

Scoops Ahoy — CITS1402项目

Gordon Royle

2019第二学期

1.0版（2019年10月6日）-初始发行

概述

该项目的目的是开发和测试您编写简单MySQL和高级MySQL的能力，而无需花费时间和精力，您可以使用计算机和MySQL文档来编写代码。

但是，这些问题并不意味着很难解决，也不是研究极其先进或复杂的MySQL。完全了解实验室和讲座资料的学生应该能够在指定的实验室时间内完成大部分项目。

这是一个单独的项目，因此您不能提交任何不是您自己开发的代码。您可以与朋友共同讨论该项目，也可以咨询实验室服务人员。您可能想知道在合理的讨论与不可接受的协作之间划清界限的原则—作为一种经验法则，您不应查看别人（书面）或在屏幕上（屏幕上）显示的任何内容。如果您在上帮助，`help1402`寻求请尝试缩小导致问题的查询或例程的确切部分，然后仅发布该部分。无论如何，请避免发布整个查询或存储的例程。

自动化工具会检查所有提交的代码是否相似。可能会要求学生口头解释其代码或复制其代码。在任何确凿的plagiarism案件中，双方都被视为对违规行为负有同等责任，并将受到同样的惩罚。（过去，由于从朋友那里复制而为该项目提供零分的学生，似乎对让朋友也得到零分这一事实感到最不高兴。）

请注意，您的实验室辅导员仅被允许提供一般指导，可帮助您识别语法错误，并指导您阅读讲义或其他文档。

可交付成果

我们将使用 `cssubmit` 再次提交项目，但提交将由多个小文件组成，而不是包含所有答案的一个文件。

您必须完全按照指定的方式命名文件，函数和过程-大多数代码将由自动脚本进行测试，因此，如果名称不正确，则脚本将无法找到并运行它们。

您将获得一个小的数据库测试版本，以测试您的代码，但是实际的

标记将在另一个更大的版本上执行。

Project Setting

Scoops Ahoy是一连串的冰淇淋商店，出售圆锥形的冰淇淋。他们的数据库管理器已跨州移动，但他们认为，由于仅使用了有限的查询范围，因此不必有新的永久性商品。数据库管理员。结果，您被雇用为顾问，为最常用的查询编写一些存储的例程。

您将获得一个文件该文件 `ScoopsAhoy.sql`，将构建并填充实际数据库的测试版本。它包含从2019年1月到2019年9月的9个月期间的销售数据。该数据库包含名称为 `Cone`，`cubesInPurchase`，`Customer`，`customerPurchases`，`Purchase`，`Scoop` 和 `scoopsInCone`的表。

您会被告知一些业务规则：

- 1.一次 购买 包含的一个或多个 锥体 在特定日期在三家商店之一购买。
2. 锥形 由1-3制成，分别位于杯勺威（可能不同）的口味化饼圆锥，糖圆锥或华夫饼圆锥上。
- 3.每次 购买 由一个做 客户 谁可能使用优惠券一定的百分比在价格减少。

为了使用存储的例程，了解以下命令来列出和删除它们很有用！

```
SHOW PROCEDURE状态 WHERE DB = 'ScoopsAhoy';
显示 功能状态 WHERE DB = 'ScoopsAhoy'; DROP
PROCEDURE <过程名称>; DROP FUNCTION <过程名称>;
```

创建和验证示例 ScoopsAhoy 数据库

使用MySQL Workbench，选择“运行SQL脚本 从”文件” 菜单中，然后导航到的位置 `CreateScoopsAhoy.sql` 计算机上文件，然后选择“打开”。

将打开第二个窗口，显示文件的前几行-只需选择“运行” 选项，就应该创建整个数据库。然后使用更改为该数据库 `USE ScoopsAhoy`；并使用MySQL的校验和功能验证数据库。

```
CHECKKSUM TABLE CONE; +
----- +
----- ++桌子校验和+
----- +
----- ++ scoopsahoy.cone
```

```
| 556540323 | +
----- +
----- ++
```

其他表的校验和为2520800458、3746423535、255292472、2615166047、205518241和1669916532表名的顺序）。

2

基本问题

第一组问题是基本SQL查询-对于每个问题，都应将查询写到具有指定文件名的单独文件中。

1.使用 `ERDplus.com` 准备一张ER图，描述数据库的当前结构。仅包括数据库中当前存在的实体，关系和属性。使用关于冰淇淋店如何在数据库中看不到的地方工作的常识，为每种关系包括适当的基数约束。

(a) 上传到 `ERDpluscssubmit` 的“导出图像”功能生成的图像文件，您应将其重命名为 `ScoopsAhoy.png`。(b) PNG或可移植网络图形是光栅图形图像的文件格式。(c) ERDPlus将使用某个默认名称保存文件（在Mac上为 `image.png`）

。2.编写一个SQL查询，列出在Scoops Ahoy进行的购买总数。

(a) 上传到 `Q2.sqlcssubmit` 将名称文件 仅 包含查询的。(b) 对于提供的样本数据库，答案应该是 3859次 购买。

3.编写一个SQL查询，该查询列出数据库的所有数据的月份（按名称）和该月份的购买次数。输出表的列应称为 `month` 和 `numPurchases`。

(一) 上传到 `cssubmit` 与的名称文件 `Q3.sql` 只包含您的查询。(b) 参见 <https://dev.mysql.com/doc/refman/8.0/en/date-and-time-functions.html> (c) 输出应包含以下行

```
+      +-----+      +---+      +
+-----+      +      | 一个月
numPurchases      |      +
+-----+      +
+-----+      +      | 一月429 |
|二月| 432 |
```

4.编写一个SQL查询，该查询为数据库中的每个客户列出其客户ID和该客户在整个一年中购买的锥体总数，首先是最大的购买者。

(a) 上传到 `Q4.sqlcssubmit` 将名称文件 仅 包含查询的。(b) 对于提供的样本数据库，客户1购买了473个圆锥体。

5.编写创建存储的函数所需的代码，该函数 `numFlavours (coneNumber INT)` 返回特定圆锥体中不同的勺状口味的数量。

(一) 上传到 `cssubmit` 与名称文件 `Q5.sql` 只包含所需的代码
创建过程。(b) 您的答案必须以 `CREATE FUNCTION numFlavours (coneNumber INT)` 开头误
(复制并
粘贴此内容，以确保您不会拼过程名称)。(c) 通过输入测试您的过程 `SELECT numFlavours (1000)` ；它应该会返回值

2.3

中问题

6. 写一个简单的SQL查询，列出最昂贵的香料 炒到 表。

(一) 上传到 `cssubmit` 与的名称文件 `Q6.sql` 只包含您的查询。(b) 仅查看表即可检查
查询 `Scoop`

7. 编写代码以创建存储的函数 `isNutFree (coneNumber INT)` ，，则返回返回 `TRUE`，如果该圆锥的每个部分都没有螺母否则 `FALSE` 。目前，华夫饼蛋卷是用花生油制成的，而威化饼和糖蛋卷都不含坚果，椰子和澳洲坚果是仅有的用坚果制成的勺子口味。

(一) 上传到 `cssubmit` 文件 `Q7.sql`。只包含创建功能所需的代码 (b) 您可能会发现
<https://dev.mysql.com/doc/refman/8.0/en/if.html> 有用。(c) 您可以假定真实数据库
将不再涉及任何包含坚果的产品。

8.编写一个SQL查询，列出每个客户/商店组合的客户电子邮件，商店位置和购买数量，以及该客户在所有商店中的购买总数。

(一) 上传到 `cssubmit` 与的名称文件 `Q8.sql` 只包含您的查询。(b) 您的输出应包含以下行

	esokullu@hotmail.com	科特索	70		
	esokullu@hotmail.com	弗里曼特尔	67		
	esokullu@hotmail.com	城市海滩	59		
	esokullu@hotmail.com	NULL	196		

高级问题

9.首先，对问题5的代码进行修改，以创建一个存储函数函数 `numScoops (coneNumber INT)`

，该返回具有指定圆锥号的圆锥体中勺的数量。接下来，编写创建存储过程所需的代码，该存储过程

```
(purchaseNum INT,
    PurchaseSummaryOUT
    oneScoop      INT, OUT
    twoScoop      INT, OUT
    threeScoop INT)
```

分析一次购买中的锥体，并将三个的值设置 OUT 变量为该购买中的一勺，二勺和三勺锥。

(一) 上传到 `cssubmit` 与名称文件 `Q9.sql` 只包含所需的代码

创建两个存储的程序（程序和支持功能）。（b）请注意，`purchaseNum` 是一个 `IN` 参数，而其他三个都是 `OUT` 参数。（c）不要太担心使之高效，如果您需要使用三个非常相似的SQL查询，则可以这样做。（d）使用在此问题开头创建的存储函数可能会很有用。（e）您应该通过致电来测试代码

```
CALL PurchaseSummary (120, @ one, @ two, @
three) ; 选择@one, @two, @three;
```

并且应该得到输出

4

```

+   +   +
----- + | @one | @two
| @≡ | + ----- + -----
+ ----- + | 1 | 1 | 1
| + ----- + ----- +
----- +

```

10. 每个月, Scoops Ahoy都会向最纯洁的商店颁发“月度商店”奖。 -

进行了追逐（仅根据购买数量）。编写一个存储过程 `createMonthlyWinners()`，该表创建一个表 `MonthlyWinners`，具有三列，分别是数据库中所有月份的月份，中奖商店以及该月份在该特定商店中的购买数量。

(一) 上传到 `cssubmit` 与的名称文件 `Q10.sql` 只包含必要的代码来创建该存储过程。

(b) 该过程本身将不会返回任何内容，而只会 **创建一个新表** 在同一数据库中具有正确值。

(c) 即使您连续多次调用该程序，也要确保它可以正常工作（从心理上想象该程序每月被调用一次）。

(d) 使用数据库中的所有数据，而忽略数据库可能仅包含数据的事实最后一个月的一部分。

(e) 确保考虑到实际数据库可能包含数年的数据，因此第一列应包含诸如 `'December 2018'`、`'January 2019'` 之类的值。

加分题

这是为那些享受写这些查询、函数和过程的挑战奖金问题。分配给它的标记只能用于抵消先前问题中

丢失的标记，因此，如果您之前犯了错误，则可以将其视为“保险政策”。如果您没有时间或意愿来解决这个问题，那么您可以简单地忽略它，并且在所有其他问题都正确的情况下仍然获得满分。

11.编写所需的代码创建一个存储功能 `costOfPurchase (purchaseNumber INT)`，将根据以下比较复杂的业务规则计算购房成本。购买中的每个锥体的成本为：

- 基本成本为周一至周五50c，周六100c和周日150c。
- 在“找到的圆锥体成本 圆锥体”表中。
- 单个瓢的瓢成本总和（可在找到 `Scoop`中）。
- 2锥筒50c的多重折扣优惠3锥筒150c的多重折扣。

然后，通过将所有锥体的成本相加，然后根据在找到的特定客户/购买折扣来减少此值获得最终购买成本 `customerPurchases` 表中，并四舍五入到最接近的美分，来。

- 上传到 `cssubmit` 与的名称文件 `Q11.sql` 只包含必要的代码来创建该存储过程。