

背单词页面的后台逻辑与数据库交互

以session中的列表为中心，数据库为辅助来判断输出的页面。

数据库设计

temp_book(word, word_book, username, degree, list)

注：用来放1天中，要背的单词，其中，list，为0，1，2。程度为0的，是立刻放入session，其他每次补充session的时候，会减1. degree：study, preview1, preview2, preview

0. 判断用户是否登录，是否存在user

1. 判断temp_book内单词数量（每天第一次登录时）（取一天数据）

判断用户的单词数量

- 该用户没有单词
 - 取单词，单词数量为设定的值；先取需要当天复习的单词，从大到小，依次取出，所有复习的单词取完（可以超出每天设定的学习单词的数量），抽出多少复习的单词，更改remember_word中的preview_number+抽出的数量；没填满，就取要学习的单词
 - 注：f系列设置temp_book中degree为preview，not learning设置为study
- 该用户有单词
 - 判断需要取单词的数量，然后执行上面的步骤，然后执行下面的步骤

2. 判断session中的列表是否为存在

- 不存在
 - 创建待读取的列表
- 存在
 - 执行下一步操作

注：session列表元素格式

{'单词': 'study/preview1/preview2/repeat'}

3. 判断session中列表是否存在元素（取一组数据）

- 存在，执行下一步操作
- 不存在，补充元素
 - 查找temp_book中有没有degree为preview的单词
 - 如果有，则全部取出，放入session列表中，并设置temp_book中degree为preview2，设置list为2，session列表设置为preview2，然后执行下一步操作
 - 如果没有，判断有无study的单词
 - 如果没有，则全部取出，放入session中
 - 如果有，则取出7个study的单词，再取出所有degree为0的单词，放入session中，按顺序，study在前
 - 执行下一步操作

4. 从列表中抽单词出来学习直到列表为空

- 顺序取出
- 为study：
 - 更新remember_word的study_number+1
 - 首先给出全英的单词，和例句
 - 如果熟悉，直接在列表中删除单词，并且在temp_book删除，并更新word_book为finished
 - 如果是模糊或者陌生，跳转到给出详细页面信息的页面，然后如果点击下一个，会有以下信息
 - 如果为陌生，放在列表最后面，然后更改为repeat，并且更新temp_book中单词的list值为1，degree为preview1；
 - 如果为模糊，放在列表最后面，然后更改为repeat，并且更新temp_book中单词的list值为2，degree为preview2
- 为preview1/preview2：
 - 首先给出测试的单词, 需要单词的拼写
 - 如果点击熟悉，直接在列表中删除单词，并且在temp_book删除，并更新word_book为高一级，oblivion的remember_number+1
 - 如果不成功拼写单词，则继续拼写
 - 如果成功拼写单词，则跳转到给出详细单词的页面，并且，让用户选择熟悉、模糊，陌生：
 - 熟悉
 - 如果原本是preview2，则在列表中删除单词，并且在temp_book删除，并更新word_book为高一级，如果为preview2,
 - 如果原本是preview1，则在列表中删除单词，在temp_book中更改为preview2，设置degree为2
 - 模糊
 - 不管原本是啥，放在列表最后面，然后更改为repeat，更新temp_book中list为1，degree为preview1. 并且，（如果word_book为degree为f系列，下降一个等级，最低为f1，oblivion的forget_number+1）
 - 陌生
 - 不管原本是啥，放在列表最后面，然后更改为repeat，更新temp_book中list为1，degree为preview1. 并且，（如果word_book为degree为f系列，直接更改为f1，oblivion的forget_number+1、remember_number+1）
- 为repeat
 - 首先给出测试的单词, 需要单词的拼写
 - 如果点击熟悉，直接在列表中删除单词
 - 如果点击模糊、陌生，继续放在列表后面