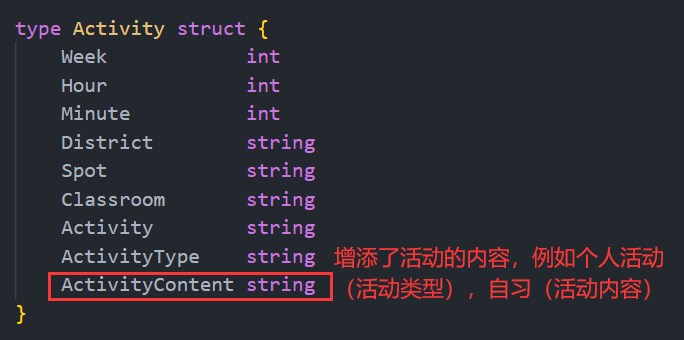
**数据结构的改动和需要增添的数据结构**

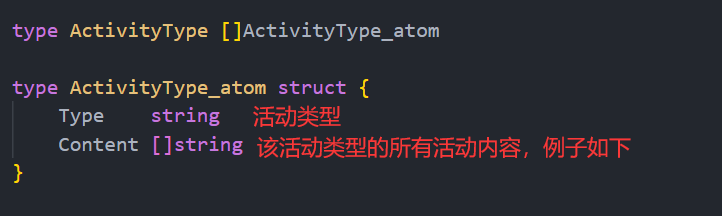
改动了一些数据结构：



需要实现的新数据结构：

保存所有活动类型和对应内容的结构体数组

构想：





**需要小修改的api**

/api/addClock

在后端将前端传来的对象压入clock数组后，需要对数组进行排序，时间小的放在数组头部。

/api/addActivity

在后端将前端传来的对象压入activity数组后，需要对数组进行排序，时间小的放在数组头部。

/api/uploadHomeworkFile

前端传来的是一个表单（FormData），后端需要接收该表单并进行一系列操作（具体操作在上次的文件里），尚未实现。

/api/uploadResourceFile

前端传来的是一个表单（FormData），后端需要接收该表单并进行一系列操作（具体操作在上次的文件里），尚未实现。

**需要实现的api**

/api/getActivityTypes

后端发送activityType数组给前端

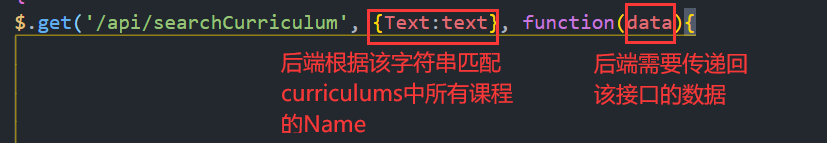
在这个接口以JSON形式呈现

-------------------------------------------------------

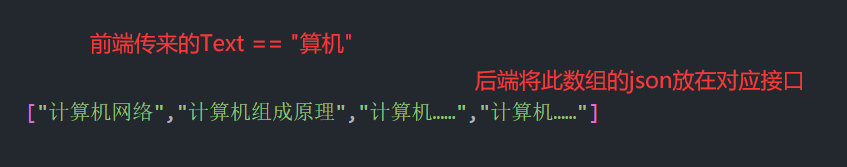
/api/searchCurriculum

前端发送给后端一个对象，该对象存储一个待搜索字符串。（前端保证待搜索字符串不为空）

格式为：



后端对curriculums数组中的所有课程名称Name进行字符串匹配，例如Text==”算机”，则匹配”计算机网络”、“计算机组成原理”等等，之后将匹配成功的Name放进新建的数组，进行字典排序（不排序也行）后将数组以json格式放在/api/searchCurriculum接口。



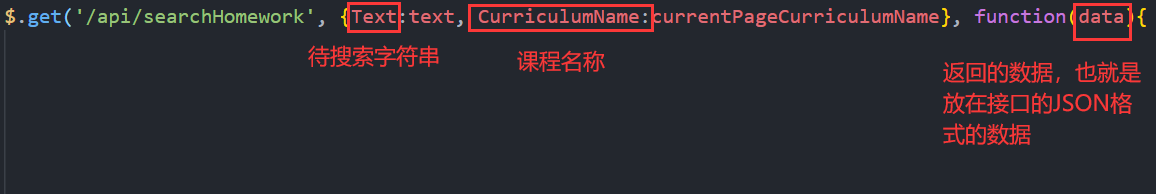
Ps：我觉得字符串匹配不用自己写，毕竟上学期写过了，老师之前好像说上学期写过的直接调用api就行了

-------------------------------------------------------

'/api/searchHomework'

前端发送给后端一个对象，该对象存储一个待搜索字符串和一个课程名称。（前端保证待搜索字符串不为空）

格式为：



后端根据CurriculumName找到curriculums数组中与Name匹配的科目信息，对该科目内的Homework数组中的所有元素的Title属性进行字符串匹配，匹配规则同上。之后将匹配成功的Homework中的元素放进新建的数组，针对时间对该数组进行排序，时间小的放在数组头部。之后将数组以json格式放在/api/searchHomework接口。



-------------------------------------------------------

/api/searchResource

前端发送给后端一个对象，该对象存储一个待搜索字符串和一个课程名称。（前端保证待搜索字符串不为空）

格式同上，为：



后端根据CurriculumName找到curriculums数组中与Name匹配的科目信息，对该科目内的Resource数组中的所有元素的ResourceName属性进行字符串匹配，匹配规则同上。之后将匹配成功的Resource中的元素放进新建的数组，针对时间对该数组进行排序，时间小的放在数组头部。之后将数组以json格式放在/api/searchResource接口。

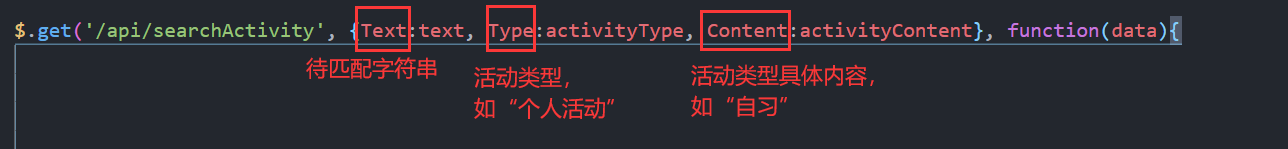


-------------------------------------------------------

/api/searchActivity

前端发送给后端一个对象，该对象存储一个待搜索字符串、一个活动类型、一个活动类型的具体内容。（字符串可能为空，后端需要判断当字符串为空时，不用匹配该字符串，将所有符合后两个要求的元素放进新建的数组）

格式如下：



后端在activity数组中找到所有活动类型相同（a.ActivityType == Type）、活动类型具体内容相同(a.ActivityContent == Content)、字符串成功匹配上(a.Activity与Text匹配)的元素，将这些元素放进新建的一个数组，针对时间对该数组进行排序，时间小的放在数组头部。之后将数组以json格式放在/api/searchActivity接口。

