[CS209A-23Spring]作业 1(100 分)

题型设计:姚昭

代码样本& OJ: Yilun Qiu;测试用例:云翔燕

Git &代码风格:云翔燕

截稿时间:3 月 29 日下午
1155pm

宽限期:48 小时。如果你错过了截止日期，你仍然可以在 48 小时的宽限期内提交，即:，即 3 月 31 日下午 2355 分之前。然而，你会得到 40%的罚款(即:，你会得到你分数的 60%。超过宽限期的投稿将不被接受。

在线课程来自 edX 分析

在这个作业中，你将设计一个 OnlineCoursesAnalyzer 类，它可以从 edX 中读取在线课程数据集，并执行各种有用的分析。你将有机会使用技术，如 Collections，Lambda，Streams，我们在课程中已经讲过了。的 OnlineCoursesAnalyzer 类有一个构造函数从给定路径读取数据集文件。这个类还有其他 6 个执行数据分析的方法。中实现这些方法 OnlineCoursesAnalyzer 类。方法细节描述如下。0.读取数据集

```
public OnlineCoursesAnalyzer(String datasetPath)
```

的构造函数 OnlineCoursesAnalyzer 取数据集文件的路径并读取数据。数据集在 csv 格式，并具有以下列:

机构——在线课程持有者

课程编号-每门课程的唯一 id

启动日期——每门课程的启动日期

课程名称——每门课程的名称

讲师——每门课程的讲师

课程主题——每门课程的主题

Year -每门课程的最后一次

荣誉代码证书-有(1)，没有(0)。

参与者(课程内容已访问)-已访问被审计课程的参与者数量(> 50%的课程内容已访问)-已审核超过 50%课程的参与者数量

认证人数-总投票人数

%被审计-被审计公司的百分比

% Certified -已认证的百分比

认证的课程内容已访问超过 50%的百分比-认证的课程访问超过 50%的百分比

%播放视频-播放视频的百分比

论坛发帖百分比-在论坛发帖的百分比

课程总学时数(千)-课程总学时数(每 1000)

认证的中位学时-认证的中位小时

Median Age -参与者年龄的中位数

% Male——男性的百分比

% Female——女性的百分比

注意:对于每个问题，如果这个特定问题所需的数据单元格为空或格式错误，您可以简单地忽略这个特定问题的整行。1.参加人数按院校计算(10 分)

```
public Map<String， Integer>
```

此方法返回一个 <institution, count> Map，其中键是机构，而值是访问过该机构课程的参与者总数。的字母顺序对地图进行排序 institution。2.参加人数按院校和课程科目计算(10 分)

```
public Map<String， Integer>
```

此方法返回一个 <institution-course Subject, count> map，其中键是使用'-'连接机构和课程 Subject(不带引号)的字符串，而值是某个机构的课程 Subject 的参与者总数。

map 应该按的降序排序 count (即。，参与者从多到少)。如果两个参与者有相同的计数，那么它们应该按字母顺序排序

institution-course 主题。

3.讲师课程列表(20 分)

```
public Map<String, List<List<String>>> getCourseListOfInstructor()
```

一个讲师可以负责多个课程，包括独立负责的课程和共同开发的课程。

这个方法返回一个<Instructor， [[course1, course2，…]，[coursek,coursek+1，…]]>映射，其中键是讲师的名称(不带引号)，而值是包含 2 门课程列表的列表，其中 list 0是讲师独立负责的课程，如果他/她没有独立负责的课程，也需要创建这个列表，但不包含任何元素。List 1是讲师共同开发的课程，如果没有共同开发的课程，做的和 List 0一样。注意，课程名称(不带引号)应在列表中按字母顺序排序，人名相同的情况应按同一人处理。

4.顶级课程(20 分)

```
public List<String> getCourses(int topK, String by)
```

该方法返回前 K 门课程(参数 topK)通过给定的标准(参数 by)。具体地说，by="hours":结果应为课程按降序排序 Total Course Hours (Thousands) (从最长课程到最短课程)。

by="participants":结果应按课程数量的降序排序 Participants (Course Content Accessed) (从多到少)。

注意，结果应该是一个 Course 标题列表。如果两个课程的总课程时数或参与人数相同，那么它们应该按照标题的字母顺序进行排序。相同的课程名称只能在列表中出现一次。

5.搜索课程(20 分)

```
public List<String> searchCourses(String courseSubject, double
percentAudited, double totalCourseHours)
```

该方法基于三个标准搜索课程:

courseSubject:支持模糊匹配，不区分大小写。如果输入 courseSubject 是“science”，则所有科目包含“science”或“science”或其他(不区分大小写)的课程都符合标准。

percentAudited:被审定的百分比应为>= percentAudited

totalCourseHours:总课程时数(千)应<= totalCourseHours 请注意，结果应该是符合给定标准的课程标题列表，并按标题的字母顺序进行排序。相同的课程标题在列表中只能出现一次。6.推荐课程(20 分)

```
public List<String> recommendCourses(int age, int gender, int .
isBachelorOrHigher)
```

该方法根据以下输入参数推荐 10 门课程: `age`:用户的年龄

`gender`: 0-女，1-男

`isBachelorOrHigher`: 0-未获得本科学历，1-本科及以上学历先计算平均值 `Median Age`,平均 `% Male`，平均 `% Bachelor's Degree or Higher` 为每门课程。请注意，`Course Number` 是每道菜的唯一 id;其次是下面的公式:

$相似度值=(年龄-平均中位年龄)^2 +(性别 100 -平均男性)^2 + (isbachelorhigher100 -平均学士学位或更高)^2$

用于计算输入用户的特征与每门课程参与者特征之间的相似度。相似度越高，值越小;最后，返回相似度值最小的前 10 门课程。

注意，结果应该是一个课程名称列表。一个 `Course Number` 可能对应不同的 `course titles`，请将 `course title` 连同最新的 `Launch Date` 而相同的课程名称只能在列表中出现一次。课程应该按照它们的相似度值进行排序。如果两个课程具有相同的相似值，那么它们应该按照标题的字母顺序进行排序。

评价

代码正确性:我们在 OJ 系统上部署了自动测试用例来测试您代码的正确性。请提交 `OnlineCoursesAnalyzer.java` 敬 OJ。在提交给 OJ 之前，您可以使用我们的样本测试用例和样本测试数据在本地测试您的代码。

版本控制:你应该使用 `GitHub` 要管理项目的代码更改(有关如何使用的进一步详细信息，请参阅实验室 1 `git`)。你应该做至少 2 次提交(2 次提交是

推荐)。您在 GitHub 上的远程回购应该设置为 `private` 在 A1 截止日期之前，这样其他人就不会看到你的代码。

编码风格:你应该注意编写可读性和可维护性的代码。如何使用请参见实验 1 `CheckStyle` 为了这个目的。在 A1 的最后期限之后，您应该将您的 GitHub 回购设置为 `public`，因为我们会检查你的提交是否有减少 `CheckStyle` 根据的警告 `google_checks.xml`。

OJ Tips

阅读时请指定 UTF-8 编码 `.csv` 文件。

请不要在代码注释中包含任何汉字。请使用 `int` 对于整数和 `double` 对于浮点数。