conda activate xxx报错 --------- 直接activate xxx就行

".;C:\Prog ...

+ ~

引用运算符后面缺少属性名称。

所在位置 D:\Programs\anaconda3\shell\condabin\Conda.psm1:76 字符: 9

+ Invoke-Expression -Command $activateCommand;

安装模块时出现 check host-name 有可能是开了vpn

使用dcloud中的basic-table插件，row-click不能点击整行，一直找不到原因，去插件那里下载了示例demo，测试发现demo里面的正常，将demo中的basic-table.vue代码复制过来替换后正常了。

django.core.exceptions.ImproperlyConfigured: Cannot import 'evaluate'. Check that 'dvadmin.evaluate.apps.EvaluateConfig.name' is correct.

在子应用的app.py中要引用完整 例如 name = "dvadmin.evaluate"

直接使用optionList=res.data的形式视图不会更新

// optionList = [...res.data.map(item => ({...item, editing: false }))];

        // 上面的代码是一个错误示范，视图不会更新，需要使用Object.assign

        Object.assign(optionList, res.data)

使用slot-scope报错scope.row未定义 需要使用v-slot

<!-- <template slot-scope="scope"> -->

                <template v-slot="scope">

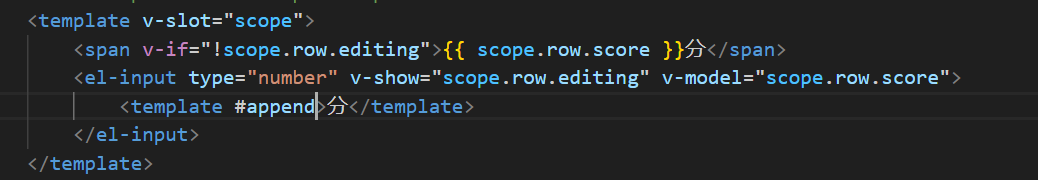
                    <span>{{ scope.row.text}}</span>

                    <el-input v-if="scope.row.editing" v-model="scope.row.text"></el-input>

                </template>

//this.$refs报错 需使用proxy.$refs

Template中需要将slot改成#



对于外键的查询

* select\_related用于处理外键（ForeignKey）关系，它会生成一个复杂的JOIN查询，在获取Scale对象时就把相关联的Question的外键数据一并获取，减少查询次数。
* prefetch\_related用于处理多对多（ManyToManyField）和一对多（ForeignKey）关系中的 “反向” 查询，它会在单独的查询中获取相关数据，然后在 Python 层面进行关联，避免了 N + 1 查询问题。

def get\_scale\_data(request, scale\_id):

    try:

        scale = Scale.objects.select\_related().prefetch\_related('questions\_\_options').get(id = scale\_id)

        data = []

        for question in scale.questions.all():

            question\_data = {

                'question\_text': question.text,

                'options': [option.text for option in question.options.all()]

            }

            data.append(question\_data)

        return JsonResponse(data, safe=False)

    except Scale.DoesNotExist:

        return JsonResponse({'error': 'Scale not found'}, status = 404)

****使用第三方库进行序列化****

* 1. Django 有很多第三方序列化库，如django - rest - framework(DRF)，它可以更方便和高效地处理数据序列化。

class ScaleDataAPIView(APIView):  
 def get(self, request, scale\_id):  
 try:  
 scale = Scale.objects.select\_related().prefetch\_related('questions\_\_options').get(id = scale\_id)  
 serializer = ScaleSerializer(scale)  
 return Response(serializer.data)  
 except Scale.DoesNotExist:  
 return Response({'error': 'Scale not found'}, status = 404)

el-radio中绑定的参数只能使用ref，用reactive点不动