**数据库设计**

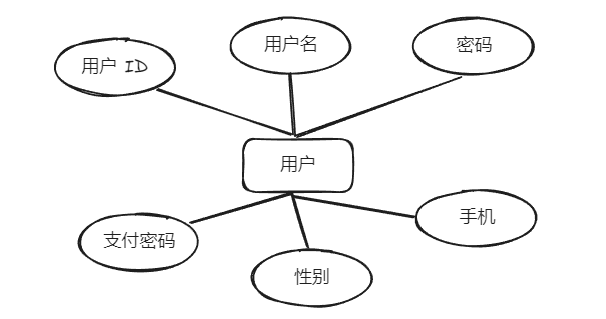
计划设计一个点餐系统，核心功能是用户能够通过菜单进行点餐，然后生成订单数据，并且能够使用优惠券来获取更加优惠的价格。

基于系统的核心功能，可以拆分为以下实体：用户、菜品、订单和优惠券。

**1.用户实体**

用户实体应该有的属性为：用户ID、用户名、用户密码、用户手机、用户性别和用户的支付密码。

其中，用户名和手机号用于登录以及注册，支付密码用于订单的支付。

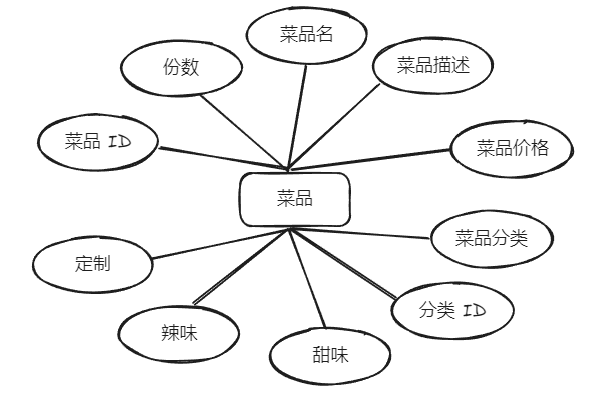


1. @Entity(tableName = "person\_table")
2. public class Person implements Serializable {
3. @PrimaryKey(autoGenerate = true)
4. public int UID;
5. public String username;   *// 姓名*
6. public String password;  *// 密码*
7. public long phoneNumber;  *// 手机号码*
8. public int gender;   *// 性别*
9. public int payPassword;   *// 支付密码*
10. public static final int GENDER\_MALE = 0;
11. public static final int GENDER\_FEMALE = 1;
12. public static final long serialVersionUID = 1L;   *// 序列化的版本号*
13. public Person(String username, String password, long phoneNumber, int gender, int payPassword) {
14. this.username = username;
15. this.password = password;
16. this.phoneNumber = phoneNumber;
17. this.gender = gender;
18. this.payPassword = payPassword;
19. }
20. }

**2.菜品实体**

菜品实体应该有的属性：菜品ID、菜品名、菜品描述、菜品价格、菜品选购份数、菜品是否支持定制、菜品辣味是否可选、菜品甜味是否可选、菜品分类名和菜品分类ID。

其中，菜品提供了定制、辣味、甜味的选项，而且设置菜品的分类名，可以对菜品进行分类显示。

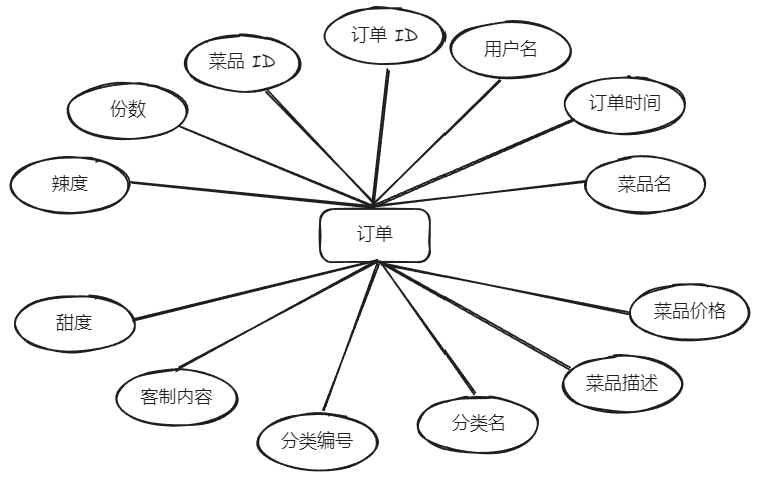


1. @Entity(tableName = "dish\_table")
2. public class Dish {     *// 菜品类*
3. @PrimaryKey
4. private int GID;    *// 菜品在菜单上的唯一编号*
5. private String name;    *// 菜品名*
6. private String description; *// 菜品描述*
7. private double price;   *// 价格(单价)*
8. private String category;    *// 分类类名*
9. private int CID;    *// 分类编号*
10. private boolean customizable;   *// 是否支持定制需求*
11. private boolean spicy;  *// 辣味是否可选*
12. private boolean sweet;  *// 甜味是否可选*
13. private int count;  *// 选购份数*
14. public Dish() {}
15. public Dish(int gid, String name, String description, double price, String category, int CID, boolean spicy, boolean sweet) {
16. this.GID = gid;
17. this.name = name;
18. this.description = description;
19. this.price = price;
20. this.category = category;
21. this.CID = CID;
22. this.spicy = spicy;
23. this.sweet = sweet;
24. this.count = 0;
25. }
26. }

**3.订单实体**

订单实体应该有的属性：订单ID、菜品ID、菜品名、菜品描述、菜品价格、菜品份数、菜品分类、菜品分类ID、菜品甜度、菜品辣度、菜品定制内容、用户名和订单时间。

其中，记录了菜品字段避免菜品下架或修改后找不到菜品。

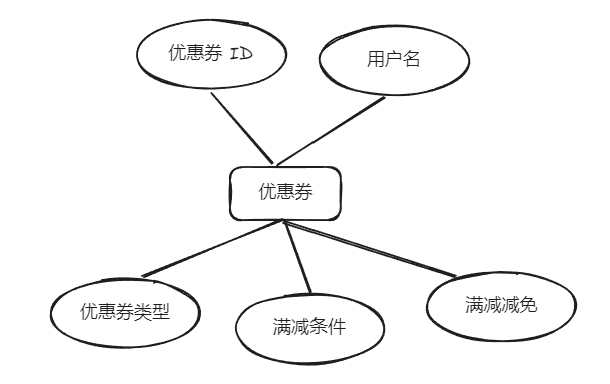


1. @Entity(tableName = "order\_table")
2. public class UserDish {
3. @PrimaryKey(autoGenerate = true)
4. public int OID; *// 订单号*
5. private int GID;    *// 菜品号*
6. private String name;    *// 菜品名*
7. private String description; *// 菜品描述*
8. private double price;   *// 价格*
9. private String category;    *// 分类类名*
10. private int CID;    *// 分类编号*
11. private int spicy;  *// 辣度*
12. private int sweet;  *// 甜度*
13. private String customText;  *// 客制化内容汇总*
14. private int count;  *// 选购份数*
15. private String userName;    *// 订单所属用户*
16. private long createdTime;   *// 订单生成日期（时间戳）*
17. public UserDish(int GID, String name, String description, double price, String category, int CID, int spicy, int sweet, String customText, int count, String userName) {
18. this.GID = GID;
19. this.name = name;
20. this.description = description;
21. this.price = price;
22. this.category = category;
23. this.CID = CID;
24. this.spicy = spicy;
25. this.sweet = sweet;
26. this.customText = customText;
27. this.count = count;
28. this.userName = userName;
29. }
30. public UserDish(Dish dish, String customText) {
31. this.GID = dish.getGID();
32. this.name = dish.getName();
33. this.description = dish.getDescription();
34. this.price = dish.getPrice();
35. this.category = dish.getCategory();
36. this.CID = dish.getCID();
37. this.customText = customText;
38. }
39. }

**4.优惠券实体**

优惠券实体应该有的属性为：优惠券ID、优惠券类型、折扣额度、满减条件、满减数额和用户名。

其中，优惠券主要分为满减券和折扣券两种类型，满减是达到满减金额可使用，折扣券是对订单价格进行打折。



1. @Entity(tableName = "coupon\_table")
2. public class Coupon {
3. @PrimaryKey(autoGenerate = true)
4. public int CID;             *//  优惠券ID*
5. private int type;       *// 优惠券类型*
6. public static final int DISCOUNT = 0;       *// 折扣券*
7. public static final int FULL\_REDUCTION = 1;     *// 满减券*
8. private double discount;        *// 折扣*
9. private double condition;   *// 满减条件*
10. private double reduction;   *// 满减减免*
11. private String username;    *// 优惠券所属用户*
12. public Coupon(int type, double discount, double condition, double reduction, String username) {
13. this.type = type;
14. this.discount = discount;
15. this.condition = condition;
16. this.reduction = reduction;
17. this.username = username;
18. }
19. }

**5.实体之间的关系**

（1）每个用户可以进行多次点餐，每次点餐生成一个订单，所以用户与订单之间的关系是一对多。

（2）每个用户可以拥有多张优惠券，每种类型的优惠券可属于不同的用户，所（3）以用户与优惠券之间的关系是多对多。

（4）每个订单只能使用一张优惠券，每张优惠券只能用于一个订单，所以订单与优惠券之间的关系是一对一。

（5）每个订单可以包含多个菜品，一个菜品也可以出现在多个订单中，所以订单与菜品之间的关系是多对多。

