**Ôn Tập Lý Thuyết SQL - 4**

1. Khóa ngoại là trường mà giá trị của nó trùng với khóa duy nhất (unique key), không trùng với khóa chính (primary key)
   1. Đúng
   2. **Sai**
2. Cú pháp nào được dùng để xem thông tin định nghĩa của view
   1. sp\_view <View\_Name>
   2. sp\_help\_fulltext\_views <View\_Name>
   3. **sp\_helptext <View\_Name>**
   4. sp\_viewinfo <View\_Name>
3. SELECT \* FROM <Table\_Name>

Ý nghĩa của câu lệnh trên là gì

* 1. Câu lệnh trên chỉ hiện thị tên của tất cả các trường trong bảng
  2. **Câu lệnh trên hiển thị tất cả các bản ghi trong bảng**
  3. Câu lệnh trên không đầy đủ
  4. Câu lệnh trên hiển hị số trường và số bản ghi trong bảng

1. Khi một bảng bị xóa, chỉ có dữ liệu của nó bị xóa. Cấu trúc của bảng, index, các ràng buộc trên bảng đó không bị xóa.
   1. Đúng
   2. **Sai**
2. Chỉ ra cú pháp đúng cho mệnh đề Group By
   1. Groupby <Tên Trường>
   2. **Group By <Tên Trường>**
   3. Groupby <Tên Bảng>
   4. Group By <Tên Bảng>
3. Giả sử bạn có một bảng tên là SSC\_Result có các trường như student\_name, Marks, Sub, School\_Name. Viết câu truy vấn để hiển thị School\_Name và Marks của những sinh viên đạt điểm cao nhất trong môn Maths
   1. SELECT MAX (Marks), School\_Name, Student\_Name FROM SSC\_Result WHERE sub = 'Maths' GROUP BY School\_Name
   2. **SELECT MAX (Marks), School\_Name FROM SSC\_Result**
   3. SELECT Marks(Max), School\_Name, Student\_Name FROM SSC\_Result WHERE sub = 'Maths' GROUP BY School\_Name
4. DML viết tắt bởi
   1. Data Modeling Language
   2. **Data Manipulation Language**
   3. Data Moulding Language
   4. Data Marketing Language
5. Cơ chế đảm bảo mỗi bản ghi trong bảng phải là duy nhất và không có bản ghi nào bị trùng nhau một cách ngẫu nhiên gọi là \_\_\_\_\_\_
   1. Toàn vẹn tham chiếu (Referential Integrity)
   2. Toàn vẹn do người dùng tự định nghĩa (User-Defined Integrity)
   3. **Toàn vẹn thực thể (Entity integrity)**
   4. Toàn vẹn miền (Domain Integrity)
6. Một trường hay một tập hợp các trường mà giá trị của chúng trùng với khóa chính hay khóa duy nhất trong chính bảng đó hay trong một bảng khác gọi là \_\_\_\_\_\_\_\_
   1. Unique Key
   2. **Foreign Key**
   3. Referential Key
   4. Candidate Key
7. Xác định các ràng buộc được hỗ trợ trong SQL Server 2012 (chọn 3)
   1. **REFERENCE**
   2. VIEW
   3. **PRIMARY KEY**
   4. **DEFAULT**
   5. INDEX
8. Xác định cơ chế dùng để đảm bảo toàn vẹn thực thể (chọn 2)
   1. **PRIMARY KEY**
   2. VIEW
   3. **UNIQUE CONSTRAINT**
   4. FOREIGN KEY
9. Giả sử chúng ta đã tạo xong bảng, xác định cú pháp để thiết lập một ràng buộc cho bảng
   1. CREATE TABLE <Table\_Name>

(<Column\_definition> <Table\_constraint>)

* 1. **ALTER TABLE <Table\_name>**

**ADD CONSTRATINT <Constraint\_Name> <Type\_Of\_Constraint>**

* 1. ALTER TABLE <Constraint\_Name>

ADD CONSTRATINT <Type\_Of\_Constraint>

* 1. CREATE TABLE <Table\_Name>

(<Table\_constraint>)

1. Xác định cú pháp đúng để tạo View
   1. CREATE VIEW view\_name [(column [,...n])]

[WITH ENCRYPTION]

* 1. CREATE VIEW <view\_name> ON <Select statement>
  2. CREATE VIEW <view\_name> ON <Table\_Name>
  3. **CREATE VIEW view\_name [(column [,...n])]**

**[WITH ENCRYPTION]**

**AS**

**select\_statement**

**[WITH CHECK OPTION]**

1. Các thủ tục nội tại hệ thống (System Stored Procedures) được tạo bởi người dung
   1. **Sai**
   2. Đúng
2. Xem xét trigger cho như sau

CREATE TRIGGER deletetrigger

ON Emp

FOR DELETE

AS

IF (Select count(\*) from deleted)<10

BEGIN

Print ‘You can delete the records if it is more than 10 records’

ROLLBACK TRANSACTION

END

Người dùng sẽ nhận được phản ứng (response) gì khi cố gắng thực hiện xóa tất cả các bản ghi trong bảng EMP ?

* 1. Người dùng sẽ nhận được một thông điệp cảnh báo và tiếp theo là thực hiện xóa các dòng.
  2. **Người dùng sẽ nhận được một thông điệp cảnh báo và giao tác (transaction) sẽ bị khôi phục lại nếu số dòng xóa nhỏ hơn 10.**
  3. Người dùng có thể dễ dàng xóa các bản ghi khi trigger không được kích hoạt.
  4. Người dùng sẽ nhận được một thông điệp cảnh báo và giao tác (transaction) sẽ bị hủy bỏ nếu số dòng xóa lớn hơn 10.

1. Giả sử bạn có một bảng tên là A, trong bảng đó có trường tên là ‘xyz’ với kiểu dữ liệu là int. Giá trị của tất cả các trường trong một bản ghi của bảng đó phải là giá trị số nguyên
   1. Đúng
   2. **Sai**
2. Tìm ra phát biểu đúng (chọn 2)
   1. Để tạo mối quan hệ trên các bảng, các bảng đó phải có cùng số trường và các trường phải giống nhau về kiểu dữ liệu
   2. **Một mối quan hệ là một sự tương ứng giữa các bản ghi của hai bảng**
   3. **Các bảng có quan hệ với nhau dựa trên các trường chung**
   4. Các bảng có quan hệ với nhau phải nằm trong cùng một CSDL
3. Trong những phát biểu dưới đây, phát biểu nào là đúng (chọn 3)
   1. **Khi sử dụng Join, việc truy xuất dữ liệu trên các bảng dựa trên mối quan hệ logic của chúng.**
   2. Join chỉ được dùng để truy xuất dữ liệu từ 2 bảng. Khi có nhiều hơn 2 bảng, Join không sử dụng được.
   3. **Join chỉ ra khóa ngoại trong một bảng và khóa chính trong một bảng khác**
   4. **Join thường chỉ ra quan hệ khóa ngoại trên các bảng.**
   5. Joins không được dùng trong các mệnh đề FROM và WHERE
4. Trong mô hình CSDL phân cấp (Hierarchical Database model) một bảng cha không thể có nhiều hơn một bảng con
   1. Đúng
   2. **Sai**
5. Xác định cú pháp đúng cho LEFT OUTER JOIN (chọn 1)
   1. SELECT <Table\_name> FROM First\_table LEFT INNER JOIN Second\_table ON First\_Table.<common.Field> = Second\_Table.<common.Field>
   2. LEFT OUTER TABLE SELECT <Field\_List> from <First\_table>

<Second\_table> ON

First\_Table.<common\_Field>=Second\_Table.<common\_Fiel>

* 1. SELECT <Field\_List> FROM <First\_table>

LEFT OUTER TABLE <Second\_table>

ON

First\_Table.<common\_Field>=Second\_Table.<common\_Fiel>

* 1. **SELECT <Field\_List>**

**FROM <First\_table> LEFT OUTER JOIN <Second\_table>**

**ON First\_Table.<common\_Field>=Second\_Table.<common\_Fiel>**