

## Definizione di vincoli tramite l'uso del comando SQL CREATE ASSERTION

Quando si vogliono specificare dei vincoli di integrità in SQL è possibile utilizzare il comando "CREATE ASSERTION".

La sintassi del comando è la seguente

```
CREATE ASSERTION <name> CHECK(<condition>);
```

"Condition" contiene la condizione che deve essere sempre vera affinché il vincolo sia verificato.

Quando si modifica l'istanza della base di dati inserendo, modificando o cancellando dei dati il sistema verifica in automatico se i vincoli specificati tramite l'assertion sono verificati oppure no. Se l'operazione che si sta eseguendo porta ad una situazione di violazione dei vincoli specificati l'operazione viene vietata dal sistema.

Seguono alcuni esempi/esercizi.

### 1. Espressione di vincoli

Si consideri il seguente schema di base di dati che vuole tenere traccia delle informazioni riguardanti i campionati mondiali di atletica.

```
CAMPIONATO (EdizioneCamp, Anno, DataInizio, DataFine, CittàOspitante, StatoOspitante)
GARA (EdizioneCamp, NumGara, Disciplina, Data, OraInizio)
ATLETA (CodiceAtleta, Nome, Cognome, Sesso, DataNascita, Nazionalità)
CLASSIFICA_GARA (EdizioneCamp, NumGara, CodiceAtleta, Posizione)
```

Dove il campo Posizione è un intero.

Considerando lo schema di base di dati fornito, specificare in SQL il vincolo che controlla che alle gare della disciplina '110 metri ostacoli' abbiano effettivamente partecipato solo atleti di sesso maschile.

```
CREATE ASSERTION ControlloGare110
CHECK (NOT EXISTS (SELECT *
                    FROM CLASSIFICA_GARA CG, GARA G, ATLETA A
                    WHERE GC.CodiceAtleta=A.CodiceAtleta
                          AND GC.EdizioneCamp=G.EdizioneCamp
                          AND GC.NumGara=G.NumGara
                          AND G.Disciplina='110 metri ostacoli'
                          AND A.Sesso='F'
                    )
);
```

### 2. Espressione di vincoli

Si consideri il seguente schema di base di dati che vuole tenere traccia delle informazioni riguardanti docenti e studenti.

```
PERSONA(CodFiscale, Nome, Cognome, CodDipartimento, AnnoIscrizione, TipoPersona)
DIPARTIMENTO(CodDipartimento, NomeDipartimento)
```

Il campo TipoPersona assume due possibili valori: 'Docente' e 'Studente'.

Per i docenti CodDipartimento contiene il loro dipartimento di appartenenza mentre AnnoIscrizione è impostato a NULL.

Per gli studenti CodDipartimento è impostato a NULL mentre AnnoIscrizione contiene l'anno in cui lo studente si è iscritto per la prima volta.

Considerando lo schema di base di dati fornito, specificare in SQL i vincoli che controllano che per docenti e studenti siano verificate le condizioni indicate nelle specifiche.

```
CREATE ASSERTION VincoliDocente
CHECK (NOT EXISTS (SELECT *
                   FROM PERSONA
                   WHERE TipoPersona='Docente'
                      AND (CodDipartimento IS NULL OR
                          AnnoIscrizione IS NOT NULL)
                   )
);
```

```
CREATE ASSERTION VincoliStudente
CHECK (NOT EXISTS (SELECT *
                   FROM PERSONA
                   WHERE TipoPersona='Studente'
                      AND (CodDipartimento IS NOT NULL OR
                          AnnoIscrizione IS NULL)
                   )
);
```