آیا حوزه تعریف در این زبان ایستا است یا پویا؟ با ذکر مثال توضیح دهید.

حوزه تعریف در زبان SQL یک زیرزبان از SQL است که برای ایجاد، تغییر و حذف ساختارهای داده مانند جداول، ستونها، کلیدها، اندیسها و محدودیتها در پایگاه داده استفاده می شود. این زیرزبان شامل دستوراتی مانند ALTER ،CREATE و DROP می شود که به ترتیب برای ساخت، تغییر و حذف ساختارهای داده مورد نظر استفاده می شوند. حوزه تعریف در زبان SQL هم می تواند ایستا باشد و هم پویا. ایستا بودن به این معناست که ساختارهای داده پس از ایجاد، تغییر یا حذف نمی کنند و برای تغییر آنها باید دستورات جدیدی اجرا شود. پویا بودن به این معناست که ساختارهای داده می تواند بر اساس شرایط خاصی که در دستورات تعریف شده اند، تغییر کنند. به عنوان مثال، می توان یک جدول را به گونهای تعریف کرد که هر بار که یک رکورد به آن اضافه می شود، یک ستون جدید هم به آن اضافه شود. این یک مثال از حوزه تعریف پویا در زبان SQL است. برای این کار می توان از دستور زیر استفاده کرد:

```
CREATE TABLE test ( id INT PRIMARY KEY, name VARCHAR(50));
```

CREATE TRIGGER add_column AFTER INSERT ON test FOR EACH ROW BEGIN

```
DECLARE new_column VARCHAR(50);
SET new_column = CONCAT('col', NEW.id);
SET @sql = CONCAT('ALTER TABLE test ADD COLUMN ', new_column, 'VARCHAR(50)');
PREPARE stmt FROM @sql;
EXECUTE stmt;
DEALLOCATE PREPARE stmt;
END;
```

این دستور یک جدول به نام test با دو ستون id و name ایجاد می کند. سپس یک add_column به نام trigger تعریف می کند که بعد از هر INSERT بر روی جدول test اجرا می شود. این trigger یک متغیر به نام new_column تعریف می کند و مقدار آن را برابر با یک رشته که شامل عبارت col و مقدار آن را برابر با یک دستور ALTER سپس یک متغیر دیگر به نام @sql تعریف می کند و مقدار آن را برابر با یک دستور TABLE که برای اضافه کردن یک ستون جدید با نام new_column به جدول test و EXECUTE ،PREPARE با استفاده از دستورات PREPARE و EXECUTE ،PREPARE اجرا می کند. به این ترتیب، هر بار که یک رکورد جدید به جدول test اضافه می شود، یک ستون جدید هم با نام متناسب با id آن رکورد به جدول اضافه می شود.

در صورتی که زبان حوزه تعریف ایستا را پشتیبانی می کند و بخواهیم امکان استفاده از حوزه تعریف پویا به آن بیافزاییم، چه تغییراتی در زبان باید ایجاد کنیم؟

برای اضافه کردن امکان استفاده از حوزه تعریف پویا به زبان SQL، باید چندین تغییر در زبان ایجاد کنیم. برخی از این تغییرات عبارتند از:

❖ اضافه کردن یک کلمه کلیدی جدید به نام DYNAMIC به زبان SQL که برای تعیین نوع حوزه تعریف استفاده می شود. به عنوان مثال، برای ایجاد یک جدول با حوزه تعریف پویا، می توان از دستور زیر استفاده کرد:

```
CREATE DYNAMIC TABLE test (
id INT PRIMARY KEY,
name VARCHAR(50)
);
```

❖ اضافه کردن یک ساختار جدید به نام RULE به زبان SQL که برای تعریف قواعدی که برای تغییر ساختار داده بر اساس شرایط خاصی اعمال میشوند، استفاده میشود. به عنوان مثال، برای اضافه کردن یک ستون جدید به جدول test هر بار که یک رکورد جدید اضافه میشود، میتوان از دستور زیر استفاده کرد:

```
CREATE RULE add_column ON test

AFTER INSERT

FOR EACH ROW

BEGIN

DECLARE new_column VARCHAR(50);

SET new_column = CONCAT('col', NEW.id);

ALTER TABLE test ADD COLUMN new_column VARCHAR(50);

END;
```

- ❖ اضافه کردن یک کلمه کلیدی جدید به نام NEW به زبان SQL که برای ارجاع به رکورد جدیدی که به جدول اضافه شده است، استفاده می شود. به عنوان مثال، در دستور بالا، NEW.id به معنای مقدار id رکورد جدید است.
- اضافه کردن یک کلمه کلیدی جدید به نام OLD به زبان SQL که برای ارجاع به رکورد قبلی که از جدول حذف شده است، استفاده می شود. به عنوان مثال، برای حذف یک ستون از جدول test هر بار که یک رکورد از آن حذف می شود، می توان از دستور زیر استفاده کرد:

CREATE RULE drop_column ON test

AFTER DELETE

FOR EACH ROW

```
BEGIN
```

```
DECLARE old_column VARCHAR(50);
SET old_column = CONCAT('col', OLD.id);
ALTER TABLE test DROP COLUMN old_column;
END;
```

اینها فقط برخی از تغییرات ممکن برای اضافه کردن امکان استفاده از حوزه تعریف پویا به زبان SQL هستند و ممکن است راهحلهای دیگری هم وجود داشته باشند.

پس از تغییر زبان، قطعه کدی نوشته شود که توسط آن حوزهٔ تعریف پویا استفاده شود. همچنین اگر زبان هر دو حوزهٔ تعریف را پشتبانی می کند، هر دو مورد توصیف شوند.

برای استفاده از حوزه تعریف پویا در زبان SQL، باید از کلمه کلیدی DYNAMIC و ساختار RULE استفاده کنیم. به عنوان مثال، قطعه کد زیر یک جدول با حوزه تعریف پویا ایجاد می کند که هر بار که یک رکورد جدید به آن اضافه می شود، یک ستون جدید هم به آن اضافه می شود:

```
CREATE DYNAMIC TABLE test (
  id INT PRIMARY KEY,
  name VARCHAR(50)
);
CREATE RULE add_column ON test
AFTER INSERT
FOR EACH ROW
BEGIN
  DECLARE new_column VARCHAR(50);
  SET new_column = CONCAT('col', NEW.id);
  ALTER TABLE test ADD COLUMN new_column VARCHAR(50);
END;
```

اگر زبان SQL هر دو حوزه تعریف ایستا و پویا را پشتیبانی کند، می توانیم با استفاده از کلمه کلیدی STATIC یک جدول با حوزه تعریف ایستا ایجاد کنیم. به عنوان مثال، قطعه کد زیر یک جدول با حوزه تعریف ایستا ایجاد می کند که ساختار آن پس از ایجاد، تغییر نمی کند:

```
CREATE STATIC TABLE test (
  id INT PRIMARY KEY,
  name VARCHAR(50)
);
```

بلوک ها در این زبان چگونه تعریف شده اند؟ آیا کلمات کلیدی ویژه ای برای اعمال تغییر در حوزهٔ تعریف متغیرها وجود دارند؟

بلوک ها در زبان SQL به عنوان یک واحد منطقی از دستورات تعریف شده اند که می توانند در یک تراکنش یا یک برنامه اجرا شوند. بلوک ها می توانند شامل متغیرها، ثابت ها، مقادیر پیش فرض، توابع، زیر برنامه ها، دستورات کنترل جریان و خطاها باشند. بلوک ها می توانند درون یکدیگر تودرتو شوند و محدوده متغیرها را تعیین کنند. برای شروع و پایان یک بلوک، از کلمات کلیدی BEGIN و END استفاده می شود. برای مثال، بلوک زیر یک متغیر به نام x را تعریف می کند و مقدار آن را به x تغییر می دهد:

```
BEGIN

DECLARE x INT DEFAULT 0;

SET x = 10;
END
```

برای اعمال تغییر در حوزه تعریف متغیرها، می توان از کلمات کلیدی مختلفی مانند GLOBAL و LOCAL ،DEFAULT ،SET ،DECLARE

:DECLARE ✓

این کلمه کلیدی برای تعریف یک متغیر در SQL استفاده می شود. برای تعریف یک متغیر، باید نام و نوع آن را مشخص کنید. مثلا:

DECLARE @x INT; -- Define an integer value named x

:SET ✓

این کلمه کلیدی برای اختصاص یا تغییر مقدار یک متغیر در SQL استفاده می شود. برای اختصاص یا تغییر مقدار یک متغیر، باید نام و مقدار جدید آن را مشخص کنید. مثلا:

SET @x = 10; -- Assign the value of 10 to variable x

:DEFAULT ✓

این کلمه کلیدی برای تعیین یک مقدار پیش فرض برای یک ستون در SQL استفاده می شود. مقدار پیش فرض برای یک ستون، آن مقداری است که در صورت عدم ورود مقدار توسط کاربر، به آن ستون اختصاص داده می شود. مثلا:

CREATE TABLE Users (
UserID INT NOT NULL,
UserName NVARCHAR(50) NOT NULL,
Email NVARCHAR(100) DEFAULT 'example@example.com' -- Set the default value for the email column
);

:LOCAL ✓

این کلمه کلیدی برای مشخص کردن حوزه یک متغیر در SQL استفاده می شود. یک متغیر محلی، آن متغیری است که فقط در بلوکی که تعریف شده است، قابل دسترسی است. برای تعریف یک متغیر محلی، باید قبل از نام آن یک علامت @ قرار دهید. مثلا:

```
BEGIN

DECLARE @x INT; -- Define a local variable named x

SET @x = 10;

PRINT @x; -- Print x

END
```

:GLOBAL ✓

این کلمه کلیدی برای مشخص کردن حوزه یک متغیر در SQL استفاده می شود. یک متغیر جهانی، آن متغیری است که در همه بلوک های یک اتصال، قابل دسترسی است. برای تعریف یک متغیر جهانی، باید قبل از نام آن دو علامت @@ قرار دهید. مثلا:

```
DECLARE @@x INT; -- Define a global variable named x
SET @@x = 10;
BEGIN
   PRINT @@x; --Print x
END
```

منابع:

- https://ostovaee.ir
- https://www.w3schools.com/sql
- https://www.roxo.ir/series
- https://blog.faradars.org
- https://www.mongard.ir
- https://sariasan.com
- https://liangroup.net
- https://sitedesign-co.com
- https://sabzdanesh.com
- https://maktabkhooneh.org/learn/database/