

في عالم الـ SQL، كتير بنحتاج نخلي الـ Stored Procedure ترجع لنا أكتر من قيمة — مش بس result set عادي. هنا بييجي دور OUTPUT parameters بدل ما تعمل PotaTable وتضطر تتعامل مع DataTable أو Cursor، ممكن تستخدم OUTPUT variable يرجعلك القيم المطلوبة Orderl أو application. مثال شائع؟ لما تعمل insert في جدول OrderlD وتاخد OUTPUT parameter. It's clean, fast, and easy الجديد باستخدام SCOPE_IDENTITY (داخل Fython أو حتى Power Bl. بس Power Bl. بوانن stored procedures المقال المقال المقال واضح في stored procedures الحين المقال المقال

Input and Output Parameter for





A) CREATE PROCEDURE GetStudentCountPerDepartment

AS

BEGIN

SELECT D.DNAME AS DepartmentName, COUNT(S.ID) AS StudentCount

FROM Student S

JOIN Department D ON S.DNO = D.DNUMBER

GROUP BY D.DNAME;

END;

B) CREATE PROCEDURE CheckEmployeesInP1

AS

BEGIN

DECLARE @EmpCount INT;

```
-- Get number of employees in project p1
 SELECT @EmpCount = COUNT(DISTINCT E.SSN)
 FROM EMPLOYEE E
 JOIN WORKS_ON W ON E.SSN = W.ESSN
 JOIN PROJECT P ON W.PNO = P.PNUMBER
 WHERE P.PNAME = 'p1';
 -- Decision based on count
 IF @EmpCount >= 3
 BEGIN
   PRINT 'The number of employees in the project p1 is 3 or more';
 END
 FLSE
 BEGIN
   PRINT 'The following employees work for the project p1:';
   SELECT E.FNAME, E.LNAME
   FROM EMPLOYEE E
   JOIN WORKS_ON W ON E.SSN = W.ESSN
   JOIN PROJECT P ON W.PNO = P.PNUMBER
   WHERE P.PNAME = 'p1';
 END
END;
C) CREATE PROCEDURE ReplaceEmployeeOnProject
 @OldEmp INT,
 @NewEmp INT,
 @ProjNo INT
AS
BEGIN
 DECLARE @Hours FLOAT;
```

```
التحقق إذا الموظف القديم كان شغال فعلاً على المشروع --
IF EXISTS (
 SELECT * FROM WORKS_ON
 WHERE ESSN = @OldEmp AND PNO = @ProjNo
)
BEGIN
 الحصول على عدد الساعات اللي كان بيشتغلها الموظف القديم --
 SELECT @Hours = HOURS
 FROM WORKS_ON
 WHERE ESSN = @OldEmp AND PNO = @ProjNo;
 حذف الربط بين الموظف القديم والمشروع --
  DELETE FROM WORKS_ON
 WHERE ESSN = @OldEmp AND PNO = @ProjNo;
 إضافة الموظف الجديد بنفس عدد الساعات (لو موجود بالفعل ما نضيفوش مرتين) --
 IF NOT EXISTS (
   SELECT * FROM WORKS_ON
   WHERE ESSN = @NewEmp AND PNO = @ProjNo
 )
  BEGIN
   INSERT INTO WORKS_ON (ESSN, PNO, HOURS)
   VALUES (@NewEmp, @ProjNo, @Hours);
  END
  ELSE
  BEGIN
   PRINT 'The new employee is already assigned to this project.';
 END
END
ELSE
```

BEGIN

PRINT 'The old employee is not assigned to this project.';

END

END;

Part 02

Transactional Backup و Differential الفرق بيول

- Full Backup
 - . هو نسخة كاملة من قاعدة البيانات بكل محتوياتها (جداول، بيانات، إجراءات...) ◆
 - يستخدم لإنشاء نقطة استرجاع شاملة 🗸
- Differential Backup
 - تخريرات التي حصلت منذ آخر Full Backup.
 - V الكنه لا يمكن استخدامه وحده يحتاج إلى Full Backup أسرع من Full Backup.
- Transaction Log Backup
 - Transaction Log Backup. التي حدثت منذ آخر (transactions) ينسخ كل العمليات
 - . خصوصًا في أنظمة تعتمد على الاستمر اريةpoint-in-time recoveryيُستخدم لتحقيق

الاستخدام الأمثل؟ 奏

- Full Backup
- Differential يومي
- Transactional كل ساعة أو أقل حسب أهمية البيانات

وعلى أي مستوى تُستخدم؟ DENY و GRANT ؟ وما الفرق بينPermission ما هو 2

- Permission (الصلاحية)
 - .، مثل قراءة جدول، تعديل بيانات، أو تنفيذ إجراءSOL Server هي القدرة على تنفيذ إجراء معين داخل
- GRANT
 - تعنى "منح" صلاحية 🗸
 - ... SELECT ، INSERT تستخدم للسماح للمستخدم بتنفيذ عملية مثل 🧠
- DENY
 - .GRANT تعنى "رفض الصلاحية" بشكل صريح وتتفوق على 😑
 - يتفوق DENY لفرد، الـ DENY إذا تم منح صلاحية لمجموعة، ولكن 🖈

على أي مستوى؟ 🔆

- Permissions تُستخدم على مستويات مختلفة
 - o Server-level (مثل: إنشاء قواعد بيانات)
 - o **Database-level** (مثل: تنفیذ SPs)
 - o Object-level (على جدول معين SELECT :مثل)

ومتى نستخدمه؟ SQL Profiler ما هر 3

- SQL Server Profiler
 - في الوقت الحقيقي SQL Server تُستخدم لمر اقبة وتحليل ما يحدث داخل GUI هو أداة

:الاستخدامات الشائعة 📠

- (Query Performance) تتبع أداء الاستعلامات
- تحليل الأحداث غير المتوقعة •
- تتبع العمليات أو الإجراءات المُنفذة من التطبيقات
- كشف مشاكل القفل (deadlocks)
- يُفضل استخدامه بحذر على بيئة الإنتاج لأنه قد يؤثر على الأداء :ملاحظة 🛕

?Stored Procedure ؟ ولماذا نستخدمه؟ وعلى أي مستوى؟ وما الفرق بينه وبينTrigger ما هو 4

- Trigger (المشغّل)
 - تا الله مثل في الله الله الله الله عند حدوث على جدول أو قاعدة بيانات، مثل INSERT ، UPDATE ، DELETE.
- لماذا نستخدمه؟
 - تلقائي أو لعمل عمليات مترابطة (Audit) أو لعمل تدقيق (Business Rules) لفرض قواعد تجارية 🔽
- :أنواعه
 - o DML Trigger (للجداول) ➤ يعمل بعد أو قبل العمليات مثل Insert / Update / Delete
 - o DDL Trigger یعمل عند عملیات مثل CREATE TABLE ، DROP....
 - o Logon Trigger ➤ يعمل عند محاولة تسجيل الدخول

PART3



Certificate of Accomplishment



PRESENTED TO

Omar Tarek

The bearer of this certificate has passed the HackerRank skill certification test

Earned on: 20 Jul, 2025

ID: 94912C3309B1

Harishankaran K

CTO, HackerRank