

## Esercizi Trigger: Gestione delle carte fedeltà

Sono date le relazioni seguenti (le chiavi primarie sono sottolineate, gli attributi opzionali sono indicati con \*).

PRODOTTO (CodP, NomeP, Prezzo, PuntiProdotto)  
CARTA\_FEDELTA' (CodC, NomeCliente, PuntiTotali)  
PREMIO (CodPremio, DescrizPremio, PuntiNecessari)  
ACQUISTO (CodA, CodP, Data, CodC\*, NumeroPezzi)  
RICHIESTA\_NOTIFICA (CodN, CodC, CodPremio, DescrizPremio)

Si vogliono gestire alcune attività relative alle carte fedeltà di un supermercato. La tabella CARTA\_FEDELTA' riporta, per ogni carta fedeltà, i punti complessivamente acquisiti da un cliente (attributo PuntiTotali). La tabella PREMIO descrive i premi disponibili. Per ciascun premio, il valore in punti è definito dall'attributo PuntiNecessari. Si scrivano i trigger per gestire le seguenti attività.

1) *Attribuzione dei punti per un acquisto ed eventuale selezione del premio.* Si scriva il trigger per aggiornare la situazione della carta fedeltà del cliente che ha eseguito l'acquisto. Quando viene eseguito un nuovo acquisto (inserimento nella tabella ACQUISTO), devono essere aggiornati i punti totali conseguiti dal cliente. I punti indicati nell'attributo PuntiProdotto sono relativi all'acquisto di un solo pezzo. Per calcolare i punti totali conseguiti nell'acquisto occorre considerare il numero totale di pezzi acquistati (attributo NumeroPezzi). Quando il prodotto non è associato all'acquisizione di punti, l'attributo PuntiProdotto vale zero. Si noti che gli acquisti non sono necessariamente associati a una carta fedeltà. Quando l'acquisto non è associato a una carta fedeltà, il valore dell'attributo CodC è NULL e non deve essere eseguito alcun aggiornamento della tabella CARTA\_FEDELTA'.

Successivamente, si deve verificare se i punti accumulati dal cliente che ha effettuato l'acquisto consentono di ottenere un premio, ossia se esiste almeno un premio di valore (attributo PuntiNecessari) inferiore o uguale ai punti totali accumulati dal cliente. In caso positivo, deve essere scelto il premio di valore massimo che può essere ricevuto con i punti accumulati. Si supponga che ci sia al più un premio che soddisfi questa condizione.

Si deve infine richiedere la notifica (inserimento nella tabella RICHIESTA\_NOTIFICA) della possibilità di ricevere il premio selezionato al possessore della carta fedeltà. La chiave primaria CodN è un contatore che deve essere incrementato ogni volta che è inserita una nuova notifica (si tenga conto che le notifiche completamente evase potrebbero essere eliminate dalla base di dati).

2) *Vincolo di integrità sul valore massimo in punti di un prodotto.* Per ogni prodotto, il valore in punti (attributo PuntiProdotto nella tabella PRODOTTO) non può essere superiore a 1/10 del prezzo di vendita del prodotto (attributo Prezzo). Si scriva il trigger che gestisce il vincolo d'integrità, assegnando il valore massimo, pari a 1/10 del prezzo, quando il valore massimo consentito viene superato.

**Trigger per Attribuzione dei punti per un acquisto ed eventuale selezione del premio.**

```

CREATE TRIGGER AggiornaPunti
AFTER INSERT ON ACQUISTO
FOR EACH ROW
WHEN (NEW.CodC IS NOT NULL) ---- verifico che il cliente abbia la carta fedeltà
DECLARE
    myDescrizPremio CHAR(10);
    N NUMBER;
    P NUMBER;
    Z NUMBER;
    PCode NUMBER;
BEGIN

--leggo i punti per il prodotto
SELECT PuntiProdotto INTO P
FROM PRODOTTO
WHERE CodP = :NEW.CodP;

---se al prodotto non sono associati punti (ossia P=0), le successive operazioni non sono eseguite
IF (P <> 0) THEN
----aggiorno la carta fedeltà
    UPDATE CARTA_FEDELTA'
    SET PuntiTotali = PuntiTotali + P * :NEW.NumeroPezzi
    WHERE CodC = :NEW.CodC;

--leggo il premio associato al massimo numero di punti che si il cliente può ottenere con i punti
--accumulati fino ad ora
SELECT MAX(PuntiNecessari) INTO Z
FROM PREMIO
WHERE PuntiNecessari <= (SELECT PuntiTotali
                        FROM CARTA_FEDELTA'
                        WHERE CodC = :NEW.CodC);

----Z è uguale a NULL se non è presente nessun premio che richiede un numero di punti (PuntiNecessari)
minore o uguale ai punti accumulati dal cliente (PuntiTotali); altrimenti Z è diverso da NULL

IF (Z IS NOT NULL) THEN
----seleziono il premio che richiede un numero di punti pari al valore in Z
----dalla specifica dell'esercizio si assume che ci sia un solo premio che verifica questa condizione
    SELECT CodPremio, DescPremio INTO PCode, myDescPremio
    FROM PREMIO
    WHERE PuntiNecessari = Z;

```

```
--leggo il valore massimo di CodN
SELECT MAX(CodN) INTO N
FROM RICHIESTA_NOTIFICA;

IF (N IS NULL) THEN
    N := 0;
END IF;

--inserisco la nuova notifica
INSERT INTO RICHIESTA_NOTIFICA(CodN, CodC, CodPremio, DescrizPremio)
VALUES (N+1, :NEW.Cod, PCod, myDescrizPremio);
END IF;
END IF;
END;
```

**Vincolo di integrità sul valore massimo in punti di un prodotto.**

```
CREATE TRIGGER checkProductPoints
BEFORE INSERT OR UPDATE OF PuntiProdotto, Prezzo ON PRODOTTO
FOR EACH ROW
WHEN (NEW.PuntiProdotto > NEW.Prezzo/10)
BEGIN
    :NEW.PuntiProdotto := :NEW.Prezzo /10;
END;
```