**PHẦN I: PHÂN TÁN**

**Câu 1: Các bước phân tán? Khi phân tán cần bật dịch vụ gì? Cách bật như thế nào?**

* Các bước phân tán:

+ Cấu hình server:

* Publisher: chứa bản góc của csdl
* Distributor: điều phối các mãnh phân tán
* Subscriber: chứa csdl sau khi phân tán

+ Định nghĩa Publication: 1 container chứa Actical(view, sp, table, UDF).

+ Định nghĩa Subsciption: 1 container nhận publication.

* Khi phân tán phải bật dịch vụ SQL Server Agent
* Cách bật: run gõ services.msc-> start SQL Server Agent

**Câu 2: thuộc tính folder RELPDATA là gì? ở đâu? Mục đích của thư mục để làm gì?**

* Thuộc tính là share, có quyền read/write. Mục đích để chứa các dữ liệu trao đổi trong quá trình update dữ liệu từ các phân mảnh về csdl gốc , và từ gốc đi đến các phân mảnh

**PHẦN II: SỰ TRONG SUỐT PHÂN TÁN**

**Câu 1: Phân mãnh ngang là gì? Phân mãnh dọc là gì? Phân mãnh ngang nguyên thủy là gì? Phân mãnh ngang dẫn xuất là gì?**

* Phân mảnh ngang: là chia 1 quan hệ theo các bộ. Mỗi phân mảnh ngang là một tập con của quan hệ. Có hai loại phân mảnh ngang đó là phân mảnh ngang dẫn xuất và phân mảnh ngang nguyên thủy.
  + Phân mảnh ngang nguyên thủy là sự phân mảnh dựa trên một qua hệ vị từ được định nghĩa trên một quan hệ
  + Phân mảng ngang nguyên thủy là sự phân rả một quan hệ dựa vào các vị từ được định nghĩa trên một quan hệ khác.
* Sự phân mảnh dọc của một quan hệ toàn cục là việc chia các thuộc tính vào hai nhóm; các mảnh nhận được từ phép chiếu quan hệ toàn cục trên mỗi nhóm

**Câu 2: Nhân bản là gì? Lí do nhân bản? Nhân bản không hiệu quả với lệnh gì?**

* Lý do : truy xuất nhanh, tính sẵn sàng cao
* Không hiệu quả: update, delete, insert

**Câu 3: Phân hoạch là gì?**

* Phân hoạch là dữ liệu không nhân bản và nó chỉ tồn tại một bản sao duy nhất cho mỗi phân mảnh trên mạng.

**Câu 4: Nêu rõ các quy tắc phân mảnh: Tính tái thiết, tính đầy đủ, tính tất biệt**

* Tính tái thiết: Một quan hệ R được phân mảnh thành các quan hệ R1, R2, … Rn , thì cần định nghĩa phép toán sao cho tái thiết lai quan hệ gốc được.
* Tính đầy đủ: Nếu một quan hệ được phân rã thành các mảnh R1, R2,.., Rn thì mỗi mục dữ liệu trong R cũng có thể có trong một hay nhiều Ri.(đặc tính này nói lên sự phân mảnh không mất thông tin)
* Tính tách biệt: Nếu một quan hệ R được phân rã thành các phân mảnh R1, R2, …Rn và mục dữ liệu di nằm trong phân mảnh Rj sẽ khong nằmt trong bất kì một phân mảnh Rk nào khác.

**Câu 5: Sp trong suốt là gì? Điều kiện để trong suốt? Có mấy mức trong suốt, nêu rõ các mức trong suốt đó?**

* SP trong suốt là sp thực hiện ở bất kì server nào cũng cho kết quả đúng như nhau.
* Điều kiện để trong suốt:
  + Cùng tên LINKSERVER
  + Cùng tên cơ sở dữ liệu
* Có 4 mức trong suốt:
  + Trong suốt phân tán: Dữ liệu phân tán xuống và tên các database phân tán giống nhau.
  + Trong suốt định vị: Khi gọi tên các SP phân mảnh này chỉ cần gọi tên mà không cần gọi ra vị trí bảng đó
  + Trong suốt ánh xạ cục bộ: phải cung cấp tênn các phân mảnh và vị trí cấp phát chúng.
  + Không trong suốt: phải viết lệnh theo hệ quản trị cơ sở dữ liệu tương ứng.

**Câu 6: LinkServer dùng để làm gì? Khi tạo LinkServer cần điều kiện gì? Muốn truy xuất đến phân mảnh gần nhất đầu tiên chúng ta phải tạo LinkServer như thế nào?**

* LinkServer: dùng để truy xuất dữ liệu các subcriber
* Điều kiện: LinkServer cùng tên link và cùng tên cơ sở dữ liêu.
* Phụ thuộc vào cách đặt tên link sao cho phù hợp (ví dụ: gần đặt 1, xa đặt 2)

**PHẦN IV: Tối ưu hóa truy vấn**

**Câu 1: Cách tối ưu hóa truy vấn**

* Thực hiện phép chọn, chiếu trước, phép kết sau. Tìm cách khử phép (nếu được). Nếu một điều kiện xuất hiện nhiều lần trong where thì dùng các biến đổi tương đương để làm cho điều kiện xuất hiện một lầ. Trong mệnh đề AND điều kiện nào có xác xuất sai cao đặt đầu, mệnh đề OR ngược lại.

**Phần V: Giao tác phân tán**

**Câu 1:** Giao tác phân tán là gì? Các loại giao tác? Sự giống nhau và khác nhau giữa hai giao tác này? Sự giống nhau và khác nhau của giao tác phân tán và giao tác tập trung? Khi viết cần bật dịch vụ gì? Cách bật như thế nào?

* Giao tác phân tán là tập các lệnh đọc ghi và các bước tính toán cần thiết lên cơ sở dữ liệu và có thêm khả năng tái thiết dữ liệu( nếu không có khả năng tái thiết thì giao tác phân tán là một SP thuần túy).
* Các loại giao tác: Giao tác phẳng (chỉ có một lệnh BEGIN TRANSACTION và một lệnh COMMIT TRACSACTION) và giao tác lồng(có thể có nhiều lệnh BEGIN và COMMIT lồng vào nhau).
* Sự khác nhau giữa giao tác phân tán và giao tác tập trung

+ Giao tác phân tán: Giao tác phân tán phân bố nhiều nơi. Khi hai giao tác chạy ở hai server thì nếu một giao tác gặp sự cố thì giao tác kia vẫn chạy bình thường.

+ Giao tác tập trung: Giao tác tập trung ở một nơi. Khi giaoc tác gặp sự cố thì tất cả các giao tác khác đều dừng.

* Khi viết giao tác cần bật dịch vụ DISTRIBUTE TRANSATION COODINATOR
* Vào run gõ services.msc->start Distribute Transaction Coodinator.

**Câu 2: Nêu rõ các tính chất giao tác: nguyên tử, nhất quán, biệt lập, bên vững? ý nghĩa của các tính chất?**

* Tính nguyên tử: Một giao tác được xử lý như một đơn vị hoạt tác. Chính vì thế nếu một lệnh trong giao tác bị lỗi thì tất cả các lẹnh trong giao tác sẽ bị hủy bỏ. và ngược lại
* Tính nhất quán: khi thực hiện các lệnh insert, update, delete nếu thành công thì tất cả dữ liệu phải được cập nhật hết,còn nếu bị lỗi thì dữ liệu phải trở về vị trí ban đầu.
* Tính biệt lập: Khi một thao tác đang thực thi thì các giao tác khác không thể thấy dữ liệu của nó.

Ví đang delete thì không thể thêm đợi delete xong mới được thêm.

* Tính bền vững: một giao tác khi được commit thì kết quả của nó vẫn được duy trì có định và không bị xóa ra khỏi csdl.

**Câu 3: Giải thích cách lệnh BEGIN TRANSACTION, COMMIT TRANSACTION, @@TRANCOUNT, DEADLOCK, ROLLBACK, ABORT?**

* @@TRANCOUT: số giao tác đang được thực hiện trên giao tác hiện tại.
* BEGIN TRANSACTION: bắt đầu một thao tác, tăng @@TRANCOUNT thêm 1
* COMMIT TRANSACTION: giảm 1 trên @@TRANCOUNT
* ROLLBACK: đưa @@TRANCOUNT về 0

**Câu 4: Dịch vụ MS Distributed Transaction Coordinator là gì?**

* Là một trình quản lí điều phối các giao tác phân tán, nó cho phép các ứng dụng của client thao tác lên dữ liệu của các datasource trong một giao tác phân tác.
* MS DTC service điều phối sự đúng đắn của một giao tác phân tán, nó bảo đảm rằng hoặc là tất cả được cập nhật trên các server là được thực hiện, hoặc trong trường hợp lỗi thif xem như chưa thực hiện giao tác gì trên giao tác đó.
* Lưu ý: Nếu MSDCT không hoạt động thì cách khắc phục như sau:
  + Vào cmd gõ msdtc.exe-install
  + Sau đó start service Distribute Distributed Coordinator

Nếu không khởi động được thì làm một trong hai cách sau:

* + - Cách 1: Re-create msdtc log for DTC service
      * Vào cmd gõ msdtc -resetlog
      * Gõ tiếp net start msdtc
    - Cách 2: set the logon account permisstion for msdtc

**Phần VI : FORM**

**Câu 1: Các thuộc tính của ComboBox là gì? Ý nghĩa của nó?**

* BindingSource: chỉ ra dữ liệu mà ComboBox đó nằm ở đâu.
* DisplayMemBer: Dữ liệu hiện thị để người dùng chọn. Dữ liệu có thể là một cột trong DataTable
* ValueMember: là giá trị tương ứng với displaymember. Dữ liệu có thể là một cột trong DataTable

**Câu 2: Tại sao đăng nhập ở form đăng nhập thì mình có thể đăng nhập thành công được? khi đăng nhập thành công , tại sao biết được mã, họ tên , nhóm của người đăng nhập**

**Câu 3: “Device” trong BACKUP DATABASE QLVR TO DEVICE\_QLVT là gì?**

* Device là tên thiết bị dùng để lưu trữ database khi backup và restore, đường dẫn: tên server /Server Object/Backup Device/DEVICE\_QLVT.

Đặt tên có Device để phân biệt đây là 1 colum, không phải là một DB

**Câu 4: Khi backup db, thì bản sao lưu đó sẽ được sẽ được lưu trữ ở đâu?**

* Được lưu ở table backupset, đường dẫn tên server.Databases/SystemDatabase/msdb/Tables/System Table/backupset
* Có 2 dạng backup là update là override
  + Dạng UPDATE: backup bản này nối tiếp bản kia
  + Dạng OVERIDE: khởi tạo lại bản backup

Câu 5: Client và server liên kết, trao đổi dữ liệu bằng cách nào?

* Thông qua DataAdapter

Câu 6: Nút lệnh thêm, cái gì gọi AddNew();

* BindingSource

Câu 7: Để update dữ liệu và db, trước khi gọi lệnh Update thì cần phỉa thực hiện lệnh gì?

* EndEdit();
* ResetCurrentItem();

Câu 8: DataSet, DataTable, DataAdapter là gì?

* DataSet có thể hiểu là một bản ghi tạm, nó lưu trữ và chỉnh sửa dữ liệu ở Local Cache, tức là khi có ngắt kết nối tớ server đi nữa thì vẫn thao tác với dữ liệu trên DataSet. Sau khi xem sét và chính sửa dữ liệu xong ta tạo kết nối (DataAdapter) để Update dữ liệu từ Local(DataSet) về DataSource(SQL)
* Dữ liệu trong DataSet được lưu ở dạng DatâTble(DT). DT là một đối tượng chứa dữ lệu từ DataBase gởi qua, cấu trức gồm có các DataRow và DataColumn. DT có thể là 1 Table, 1 View, hoặc 1 SP, chứa dữ lệu nhưung không chứa lệnh.
* DataAdapter(DA) là một dạng cầu nối, giúp cácc lệnh như Select, Insert, Update, Delete để trao đỏi dữ liệu giữa Client và Server, và DA đổ dữ liệu từ DataSource về DS bằng phương thức Fill, GetData

Câu 9: BindingSource là gì?

* Là nơi chỉ ra (Binding) để thao tác dữ liệu, là cầu nối giưuã DataGridView và DataTable.

Câu 10: Các lệnh EndEdit, ResetCurrentItem sử dụng trong bài có ý nghĩa gì?

* AddNew() thêm một item/row vào danhh sách,
* CancelEdit() bỏ qua các dữ liệu đang chỉnh sửa trong rrơ và tả con trỏ về row cuối cùng. Chỉ có tác dụng với lệnh AddNew và dữ liệu đang được hiệu chỉnh nhưng chưa ghi vào DataSet.
* EndEdit: kết thúc quá trình chỉnh sửa dữ liệu và gơi và DataSet.
* ResetCurrentItem: Reread/ đọc lại row hiện tại ta đang đứng, dữ liệu hiện tại ở row ta đang đưungs không còn ở dạng tạm nữa(tức là dữ liệu có thể cancle edit) mà hiện thị dữ liệu đã chính thức ghi vào DataSet
* Update cập nhật data từ DataSet về DB thông qua TableAdapter.
* Count(): đếm xem số record tương ứng với row đang chọn tồn tại bao nhiêu bảng trong bindingsource được kiểm tra.
* RemoveCurrent Xóa row đang chọn ra khỏi DataSet