

## MySQL 数据库系统概念（主键、外键、索引）练习卷

### 一、选择题（每题 2 分，共 30 分）

1. 以下关于主键的说法，正确的是（ ）

- A. 一个表可以有多个主键
- B. 主键字段的值可以为 NULL
- C. 主键用于唯一标识表中的每一行记录
- D. 主键只能由单个字段组成

1. 在 MySQL 中，创建外键约束时，要求外键列的数据类型必须与被引用表的主键列数据类型（ ）

- A. 完全相同
- B. 可以不同，但必须兼容
- C. 外键列数据类型范围更大
- D. 没有要求

1. 下列哪种情况不适合创建索引？（ ）

- A. 经常用于 WHERE 子句条件判断的字段
- B. 表记录数较少的情况
- C. 经常用于连接查询的字段
- D. 经常用于排序的字段

1. 若创建一个包含外键约束的表，当主表中的相关记录被删除时，希望子表中对应的记录也自动删除，应使用的外键约束选项是（ ）

- A. ON DELETE CASCADE
- B. ON DELETE SET NULL
- C. ON DELETE RESTRICT
- D. ON DELETE NO ACTION

1. 关于联合主键，以下说法错误的是（ ）

- A. 联合主键由多个字段共同组成
- B. 联合主键中所有字段的组合值必须唯一
- C. 联合主键可以提高数据的查询效率
- D. 联合主键中每个字段都不能为 NULL

1. 在 MySQL 中，索引的类型不包括（ ）

- A. 唯一索引 (UNIQUE INDEX)
- B. 普通索引 (INDEX)
- C. 全文索引 (FULLTEXT INDEX)
- D. 循环索引 (LOOP INDEX)

1. 当在表中创建外键时，数据库会自动在（ ）上创建索引。

- A. 主键列
- B. 外键列
- C. 主键列和外键列
- D. 表中所有列

1. 以下关于索引的描述，正确的是（ ）

- A. 索引越多越好，能提高所有操作的性能
- B. 创建索引会增加数据插入、更新和删除的时间开销
- C. 索引只能在创建表时定义
- D. 索引不会占用数据库的存储空间

1. 若要在 MySQL 的 students 表中，将 student\_id 字段设置为主键，最恰当的 SQL 语句是（ ）

- A. ALTER TABLE students ADD PRIMARY KEY (student\_id);
- B. CREATE TABLE students (student\_id INT PRIMARY KEY);
- C. CREATE INDEX pk\_student\_id ON students (student\_id);
- D. ALTER TABLE students MODIFY COLUMN student\_id INT PRIMARY KEY;

1. 在数据库设计中，外键的主要作用是（ ）

- A. 提高查询效率
- B. 保证数据的完整性和一致性

C. 唯一标识表中的记录 D. 减少数据的存储空间

1. 对于一个包含大量数据的表，在以下哪种情况下，添加索引能最显著提高查询性能？（ ）

A. 查询语句中使用了 ORDER BY 子句，且排序字段没有索引 B. 查询语句中使用了 SELECT \*

C. 查询语句只涉及表中的少量字段 D. 表中的数据经常进行更新操作

1. 关于主键和唯一键的区别，以下说法正确的是（ ）

A. 主键和唯一键都允许有 NULL 值

B. 一个表只能有一个主键，但可以有多个唯一键

C. 主键和唯一键的约束强度相同

D. 唯一键不能用于创建外键，而主键可以

1. 在 MySQL 中，删除一个名为 idx\_name 的索引，正确的 SQL 语句是（ ）

A. DROP INDEX idx\_name; B. DELETE INDEX idx\_name;

C. REMOVE INDEX idx\_name; D. TRUNCATE INDEX idx\_name;

1. 当表中存在外键约束时，若要删除主表中的记录，以下说法正确的是（ ）

A. 可以直接删除，不受外键约束影响

B. 必须先删除子表中相关的记录，才能删除主表记录

C. 根据外键约束的设置，可能需要先删除子表记录，或者对子表记录进行相应处理

D. 只能通过修改外键约束选项来删除主表记录

1. 以下关于全文索引的说法，错误的是（ ）

A. 全文索引主要用于文本字段的快速搜索

B. 全文索引比普通索引更适合处理大量文本数据的模糊查询

C. 在创建全文索引时，不需要考虑字段的数据类型

D. MySQL 的 MyISAM 和 InnoDB 存储引擎都支持全文索引

## 二、判断题（每题 1 分，共 20 分）

1. 主键可以唯一标识表中的每一行记录，并且不允许有重复值和 NULL 值。（ ）

1. 外键必须是另一个表的主键，不能是唯一键。（ ）

1. 创建索引可以提高所有查询的性能，没有任何负面影响。（ ）

1. 一个表中可以同时存在多个外键约束。（ ）

1. 联合主键中的每个字段都可以单独作为主键使用。（ ）

1. 在 MySQL 中，删除表时，表上的所有索引和约束也会被自动删除。（ ）

1. 唯一索引和主键的功能完全相同，只是名称不同。（ ）

1. 外键约束可以防止在子表中插入无效的数据。（ ）

1. 索引会占用数据库的存储空间，并且会增加数据维护的成本。（ ）

1. 若要在已存在的表中添加主键，只能通过删除表并重新创建的方式实现。（ ）

1. 当主表中的主键值发生改变时，子表中的外键值会自动更新。（ ）

1. 普通索引可以加快数据的查询、插入、更新和删除操作。 ( )
1. 在创建外键约束时，不需要指定被引用表的名称和主键列。 ( )
1. 主键和外键的组合可以实现表之间的关联关系。 ( )
1. 全文索引只能在 CHAR、VARCHAR 和 TEXT 类型的字段上创建。 ( )
1. 索引一旦创建，就不能进行修改或删除。 ( )
1. 外键约束可以保证数据的参照完整性，避免出现孤立的数据。 ( )
1. 联合主键的查询效率一定比单个主键的查询效率高。 ( )
1. 在 MySQL 中，一个表只能有一个聚集索引，通常主键就是聚集索引。 ( )
1. 为了提高查询性能，应该为表中的每一个字段都创建索引。 ( )

### 三、简答题（每题 5 分，共 30 分）

1. 简述主键的作用及其与唯一键的主要区别。
1. 说明外键约束的作用，并解释 ON DELETE CASCADE、ON DELETE SET NULL、ON DELETE RESTRICT 和 ON DELETE NO ACTION 这几种选项的含义和应用场景。
1. 分析在什么情况下适合创建索引，什么情况下不适合创建索引。
1. 阐述联合主键的概念和使用场景，并说明其优缺点。
1. 简述索引的类型及其各自的特点和适用场景。
1. 当表中存在外键约束时，在对主表和子表进行数据操作（如插入、更新、删除）时，需要注意哪些问题？

### 四、编程题（20 分）

假设有两个表：departments 表，包含字段 department\_id（部门 ID，主键）、department\_name（部门名称）；employees 表，包含字段 employee\_id（员工 ID，主键）、employee\_name（员工姓名）、department\_id（所属部门 ID）。请完成以下操作：

1. 使用 SQL 语句创建 departments 表和 employees 表，并在 employees 表中建立外键约束，关联 departments 表的 department\_id 字段，当 departments 表中的记录被删除时，employees 表中对应的记录也自动删除。
1. 为 employees 表的 employee\_name 字段创建一个普通索引，为 department\_id 字段创建一个唯一索引（假设 department\_id 在 employees 表中不允许重复）。
1. 编写 SQL 语句查询每个部门的名称以及该部门的员工数量，要求使用表连接，并利用索引提高查询性能。