


## 期待已久的动态数据库工具终于来了！



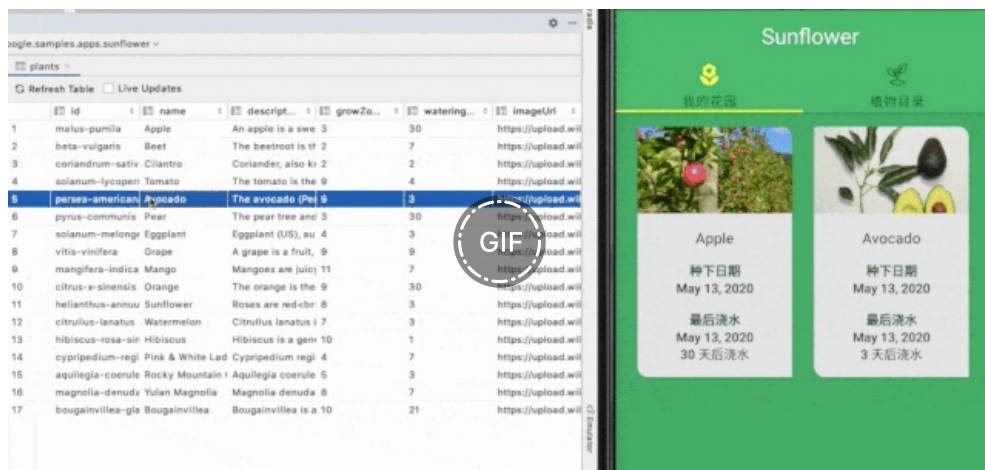
谷歌开发者   
已认证的官方帐号

已关注

2 人赞同了该文章

本地数据库的创建和管理是大多数移动应用工程中的核心组件之一，一般会通过直接使用 **SQLite** 或 **Jetpack Room 持久化库** 来完成。开发者们也在不断地寻求着更好的方式，使其能够在运行中的应用中直接检查和调试数据库。

最新的 Android Studio 4.1 (目前处于 Canary 版本) 内置了叫作 **Database Inspector** (数据库检查器) 的工具，它可以帮助开发者在已运行的应用中检查、请求以及修改数据库。



△ Database Inspector 可用于修改数据，就像修改电子表格一样

有了 Database Inspector，修改数据库就和编辑电子表格一样简单。如果您使用 Jetpack Room 并观察请求结果，对数据库的修改会直接体现在应用里。

本文中，我们将会在 Sunflower 应用中使用 Database Inspector 来测试一些边界值用例。Sunflower 是一款园艺类应用，它向开发者展示了基于 Android Jetpack 进行应用开发的最佳实践。所以强烈推荐大家克隆这个 [代码仓库](#)，并且按照本文的操作尝试一下。

### 概览

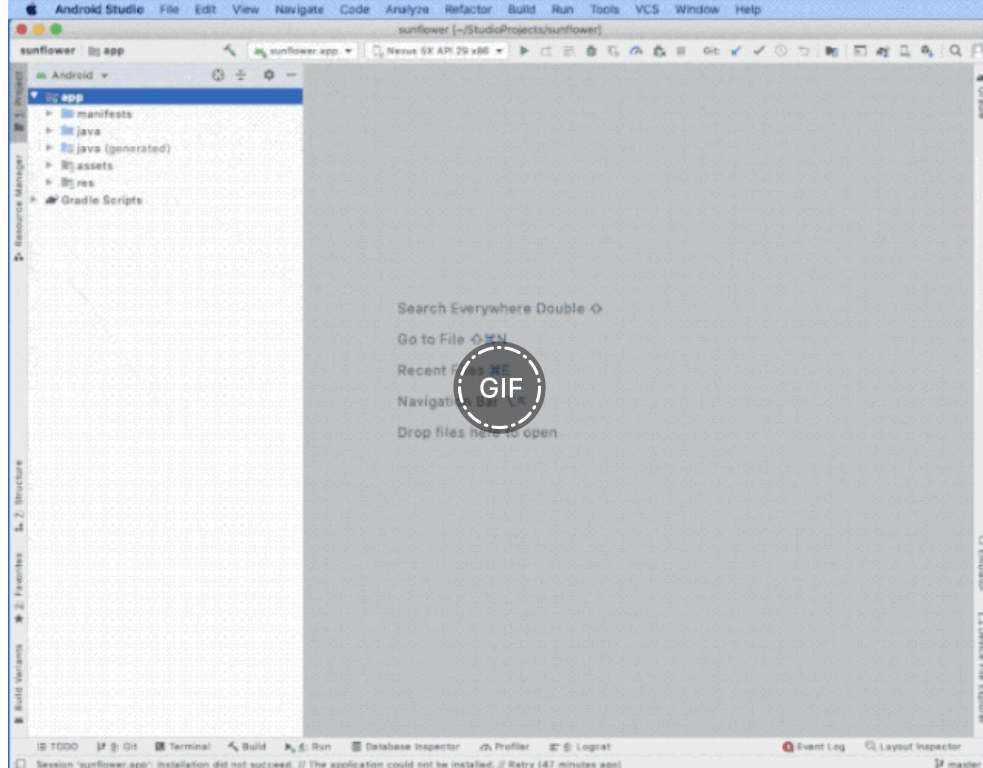
Sunflower 应用的 UI 包含两个标签页。先看一下 **植物目录** 标签页，里面列出了可以添加到 **我的花园** 的植物。在右上角有一个筛选按钮，点击该按钮会按生长区筛选植物，出现如下列表：



△ 点击该按钮来按照植物生长区筛选植物

可以很直观地看到，这个按钮会根据一些条件来筛选植物。假设我们对整个工程完全不了解，希望通过 Database Inspector 来搞清楚如何实现筛选功能。

从菜单栏依次选择 **View > Tool Windows > Database Inspector**，就能在 Android Studio 里打开 Database Inspector。初次打开工程的时候，需要静待工程编译完成后即可出现 **Database Inspector** 的选项。



△ 在菜单栏中依次选择 View &gt; Tool Windows &gt; Database Inspector 来打开 Database Inspector

这样就打开了 Database Inspector:

△ Database Inspector 窗口

运行 Database Inspector 需要在 API Level 26 或者更高的设备上运行应用，在下拉菜单里选择该应用的进程:

△ 在下拉菜单中选择运行中的应用进程

选择所要检查的应用进程之后，数据库的架构就会出现在下面的面板上。如果要查看 Sunflower 数据库表，需要找到**Databases**并展开**sunflower-db**:

△ 一旦选择了进程，应用相关的数据库就会列出来

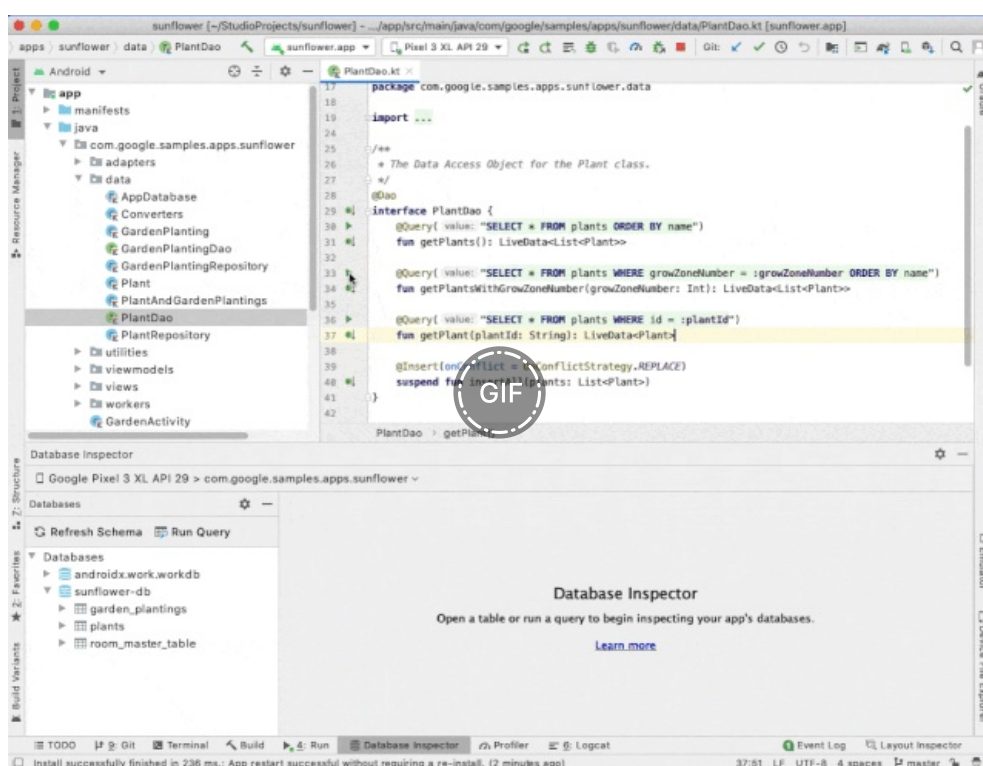
回到应用，我们看一下经过筛选的植物列表: 这个列表里有 Avocado (鳄梨)、Grape (葡萄)、Orange (橙子) 和 Tomato (番茄)。如果我按照植物名称进行排序，那么要找 Avocado 的话就会很方便。所以我们可以看看 Database Inspector 可不可以做到这点。

首先，双击 Database Inspector 里的 plants 表来显示表里的数据。数据是以默认每页 50 条记录来显示，不过如果数据总数更短，那么每页的结果数量和页数也会发生变化。点击名称列可以让植物表按照名称进行排序。正如我们所想，Avocado 已经排在列表的前面了，它就在数据表的第二行。

## 请求数据库

看一下 Avocado 的数据输入，在应用中可以使用 growZoneNumber 进行筛选。要验证这一点，我们运行一个请求，将 growZoneNumber 设置为 9，对应结果应该是 Avocado。实际上，这个请求已经在 PlantDao.kt 出现了，我们可以直接通过 Room 的 @Query 注解来调用这个请求。每个 @Query 注解的代码行数旁边都有一个运行小图标。

当我点击运行图标来请求 getPlantsWithGrowZoneNumber() 并且选择对应的数据库的时候，会弹出一个对话框让我们填写 growZoneNumber 的值。



这里我们输入 9 然后点击Run来查看请求结果。



除此之外，我们也可以在工具窗口里输入我们自己的请求并运行。这样能够提供更大的自由度，也不会受限于 Dao 接口里的请求语句定义。要运行我们自己的查询语句，点击**Run Query**并且从右侧新打开的标签页选择 plants 数据库。

△ 点击 Run Query 并且选择应用数据库

然后，在数据库下拉菜单旁边的文本框里输入下面这句查询语句，并点击**Run**。

*Select \* from plants where growZoneNumber=9*



△ 您可以在 Database Inspector 里执行 SQL 请求

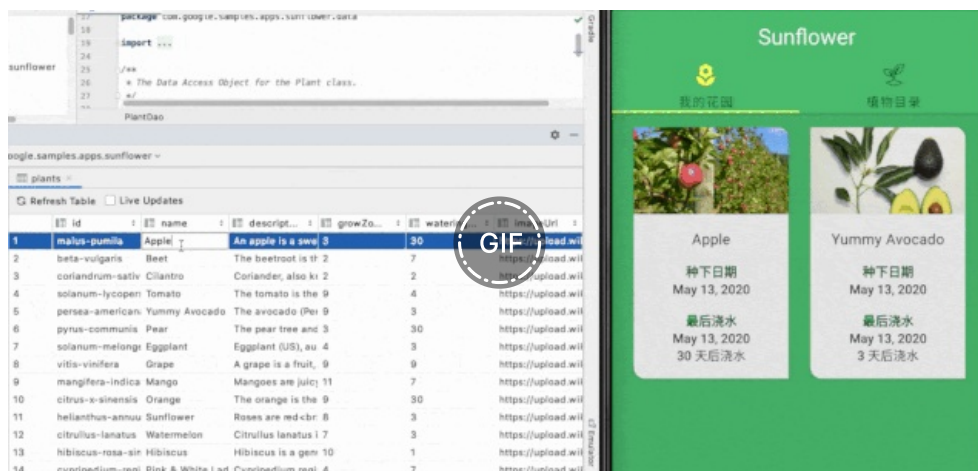
大家可以看到，这里请求的筛选结果和应用里开启筛选器的时候是一样的。

## 修改以及调试数据库

Database Inspector 使得调试应用变得轻而易举，它还可以让开发者直接修改设备上已运行应用的数据库值。

首先，我想在应用 UI 上测试一下较长的植物名称。我们打算使用 Database Inspector 直接修改数据库里的值，而不是通过修改数据源再刷新数据来进行测试。

现在这里的数据元素已经可以编辑，我将这里的 Apple 改为 A really special type of Apple，然后点确认。如果您也在同步进行操作，您可以输入一些期望长度的名称数据在 UI 中进行测试。



△ 如果您的应用使用 Jetpack Room 并且监听请求结果，那么您无需重启应用就能看到数据的变化

回到应用会发现我们什么操作都没做，应用就已经显示了最新的数据。如果您的应用使用 Jetpack Room，并且监听着请求结果 (使用 LiveData/Flow)，您就没必要发起数据库请求来刷新数据。否则，根据您应用触发请求的方式，可能需要重启应用或者重新打开相关的 activity 或者 fragment 才可以。这也算是迁移到 LiveData 或者 Flow 的一个不错的理由，这样也能发挥 Database Inspector 的全部潜能。

再来看看应用，可以发现这里的 CardView 并没有很好地显示较长的植物名称。我会在未来修复这个问题，接下来我们先来看看下面的测试。



▲ 赞同 2 ▼

● 添加评论

📌 分享

♥ 喜欢

★ 收藏

📄 申请转载

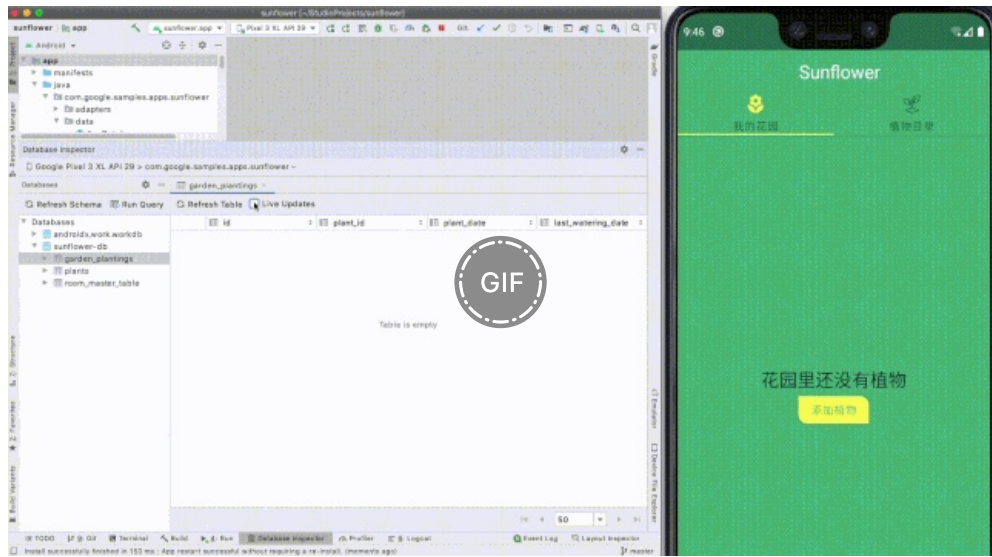
...



每个植物都有不同的浇水间隔，我想试试看如果错过了浇水的日子会发生什么。要实现这个目的，我们需要在花园里添加一些植物，但是首先，这里我们先在 Database Inspector 里勾选**Live updates**(实时刷新)。当启用**Live updates**之后，Database Inspector 会自动显示应用里对数据库所做的修改。

△ 选择 Live updates

然后回到**我的花园**标签页，添加一些植物，比如 Avocado 和 Eggplant，但是首先，回到 Database Inspector，然后双击*garden\_plantings* 来观察数据表。请注意观察图片，当添加新植物的时候，*garden\_plantings* 表会有哪些变化。



△ Database Inspector 会自动显示数据修改

这些植物的浇水周期都是三天。当然我不会等待三天来测试这个功能，这里我们直接编辑数据库，然后修改 *last\_watering\_day* 的值。我们再回到 Database Inspector，双击进入*garden\_plantings* 数据表，*last\_watering\_day* 位于数据库的最后一列。我会把两个记录的*last\_watering\_day* 值修改为目前日期稍早一些时候。





△ 对数据库的修改会直接展示出来

好吧，看上去好像调整的日期有点太靠前了，不过应该也能达到测试的目的。应用的 UI 似乎正常显示了所需浇水的日期。在下一步开发中，我们也许可以增加一个提醒功能，当已经超出浇水日期的时候，可以向用户发出提醒。

快来尝试新推出的 Database Inspector! 欢迎大家和我们分享心得。如果遇到任何问题，请在这里提出:

<https://issuetracker.google.com/issues/new?...>

[issuetracker.google.com](https://issuetracker.google.com)



[点击这里](#)下载最新 Android Studio 4.1 Canary 版本，即刻体验 Database Inspector 工具!

发布于 05-18

[数据库技术](#) [数据库](#) [Android](#)

## MySQL：创建数据库和数据表操作

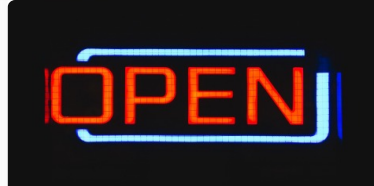
· 使用SQL创建数据库，数据库名：studentDB,如果数据库studentDB已存在，那么，先删除再创建。DROP DATABASE IF EXISTS StudentDB ;-----如果存在数据库StudentDB则删除数据库...aci

## 【笔记】MySQL常用命令

MySQL 第一章 认识数据库1.数据库介绍 数据库DB：data base 就是一个文件集合，用来存储数据。DB分类：关系型数据库、非关系型数据库 RDBMS关系型数据库管理系统：Relational Database Man...

CC是谁

发表于测试小白



这两位中国工程师，被MySQL官方点名了（有照片）  
腾讯云

## 看MySQL的参数调优及数据库锁实践有这一篇足够了

史上最全MySQL参数调优及数据库锁实践 1. 应用优化1.2 减少对MySQL的访问1.2.1 避免对数据进行重复检索1.2.2 增加cache层1.3 负载均衡1.3.1 利用MySQL复制分流查询1.3.2 采用分布式数据架构...  
海仔 发表于数据库优化...

还没有评论

写下你的评论...

