暑期学校实验项目：高考志愿填报助手

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 小组名称 | Django开发 | | | | | | |
| 姓 名 | 孙诚 | 专业 | 人工智能 | 班级 | AI 2班 | 学号 | 09118213 |
| 实验时间 | 2020.8.31-2020.9.23 | | 指导教师 | 孔祥龙 | | 成绩 |  |
| **一、实验背景和目的**  实验背景：2020年高考结束后许多高考生面临志愿填报的难题，志愿填报需要综合考虑大量的数据。  实验目的：为广大高考生提供一个数据齐全可靠的推荐系统，能够以“保一保，稳一稳，冲一冲”为原则根据考生的分数为其推荐合适的国内名牌大学。 | | | | | | | |
| **二、小组任务和个人任务**  小组任务：  ①完善后台管理系统，根据需求修改setting.py和url.py等配置文件  ②设计sqlite3数据库的存入和读取接口，将第一组的数据提供给其他组。  个人任务：  ①创建登陆页面与数据表格  ②与第八小组对接  ③与他人合作制作Majors类  ④数据修改  ⑤参与完成数据库的四次更新 | | | | | | | |
| **三、个人任务需求分析**  ①需求确定：为数据库补充Majors类数据并实现数据可视化，维护修补数据库内数据。  ②需求协商与验证：数据需要从第一组获取。 | | | | | | | |
| **四、实验过程（需附上关键代码及相关说明）**  将Majors信息录入数据库：  def getMajors(request):  df=pd.read\_csv('./extFiles/高校录取分数线整合（省份名字统一）.csv',encoding="gbk")  num=df.shape[0]  for i in range(num):  if df.at[i,'category']=='不分文理':  df.at[i,'category']='综合'  if df.at[i, 'College']=='华东理科大学':  df.at[i, 'College']='华东理工大学'  if df.at[i, 'College'] == '北京理科大学':  df.at[i, 'College'] = '北京理工大学'  if df.at[i, 'College'] == '\ufeff北京航空航天大学':  df.at[i, 'College'] = '北京航空航天大学'  c1=models.Majors(majorName=df.at[i,'Major'],  year=df.at[i,'Year'],  id=i,  categoryID=models.Category.objects.get(categoryname=df.at[i,'category']),  minScore=df.at[i,'score'],  collegeID=models.Colleges.objects.get(collegeName=df.at[i,'College']),  provinceID=models.Provinces.objects.get(provinceName = df.at[i,'Province']))  c1.save()  return HttpResponse("Majors in!")  该函数以dataframe格式从csv文件中读取大学往年录取线信息，并将其存入数据库。中间有很多if是为了兼容许多有瑕疵的数据。 | | | | | | | |
| **五、实验结果与分析**  实验结果：  ①个人的登陆页面与数据表格正常运作  ②与第八小组实现了良好高效的沟通交流  ③Majors信息被成功录入数据库  ④大量错误/缺失数据得到修改  分析：实验结果均符合预期 | | | | | | | | |
| **六、实验总结与心得体会**  实验总结：在这次实验中学会了许多Django的知识，复习了Python文件处理excel和json文件  的方法，对大型项目中的团队间合作以及团队内合作有了更加深刻的理解。  心得体会：  ①一个成熟的大项目往往需要很大的工作量，需要许多人长时间的共同努力。  ②众人拾柴火焰高。  ③数据的清洗是一项繁重却重要的任务。 | | | | | | | | |

2020年9月制