**数据库表设计规范**

规范一：建议表名为{业务线前缀}+{实体名称}+{动作}，单词用下划线分开，例如：idso\_trade\_order, idso\_trade\_order\_pay。

注：其中idso为业务前缀，trade\_order为实体名称，pay为动作。

规范二：所有表必须添加名为id的主键，类型为BIGINT，自增。

规范三：所有字段必须有默认值，不能允许为NULL，二进制字段除外。

规范三：所有表必须名为 created\_at 和 updated\_at 的日期型字段，默认值分别为CURRENT\_TIMESTAMP 和CURRENT\_TIMESTAMP ON UPDATE CURRENT\_TIMESTAMP

规范四：所有允许用户输入的文本型字段的字符集必须是utf8mb4

规范五：所有字段名称必须是小写字母+数字的组合，单词之间用下划线分割

规范六：区别度不高的字段不加索引，不作为复合索引的首字段，比如性别、是/否开关等

规范七：超长字符串字段不要加全字段索引，可以只索引固定长度的前缀，例：ADD INDEX `idx` (field(30))

规范八：表引擎统一用 innoDB，表字符集统一为utf8

规范九：如无必要，不要使用触发器、存储过程、以及外键关联删除更新

规范十：所有表及字段都应该有清晰明了的注释

**SQL语句编写规范**

规范一：查询条件中的字段尽量加索引，区别度不超过30%的除外

规范二：查询条件中尽量不要出现表达式及LIKE操作符，尤其是只有表达式的情况

规范三：不要把NULL当做空字符串或0值来比较和计算，千万不要。

规范四：DISTINCT，GROUP BY，ORDER BY涉及到的字段应尽量加上索引，区别度不高的除外

规范五：尽量列出全部需要查询的字段，而不是用 \*

规范六：查询语句一次性返回数据不要太多，如果列表数据中单条记录中有比较大的字段，可以先不返回，后面单条查询补全

规范七：尽量用状态字段或软删除标志字段来代替DELETE操作

规范八：UPDATE语句必须加过滤条件，尤其是扣款、减库存等操作，避免超扣、超卖

规范九：UPDATE语句一次性更新的数据条目数不要太多

规范十：尽量避免多表JOIN，如果可以用内连接就不要用外连接，可以用左外连接就不要有右外连接