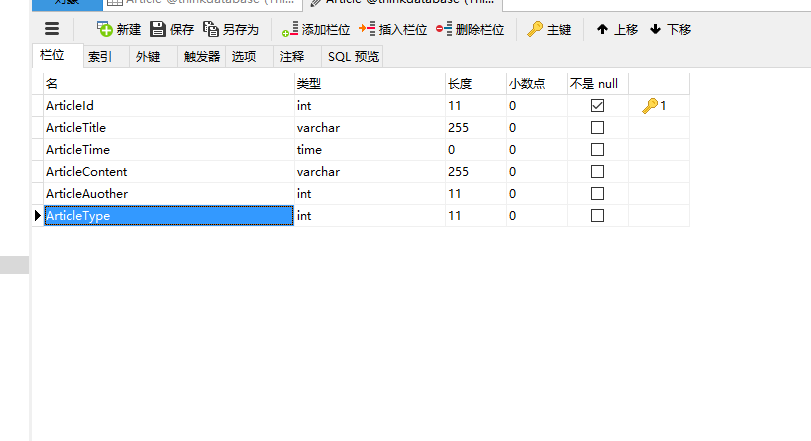
# react\_think

第一次尝试全栈

数据库部分：

​ 分为四张表，分别对应User(用户表), Article(文章表), Login(登录信息表), MenuType(导航栏分类，指定文章分类)

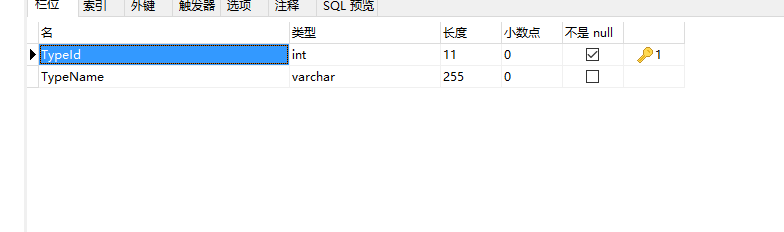
Article（文章表，设置了ID自增）



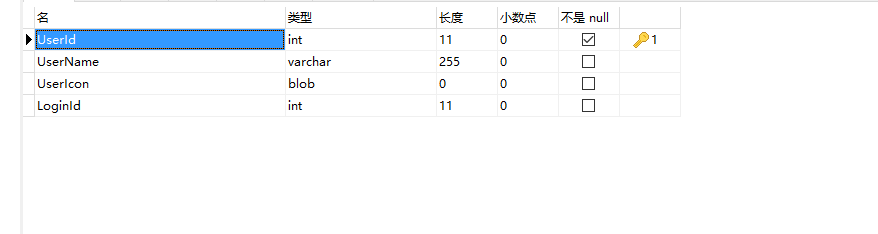
Login(登录信息表)



MenuType(文章分类表)



User(用户表)



后端部分：

技术栈： thinkjs

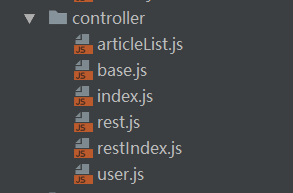
作用： 后端进行数据的转发，不进行视图层的控制，视图层交给react进行控制

数据库配置：



Controller层面：

创建了两个Controller，分别控制：用户登录（user.js）和文章增删减操作 （articleList.js）



前端部分：

​ 技术栈： react + react-router 4.0 + react-redux + antd-desgin

​ 跨域：使用proxy进行代理转发

工程目录：

1. ActionCreator：产生ActionCreator
2. ActionTypes： 分别Action对象
3. Component： 采用了Vue的项目架构，因为项目比较小，不用采用Flux 的架构
4. Details： 使用不同的component组成的Page
5. Img: 图片存放文件夹
6. Router： 设置了前端路由
7. Store： 对于react-redux的Store，使用了combineReducers进行合并， 进行处理（现在Reducer有问题，需要调试）

