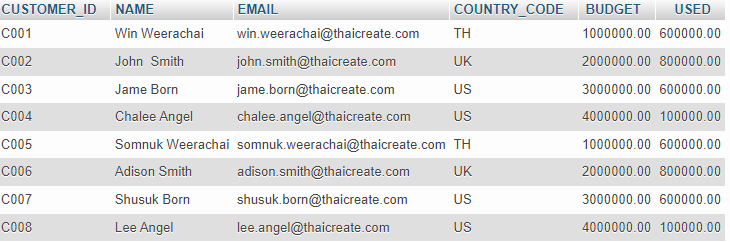
**ปฏิบัติการที่ 6 Trigger on Database**

**วัตถุประสงค์**

* เพื่อให้นิสิตเข้าใจ และประยุกต์ใช้ Trigger

ตารางข้อมูลที่ใช้ประกอบ

**1. ตาราง Customer**



**2. ตาราง Audit**



1. จงสร้าง Trigger โดยมีเงื่อนไขดังนี้

1. จงสร้าง Trigger ชื่อ delAudit
2. เมื่อลบข้อมูลจากตาราง Customer ให้ทำการลบข้อมูลในตาราง Audit ตามด้วย
3. เมื่อข้อมูลจากตาราง Customer ถูกลบออกแล้ว**จึงค่อยลบข้อมูล**ในตาราง Audit

**จงเติมผลลัพธ์การสร้างลงในตารางด้านล่าง**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Condition** | **ผลลัพธ์** | | | |
| **Code** Triggerชื่อ delAudit | DROP TRIGGER IF EXISTS delAudit;  DELIMITER //  CREATE TRIGGER delAudit AFTER DELETE ON customer      FOR EACH ROW  BEGIN      DELETE FROM `audit` WHERE CUSTOMER\_ID = old.CUSTOMER\_ID;  END //  DELIMITER ; | | | |
| **เรียกใช้** Trigger ชื่อ delAudit เงื่อนไขจงลบข้อมูล C001 ในตาราง Customer | **ภาพหน้าจอผลลัพธ์ตาราง customer หลังลบข้อมูล** C001 | | | |
|  | | | |
|  | **ภาพหน้าจอผลลัพธ์ตาราง Audit หลังลบข้อมูล C001** | | | |
|  | | | |
| **การทดลองที่** | | **คะแนนเต็ม** | **คะแนนที่ได้** | **ผู้ตรวจ** |
| 1 | | 2 |  |  |

2. จงสร้าง Trigger โดยมีเงื่อนไขดังนี้

1. จงสร้าง Trigger ชื่อ sumUsed
2. เมื่อเพิ่มข้อมูลในตาราง Audit ให้ทำการ**คำนวนผลรวมขอเงินที่ใช้ใหม่**พร้อมแก่ไขข้อมูลในตาราง Customer ใหม่อีกครั้ง
3. เมื่อข้อมูลจากตาราง Audit ถูกเพิ่มเข้าไปแล้ว**จึงค่อยแก้ไขข้อมูล**ในตาราง Customer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Condition** | **ผลลัพธ์** | | | |
| **Code**  สร้าง Trigger ชื่อ sumUsed | DROP PROCEDURE IF EXISTS updateCustomerUsedFromAudit;  DELIMITER //  CREATE PROCEDURE updateCustomerUsedFromAudit(IN toUpdateCustomerId varchar(4))  BEGIN      DECLARE dUsed DECIMAL(18, 2) DEFAULT 0;      SELECT COALESCE(SUM(USED), 0) INTO dUsed FROM `audit` WHERE CUSTOMER\_ID = toUpdateCustomerId;      UPDATE customer SET USED = dUsed WHERE CUSTOMER\_ID = toUpdateCustomerId;  END //  DELIMITER ;    DROP TRIGGER IF EXISTS sumUsed;  DELIMITER //  CREATE TRIGGER sumUsed AFTER INSERT ON `audit`      FOR EACH ROW  BEGIN      CALL updateCustomerUsedFromAudit(new.CUSTOMER\_ID);  END //  DELIMITER ; | | | |
| **เรียกใช้** Trigger ชื่อ sumUsed เงื่อนไขจงเพิ่มข้อมูลในตาราง Audit  AUDIT\_ID: 19  CUSTOMER\_ID: C002  LOG\_DATE: 2015-10-10  USED: 200000 | **ภาพหน้าจอผลลัพธ์ตาราง Audit หลังเพิ่มข้อมูล** | | | |
|  | | | |
|  | **ภาพหน้าจอผลลัพธ์ตาราง Customer หลังเพิ่มข้อมูล** | | | |
|  | | | |
| **การทดลองที่** | | **คะแนนเต็ม** | **คะแนนที่ได้** | **ผู้ตรวจ** |
| 2 | | 2 |  |  |

3. จงสร้าง Trigger โดยมีเงื่อนไขดังนี้

1. จงสร้าง Trigger ชื่อ sumUsed\_Delete
2. เมื่อลบข้อมูลในตาราง Audit ให้ทำการ**คำนวนผลรวมขอเงินที่ใช้ใหม่**พร้อมแก่ไขข้อมูลในตาราง Customer ใหม่อีกครั้ง
3. เมื่อข้อมูลจากตาราง Audit ถูกลบไปแล้ว**จึงค่อยแก้ไขข้อมูล**ในตาราง Customer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Condition** | **ผลลัพธ์** | | | |
| **Code** สร้าง Trigger ชื่อsumUsed\_Delete | DROP PROCEDURE IF EXISTS updateCustomerUsedFromAudit;  DELIMITER //  CREATE PROCEDURE updateCustomerUsedFromAudit(IN toUpdateCustomerId varchar(4))  BEGIN      DECLARE dUsed DECIMAL(18, 2) DEFAULT 0;      SELECT COALESCE(SUM(USED), 0) INTO dUsed FROM `audit` WHERE CUSTOMER\_ID = toUpdateCustomerId;      UPDATE customer SET USED = dUsed WHERE CUSTOMER\_ID = toUpdateCustomerId;  END //  DELIMITER ;    DROP TRIGGER IF EXISTS sumUsed\_Delete;  DELIMITER //  CREATE TRIGGER sumUsed\_Delete AFTER DELETE ON `audit`      FOR EACH ROW  BEGIN      CALL updateCustomerUsedFromAudit(old.CUSTOMER\_ID);  END //  DELIMITER ; | | | |
| **เรียกใช้** Trigger ชื่อ sumUsed\_Delete เงื่อนไขจงลบข้อมูลในตาราง audit  CUSTOMER\_ID: C003  USED: 200000 | **ภาพหน้าจอผลลัพธ์ตาราง Audit หลังลบข้อมูล** | | | |
|  | | | |
|  | **ภาพหน้าจอผลลัพธ์ตาราง Customer หลังลบข้อมูล** | | | |
|  | | | |
| **การทดลองที่** | | **คะแนนเต็ม** | **คะแนนที่ได้** | **ผู้ตรวจ** |
| 3 | | 3 |  |  |

4. จงเรียกใช้ Trigger โดยมีเงื่อนไขดังนี้

1. จงสร้าง Trigger ชื่อ editAudit
2. เมื่อแก้ไขข้อมูลในตาราง Audit ให้ทำการ**คำนวนผลรวมขอเงินที่ใช้ใหม่**พร้อมแก่ไขข้อมูลในตาราง Customer ใหม่อีกครั้ง
3. เมื่อข้อมูลจากตาราง Audit ถูกแก้ไขแล้ว**จึงค่อยแก้ไขข้อมูล**ในตาราง Customer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Condition** | **ผลลัพธ์** | | | |
| **Code** สร้าง Trigger ชื่อ editAudit | DROP PROCEDURE IF EXISTS updateCustomerUsedFromAudit;  DELIMITER //  CREATE PROCEDURE updateCustomerUsedFromAudit(IN toUpdateCustomerId varchar(4))  BEGIN      DECLARE dUsed DECIMAL(18, 2) DEFAULT 0;      SELECT COALESCE(SUM(USED), 0) INTO dUsed FROM `audit` WHERE CUSTOMER\_ID = toUpdateCustomerId;      UPDATE customer SET USED = dUsed WHERE CUSTOMER\_ID = toUpdateCustomerId;  END //  DELIMITER ;  DROP TRIGGER IF EXISTS editAudit;  DELIMITER //  CREATE TRIGGER editAudit AFTER UPDATE ON `audit`      FOR EACH ROW  BEGIN      IF new.USED <> old.USED THEN          CALL updateCustomerUsedFromAudit(old.CUSTOMER\_ID);      END IF;  END // | | | |
| **เรียกใช้** Trigger ชื่อ editAudit เงื่อนไขจงแก้ไขข้อมูลในตาราง audit  AUDIT\_ID: 7  CUSTOMER\_ID: C003  USED: 100000 | **ภาพหน้าจอผลลัพธ์ตาราง Audit หลังลบข้อมูล** | | | |
|  | | | |
|  | **ภาพหน้าจอผลลัพธ์ตาราง customer หลังลบข้อมูล** | | | |
|  | | | |
| **การทดลองที่** | | **คะแนนเต็ม** | **คะแนนที่ได้** | **ผู้ตรวจ** |
| 4 | | 3 |  |  |