

# Esempio SQL su tabella ISCRIZIONI UNIVERSITARIE

[schemi\\_db](#)

## corsi\_laurea

Creiamo all'interno del nostro client `sql` un database che possiamo chiamare come vogliamo:

```
CREATE DATABASE universita;  
USE universita;
```

Al suo interno creiamo una tabella, `corsi_laurea`, che e' la stessa tabella negli esempi SQL visti in precedenza:

### Iscrizioni universitarie

CORSI\_LAUREA(codice, nome, descrizione)  
INSEGNAMENTI(codice, nome, crediti, ssd)  
MANIFESTI(laurea(fk), insegnamento(fk), fondamentale(boolean), anno\_corso)  
STUDENTI (matricola, nome, cognome, data\_nascita)  
ISCRIZIONI(studente(fk), anno\_iscrizione, laurea(fk), data\_iscrizione, anno\_corso)

```
CREATE TABLE corsi_laurea (  
  codice integer NOT NULL PRIMARY KEY,  
  nome VARCHAR(200) NOT NULL,  
  descrizione VARCHAR(200) NOT NULL  
  
  -- alternative  
  -- PRIMARY KEY (codice)  
  -- creiamo i nostri vincoli  
  -- CONSTRAINT codice_non_nullo CHECK(codice IS NOT NULL)  
  -- CONSRAINT codice_pk PRIMARY KEY (codice)  
);
```

La tabella puo' essere modificata con DML `INSERT INTO`:

```
INSERT INTO corsi_laurea (codice, nome, descrizione)  
VALUES (1, 'Informatica', 'Corso di Laurea triennale Informatica');  
  
INSERT INTO corsi_laurea (codice, nome, descrizione)  
VALUES (2, 'Matematica', 'Corso di Laurea triennale Matematica');
```

Possiamo aggiungere specifiche alla tabella, alterandola con `ALTER TABLE`:

```
ALTER TABLE corsi_laurea
ADD UNIQUE(nome);
```

```
-- codice e nome degli insegnamenti disattivati in ordine
-- alfabetico
SELECT codice, nome
FROM insegnamenti
EXCEPT
SELECT codice, nome
FROM insegnamenti I, manifesti M
WHERE I.codice = M.insegnamento
ORDER BY nome ASC;
```

```
SELECT codice, nome
FROM insegnamenti
WHERE codice NOT IN (SELECT insegnamento
                     FROM manifesti)

ORDER BY nome;
```

```
-- codice e nome degli insegnamenti obbligatori
SELECT DISTINCT I.codice, I.nome
FROM insegnamenti I, manifesti M
WHERE I.codice = M.insegnamento
      AND M.fondamentale
-- posizionale sugli attributi selezionati
-- ORDER BY I.nome
ORDER BY 2;
```

```
-- codice e nome degli insegnamenti solo a scelta
SELECT DISTINCT I.codice, I.nome
FROM insegnamenti I, manifesti M
WHERE I.codice = M.insegnamento
      AND NOT M.fondamentale
      AND NOT EXISTS (SELECT *
                     FROM manifesti M2
                     WHERE M2.insegnamento = I.codice
                     AND M2.fondamentale)

ORDER BY I.nome;
```

```
-- iscrizioni "proseguimento" nel 2022
-- creiamo una vista virtuale da usare sotto
CREATE OR REPLACE VIEW proseguimento (codice, cognome, nome) AS
SELECT S.codice, S.cognome, S.nome
FROM studenti S, iscrizione I22, iscrizione I21
WHERE I22.studente = I21.studente
      AND I22.anno_iscrizione = 2022
      AND I21.anno_iscrizione = 2021
      AND I22.laurea = I21.laurea
```

```
AND I22.anno_corso = I21.anno_corso + 1
AND S.matricola = I22.studente
```

```
-- iscrizioni "naturali" nel 2022
```

```
SELECT *
```

```
FROM proseguimenti
```

```
UNION
```

```
select S.*
```

```
FROM studenti S, iscrizione I22
```

```
WHERE I22.anno_iscrizione = 2022
```

```
AND I22.anno_corso = 1
```

```
AND S.matricola = I22.studente
```

```
AND NOT EXISTS (SELECT *
```

```
FROM iscrizione I_OLD
```

```
WHERE I_OLD.studente = S.matricola
```

```
AND I_OLD.anno_iscrizione < 2022)
```

```
ORDER BY cognome, nome;
```