CoolShark商城
数据库与数据表设计

4. 用户管理相关数据表设计

目录

CONTENTS

1. 数据表概览

数据表概览

- 用户管理相关的数据表有：
  - ums_user：用户基本（常用）信息表
  - ums_user_detail：用户详细（不常用）信息表
  - ums_delivery_address：收货地址表
  - ums_login_log：用户登录日志表
  - ums_reward_point_log：用户积分日志表
  - ums_change_password_log：用户修改密码日志表

2. 设计思路

关于1对1的数据表

- 关于用户信息使用了ums_user和ums_user_detail这2张表来存储，这2张表是“1对1”的关系，其实，这些信息完全可以只使用1张表来存储！使用2张表的设计观点来自于：
  - 某些信息，可能在绝大部分情况下你都不需要使用到，例如用户的学历；
  - 数据表的字段越少，程序中的数据模型（实体类）就越小，当程序运行时，占用的内存空间越少。

3. 常见问题

常见问题

- **问**：用户为什么没有权限相关的设计？
- **答**：本项目基于“自营”性的电商平台来设计，所以不存在“入驻商家”的概念，那么，在不考虑用户等级（或是否VIP会员）的情况下，所有用户的权限都是相同的，所以，不设计权限相关的概念。

常见问题

- **问**：在ums_user_detail表中的id有什么作用?
- **答**：由于ums_user和ums_user_detail是一对一的关系，在ums_user表中的id是自动递增的，则ums_user_detail表中的user_id一定是唯一的，甚至可以作为主键，所以，ums_user_detail表中的id确实没有意义，甚至在开发过程中都不会使用到这个字段的值（即使查询数据也会是通过user_id字段来查询），之所以仍然存在id，并且和其它表一样设计为自动递增的主键，主要还是为了满足设计规范，毕竟各种各样的规范都是资深前辈们的经验的总结，也许在某些场景中体会不到其中的价值，但是，按照规范来做，也许可以规避一些特定场景下才会出现的问题。

常见问题

- **问：**在收货地址表中，省、市、区为什么既存在代码，又存了名称？（1/2）
- **答：**我国的每个省、市、区均有特定的代码，例如“河北省石家庄市桥西区”的代码就是“130104”。在数据库中存储代码是一种规范的做法，更利于检索或统计特定的数据，例如河北省的石家庄市有“桥西区”（代码130104），河北省的刑台市也有“桥西区”（代码130503），甚至河北省的张家口市还有“桥西区”（代码130703），如果没有区的代码，要找出“河北省石家庄市桥西区”的相关数据，至少需要将“石家庄市”和“桥西区”同时作为条件来查询，即使是这样，可能还存在名称可能改变的风险，例如湖北省的襄樊市就于2010年更名为襄阳市，而有区的代码时，只需要根据其代码进行查询即可，并且区域名称改变时并不会改变代码，可以更好的管理数据。

常见问题

- **问：**在收货地址表中，省、市、区为什么既存在代码，又存了名称？（2/2）
- **答（续）：**虽然存代码非常规范，但是代码并不能用于显示，站在用户的角度，更习惯于看“河北省石家庄市桥西区”这样的地址，如果在数据表中不存储这些名称，就需要进行关联查询（或类似操作），导致查询数据的效率比较低，如果直接存储在数据表中，查询一张表就可以了，查询效率就高一些，所以，一定程度上可以把名称视为冗余数据，存储这些数据的目的就是为了提高查询效率。

常见问题

- **问：**用户相关的数据表中有3张日志表，这些表有什么作用？
- **答：**关于ums_login_log、ums_reward_point_log、ums_change_password_log，都是日志信息表，是用于培养同学们的设计思想的，事实上，3张表并不多，在实际的企业应用中，可能还有更多的日志表，主要是为了出现某些未预期的问题时提供一些解决问题的参考或依据，如果系统能够长期稳定运行，或存储空间不足时，可以考虑删除，甚至在存储时，还可以应用消息队列缓解高并发时的写操作的压力，整体来说，日志表并不会明显的影响系统的性能，由于不是核心数据所以可以按需删除，预先存下来，以备不时之需，肯定是更好的。

谢谢！