

## 数据 70141 转让 1 -独占权

1 重要信息作业发布 2023 年 10 月

16 日截止时间 18:00 2023 年 11 月 3 日

可交付成果 1 SQL 代码(兼容 SQLite)可交付成果 2  
视频

批改与反馈提交截止日期后 3 个大学工作周内(2022 年 11 月 28 日或之前)

成绩占最终模块成绩的 30%

## 2 介绍

您的任务是使用关系数据库和 SQL 查询对简化版的大富翁游戏玩法进行建模。您的数据库和查询必须与 SQLite 兼容。

没有唯一正确的方法来完成这个任务。没有完美的解决方案。如果你把这个问题交给 10 个数据库专业人士，你会得到 10 个答案在细节上有所不同，我怀疑至少有 5 个答案在总体方法上有所不同。这是可以预料到的。需求分析和数据库设计在某种程度上是公式化的，但它们也是艺术，依赖于经验、直觉和创造力。这个作业既是关于解决问题的过程本身，也是关于实际解决问题的过程。所以评估会考虑到你的创造力。你的方法必须满足要求，但具体如何去做，取决于你自己。要有创意，要玩得开心!请记住，SQL 比我们在课程中看到的更多，所以鼓励您研究和使用我们没有涉及的 SQL 命令，尽管这不是评估的要求。

请注意:SQLite 不支持过程，在创建数据库时需要考虑到这一点。

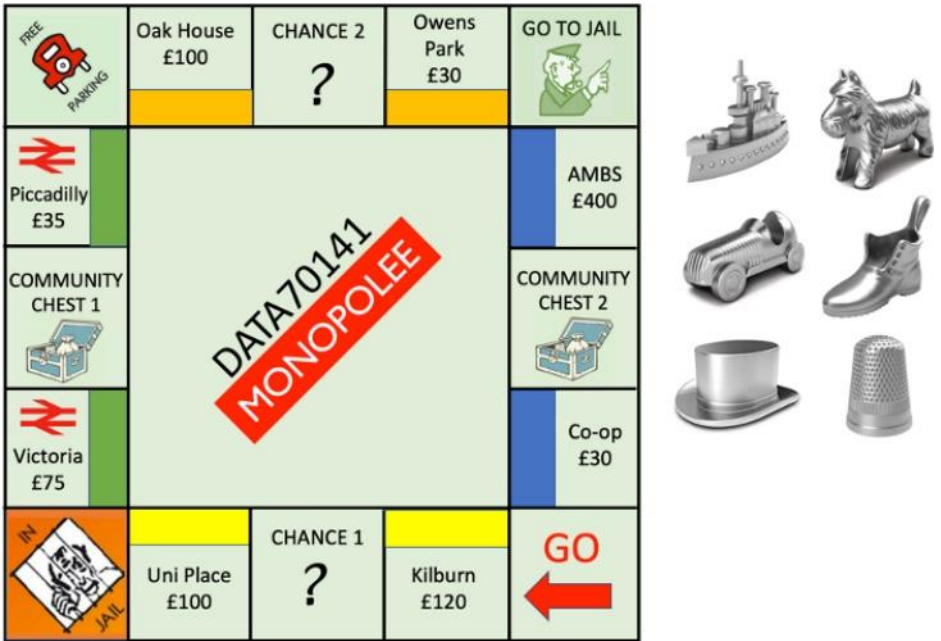


图 1:Monopolee Board

3 场景

下面的列表给出了游戏的数据定义和操作要求。这是故意没有用超精确的措辞写出来的。在实际项目中，需求的初稿很少是完整和明确的。如果你在这里发现任何含糊不清、遗漏或不精确的地方，你可以自己决定该怎么做，并在视频中提到这个决定。

- 1.最多有 6 名玩家。
- 2.有六种代币可供玩家选择。每个代币都有一个唯一的名字(狗、车、战舰、大礼帽、顶针、靴子)。
- 3.玩家必须选择一个且只能选择一个代币。
- 4.棋盘上的每个空间称为一个位置。一个地点是:一个角落，一个机会，一个社区基金，或一个财产。
- 5.机会，社区基金和角落可以组合在一起作为“奖励”。每个奖励都有一个唯一的 id，它的名字(例如，“机会 1”)和它的作用的文字描述。
- 6.存储关于玩家、属性和奖励的数据。
- 7.一个房产有一个独特的名字，还有一个购买成本，这也是如果其他玩家登陆该房产，该房产需要支付给业主的租金。
- 8.一个财产要么只有一个所有者，要么没有所有者。
- 9.每个玩家都有一个唯一的 id，名字，他们选择的代币，银行余额，他们当前的位置，以及他们 在该位置可能拥有的任何奖励。所有这些数据都应该有合适的默认值和约束。
- 10.奖励可以被许多玩家使用。在游戏过程中，一个玩家在任何时候最多只能拥有一个奖励。
- 11.必须对玩法进行审计追踪。对于玩家的每一个回合，审计跟踪应该存储玩家的 id、登陆地点、当前银行余额和游戏回合数。

3.1 游戏初始状态

属性和奖励的定义，以及玩家玩了一段时间后的游戏状态，如下图所示。编写 SQL 命令来相应地填充你的数据库表;你可以通过显式的 INSERT INTO 命令来完成。

| 成本(£) |              |     |
|-------|--------------|-----|
| 财产    |              | 颜色  |
| 橡树屋   | One hundred. | 橙色  |
| 欧文斯公园 | 30.          | 橙色  |
| amb   | 400          | 蓝色的 |
| 合     | 30.          | 蓝   |

| 奖金    | 描述                   |
|-------|----------------------|
| 机会 1  | 付给其他玩家每人 50 英镑       |
| 机会 2  | 向前移动 3 个空格           |
| 公益金 1 | 赢得选美比赛，你可以赢得 100 英镑  |
| 公益金 2 | 你从图书馆借的书过期了。罚款 30 英镑 |
| 免费停车  | 不采取行动                |
| 进监狱   | 进监狱，不通过 GO，不收 200 英镑 |
| GO    | 收集£200               |

| Bank Balance(£) |    |       |     |       |
|-----------------|----|-------|-----|-------|
| 球员              | 令牌 | 位置    |     | 属性拥有  |
| 玛丽              | 战舰 | 免费停车  | 190 | 大学的地方 |
| 比尔              | 狗  | 欧文斯公园 | 500 | 维多利亚  |

游戏现在将从上述状态继续，应用以下规则 R1-7。游戏按顺时针方向进行。

如果一个玩家在一个没有主人的地方着陆，他必须购买。

R2 如果玩家 P 降落在玩家 Q 拥有的房产上，则 P 向 Q 支付相当于该房产成本的租金。如果 Q 拥有某一特定颜色的所有属性，则 P 支付双倍租金。

R3 如果一个玩家在监狱里，他们必须掷出 6 才能出来。他们立即再掷一次。

如果玩家落在 GO 上或通过 GO，他们会得到 200 英镑。

如果玩家掷出 6，他们移动 6 格;无论他们落在什么位置都没有影响。然后他们立即得到另一个滚动。

R6 如果玩家落在 “Goto Jail” 上，他们移动到 Jail，而不通过 Go。

R7 如果玩家降落在机会或社区宝箱位置，则会发生奖励所描述的动作。

游戏的一轮被定义为每个玩家进入他们的下一轮。对于以下每一个玩法步骤 G1 到 G8，分别代表两轮游戏:

- 1.应用游戏规则，计算出在数据库中反映游戏状态所需的变化。

2.编写并运行更新数据库所需的 SQL 命令。您可能想研究“SQL UPDATE 算术运算”。

游戏发生在哈瓦那的 2024 年垄断者世界锦标赛上，你需要创建一个视图，“gameView”，数据库将显示在一个巨大的屏幕上，并将给出每个玩家的当前状态，整数，玩家的名字，他们当前的余额，他们当前的棋盘位置等。

游戏玩法回合 1:G1

简掷出 3

G2 诺曼掷出 1

玛丽掷出 4

G4 比尔掷出 2

游戏玩法第二轮:G5

简掷出 5

G6 诺曼掷出 4

G7 玛丽掷出 6，然后是 5 G8 比  
尔掷出 6，然后是 3

## 4 任务 1 -乌鸦脚符号中的 ER 图

首先，阅读上面详细介绍的需求，并以此为基础创建一个乌鸦脚符号的 ER 图。你应该把完成的图表画得整洁清晰，这样看起来就很专业，也很容易阅读。你可以用笔和纸来完成，或者有很多免费的工具可用(不需要为此付费购买绘图工具)。最重要的提示:先用笔和纸画出你的图表，只有在你画出来的时候才能画得整齐。否则你会浪费很多时间。

逻辑应该是第一位的;之后再让它看起来好看。

你的图表将在视频中用于解释数据库的设计。

## 任务 2 -架构

现在，您应该从 ER 图中派生出关系数据库模式图。你的模式应该包括每个属性域和任何默认值和约束。您应该确保图被清晰而整齐地表示出来。

你的模式将在你创建数据库时使用，也将在你的视频中用于解释设计。

## 6 任务 3 -实现

你需要创建几个 SQL 文件，这些文件将创建你的数据库，相应地填充它，并执行模拟游戏玩法的查询。您必须使用此处指定的文件名，以便自动测试工作。如果您不使用这些名称，则自动化测试可能无法工作，并且您可能无法获得您应得的所有分数:

创造。SQL this 必须包含成功创建数据库所需的所有查询，并具有正确的约束。

填充。SQL this 必须包含成功填充数据库以匹配初始状态所需的所有查询，如上所述。

q1。sql 这应该模拟 G1 的玩法，如上所述。

q2。sql 这应该模拟 G2 的打法，如上所述。

第三季。sql 这应该模拟 G3 的玩法，如上所述。

第四季度。sql 这应该模拟 G4 的玩法，如上所述。

q5。sql 这应该模拟 G5 的玩法，如上所述。

q6。sql 这应该模拟 G6 的玩法，如上所述。

迄今为止。sql 这应该模拟 G7 的玩法，如上所述。

处置。sql 这应该模拟 G8 的玩法，如上所述。

视图。这应该包含一个显示游戏玩法排行榜的 SQL 视图，可以在每个回合之后调用它(例如:在一个回合的中间)。至少，这应该包含玩家的名字，他们在棋盘上的位置，他们的银行余额，以及拥有的属性。

这些文件要上传，按照黑板上提交页面上的说明。

## 7 任务 4 -视频

你需要制作一个包含以下内容的视频:

解释你的数据库设计，清晰显示你的 ER 图和 Schema(请清晰显示至少 5 秒)。你应该确保解释了应用于模式的任何规范化。

对数据库 SQL 代码的解释。你不需要解释每一行代码，甚至每一个查询，但你应该解释代码的关键方面，以及你用来帮助模拟游戏玩法的任何功能。

你应该展示你创建的视图，显示第 1 轮和第 2 轮之后的游戏状态。该视频的时长不应超过 7 分钟。

请按照 Blackboard 提交页面上的说明上传视频。

(结束)