**Questions**

1. Cosa si intende per database?
2. Cos’è un DBMS?
3. Indica le principali clausole di uno statement SELECT in ordine di esecuzione logica. Descrivi per ciascuna delle clausole indicate la logica di funzionamento.
4. Descrivi, immaginando uno scenario a te familiare, il concetto di group by. Utilizza l’approccio che ritieni più efficiente per trasmettere il concetto (suggerimento: disegna anche una sola tabella in Excel o in word con poche colonne e pochi record e descrivi, basandosi sulla tabella stessa, un esempio di group by).
5. Descrivi la differenza tra uno schema OLTP e uno schema OLAP.
6. Dato un medesimo scenario di analisi, qual è la differenza in termini di risultato ottenibile tra una join e una subquery?
7. Cosa si intende per DML e DDL?
8. Quali istruzioni possono utilizzare per estrarre l’anno da un campo data? Proponi degli esempi.

1.Per **Database** si intende un contenitore capace di immagazzinare grandi quantità di dati e permette di manipolarli grazie alle operazioni CRUD.

2.Un **DBMS** (Database Management System) è un software che permette la manipolazione e interrogazione di Database.

3.La **SELECT** è un’istruzione che viene scritta per prima ma eseguita per ultima; essa espone ciò che vogliamo visualizzare ed è SEMPRE seguita da un FROM (da dove) e se vogliamo da WHERE(condizione-filtro), GroupBy(raggruppa e si utilizza con le formule di aggregazione), Having(filtro sulle formule di aggregazione) e OrderBy(ordinamento) e si possono aggiungere delle join(collegamenti con tabelle).

4.La **GroupBy** è una clausola che inseriamo quando utilizziamo formule di aggregazione per raggruppare i dati calcolati.

5.Lo schema **OLTP** è ottimizzato per la gestione dei dai (Data Entry) e il support delle transazioni quotidiane con focus su integrità, consistenza e sicurezza, mente, lo schema **OLAP** è ottimizzato per l’analisi dei dati e per analisi complessa di grosse qantità di dati (reporting).

6.Le **join** servono a combinare le righe da 2 o piu’tabelle secondo una condizione comune; la **subquery** è una query all’interno di un’altra capace di restituire uno o piu’ risultati dettate da delle clausole.

7.Il **DDL** crea modifica e elimina oggetti**,** mentre il **DML** sono delle istruzioni per interrogare e modificare i dati.

8.SELECT **YEAR**(Data\_Acquisto) FROM acquisto;

9.L’opertatore **AND** restituisce vero se entambe le condizioni sono vere, mentre, **OR** restituisce vero anche se solo una condizione è vera.

10.Si può innestare una subquery nella calusola SELECT.

11.La differenza principale tra gli opertaori **ON** e **IN** è che quest’ultimo è piu’ facile da gestire dal database e piu’ leggibile.

12.Si, l’operatore BETWEEN include anche gli estremi del range specificato.

**ULTIMA PAGINA C’È UNA SPIEGAZIONE DEL DATABASE CREATO**

1. Qual è la differenza tra gli operatori logici AND e OR?
2. È possibile innestare una query nella clausola SELECT?
3. Qual è la differenza tra l’operatore logico OR e l’operatore logico IN?
4. L’operatore logico BETWEEN include anche gli estremi del range specificato?

**Case Study**

ToysGroup è un’azienda che distribuisce articoli (giocattoli) in diverse aree geografiche del mondo.

I prodotti sono classificati in categorie e i mercati di riferimento dell’azienda sono classificati in regioni di vendita.

In particolare:

1. Le entità individuabili in questo scenario sono le seguenti:

* Product
* Region
* Sales

1. Le relazioni tra le entità possono essere descritte nel modo seguente:

* Product e Sales
* Un prodotto puo’ essere venduto tante volte (o nessuna) per cui è contenuto in una o più transazioni di vendita.
* Ciascuna transazione di vendita è riferita ad uno solo prodotto
* Region e Sales
* Possono esserci molte o nessuna transazione per ciascuna regione
* Ciascuna transazione di vendita è riferita ad una sola regione

1. Le entità Product e Region presentano delle gerarchie:

* L’entità prodotto contiene, oltre alle informazioni del singolo prodotto, anche la descrizione della categoria di appartenenza. L’entità prodotto contiene quindi una gerarchia: un prodotto puo’ appartenere ad una sola categoria mentre la stessa categoria puo’ essere associata a molti prodotti diversi.

*Esempio: gli articoli ‘Bikes-100’ e ‘Bikes-200’ appartengono alla categoria Bikes; gli articoli ‘Bike Glove M’ e ‘Bike Gloves L’ sono classificati come Clothing.*

* L’entità regione contiene una gerarchia: più stati sono classificati in una stessa regione di vendita e una stessa regione di vendita include molti stati.

*Esempio: gli stati ‘France’ e ‘Germany’ sono classificati nella region WestEurope; gli stati ‘Italy’ e ‘Greece’ sono classificati nel mercato SouthEurope*.

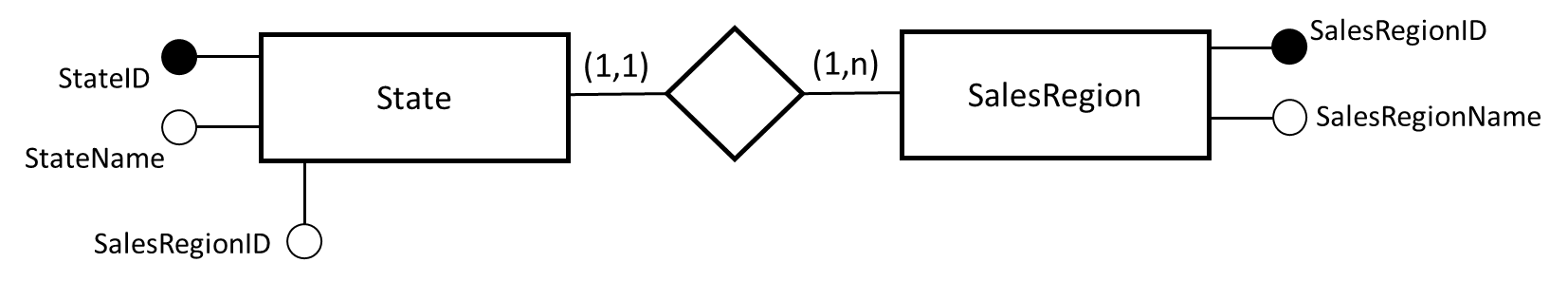
È necessario progettare e implementare fisicamente un database che modelli lo scenario garantendo l’**integrità referenziale** e la **minimizzazione della ridondanza dei dati.**

In altre parole, progetta opportunamente un numero di tabelle e di relazioni tra queste sufficiente a garantire la **consistenza del dato.**

**Task 1: Proponi una progettazione concettuale e logica della base dati**

La progettazione concettuale deve includere tutte le entità coinvolte e le relazioni tra queste. Per ciascuna entità indica l’attributo chiave e i principali attributi descrittivi (non è necessario indicare tutti gli attributi).

*Esempio di schema E/R*



*Lo schema proposto è puramente esemplificativo e non esaustivo o completo per la soluzione!*

La progettazione logica deve includere, per ciascuna tabella, tutte le colonne che poi verranno implementate fisicamente e deve esplicitare la cardinalità dei campi utilizzati per definire la relazione.

*Esempio di schema grafico delle tabelle e delle relazioni tra le stesse.*



n

1



*Il diagramma è puramente esemplificativo e non esaustivo o completo per la soluzione!*

**Task 2: Descrivi la struttura delle tabelle che reputi utili e sufficienti a modellare lo scenario proposto tramite la sintassi DDL. Implementa fisicamente le tabelle utilizzando il DBMS SQL Server(o altro).**

CREATE TABLE table\_name (

column1 datatype option,

column2 datatype option,

column3 datatype,

....

);

**Task 3: Popola le tabelle utilizzando dati a tua discrezione (sono sufficienti pochi record per tabella; riporta le query utilizzate)**

**Task 4: Dopo aver popolate le tabelle, scrivi delle query utili a:**

1. Verificare che i campi definiti come PK siano univoci. In altre parole, scrivi una query per determinare l’univocità dei valori di ciascuna PK (una query per tabella implementata).
2. Esporre l’elenco delle transazioni indicando nel result set il codice documento, la data, il nome del prodotto, la categoria del prodotto, il nome dello stato, il nome della regione di vendita e un campo booleano valorizzato in base alla condizione che siano passati più di 180 giorni dalla data vendita o meno (>180 -> True, <= 180 -> False)
3. Esporre l’elenco dei prodotti che hanno venduto, in totale, una quantità maggiore della media delle vendite realizzate nell’ultimo anno censito. (ogni valore della condizione deve risultare da una query e non deve essere inserito a mano). Nel result set devono comparire solo il codice prodotto e il totale venduto.
4. Esporre l’elenco dei soli prodotti venduti e per ognuno di questi il fatturato totale per anno.
5. Esporre il fatturato totale per stato per anno. Ordina il risultato per data e per fatturato decrescente.
6. Rispondere alla seguente domanda: qual è la categoria di articoli maggiormente richiesta dal mercato?
7. Rispondere alla seguente domanda: quali sono i prodotti invenduti? Proponi due approcci risolutivi differenti.
8. Creare una vista sui prodotti in modo tale da esporre una “versione denormalizzata” delle informazioni utili (codice prodotto, nome prodotto, nome categoria)
9. Creare una vista per le informazioni geografiche

SPIEGAZIONE

MODELLO E/R

Immagine che contiene testo, schermata, Carattere

Il contenuto generato dall'IA potrebbe non essere corretto.

Spiegazione database

ll database che ho creato si chiama ToysGroup ed è nato per gestire e analizzare le vendite di prodotti di una società di giocattoli. Contiene informazioni relative ai prodotti, alle vendite e alle regioni di vendita.

**Tabelle principali:**

1. **Sales**:
   * Contiene 20 righe di dati.
   * Non include il prezzo e la quantità perché si assume che ogni ordine si riferisca a un singolo pezzo del prodotto dove il prezzo è specificato in quest’ultima tabella.
   * Campi principali:
     + IDSales (Identificatore univoco della vendita PK)
     + Order\_Number (Numero dell’ordine)
     + Order\_Date (Data dell’ordine)
     + IDProduct (Collegamento alla tabella Product FK)
     + IDRegion (Collegamento alla tabella Region FK)
2. **Product**:
   * Contiene 20 righe di dati.
   * Campi principali:
     + IDProduct (Identificatore univoco del prodotto PK)
     + Product\_Name (Nome del prodotto)
     + Category (Categoria del prodotto)
     + Price (Prezzo del prodotto)
3. **Region**:
   * Contiene 10 righe di dati.
   * Campi principali:
     + IDRegion (Identificatore univoco della regione PK)
     + State (Nome dello stato)
     + Area (Area geografica)

**Relazioni tra le tabelle:**

* La tabella **Sales** è collegata alla tabella **Product** tramite la chiave **ProductID**.
* La tabella **Sales** è collegata alla tabella **Region** tramite il la chiave **RegionID**.