



**UNIMORE**  
UNIVERSITÀ DEGLI STUDI DI  
MODENA E REGGIO EMILIA

Dipartimento di Scienze Fisiche,  
Informatiche e Matematiche

# Basi di Dati

*Corso di Laurea in Informatica*

E' vietata la copia e la riproduzione dei contenuti e immagini in qualsiasi forma. E' inoltre vietata la redistribuzione e la pubblicazione dei contenuti e immagini non autorizzata espressamente dall'autore o dall'Università di Modena e Reggio Emilia.



# Basi di Dati



Esercizi di interrogazioni complesse (I parte)

# DB Esami

---

S (Matr, SNome, Citta, ACorso)

C (CC, CNome, CD)

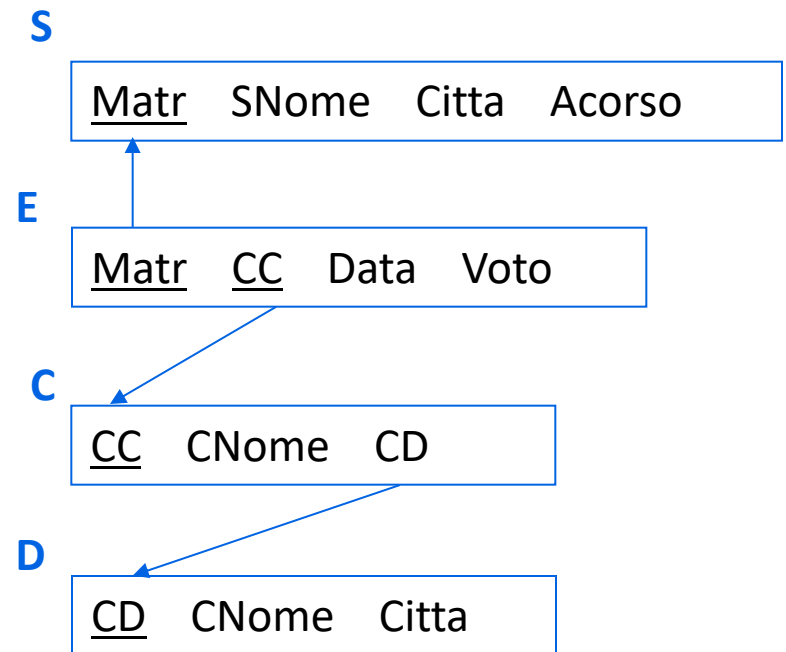
FK: CD REFERENCES D

D (CD, CNome, Citta)

E (Matr, CC, Data, Voto)

FK: Matr REFERENCES S

FK: CC REFERENCES C



# Database Esami – Ordinamento

---

- ▶ Studenti di Modena ordinati in senso ascendente rispetto all'anno di corso

```
SELECT Matr, ACorso  
FROM S  
WHERE Citta='MO'  
ORDER BY ACorso
```

# Database Esami – Ordinamento

---

- ▶ Esami del corso C1 ordinati in senso discendente rispetto al voto espresso in sessantesimi, e a parità di voto rispetto alla matricola

```
SELECT Matr,CC,(60*Voto)/30  
FROM E  
WHERE CC='C1'  
ORDER BY 3 DESC, Matr
```

# DB Esami – Aggregazione e raggruppamento

---

- ▶ Numero di studenti presenti

```
SELECT COUNT(*)  
FROM S
```

# DB Esami – Aggregazione e raggruppamento

---

- ▶ Numero di studenti che hanno sostenuto almeno un esame

```
SELECT COUNT(DISTINCT Matr)  
FROM E
```

# DB Esami – Aggregazione e raggruppamento

---

- ▶ Voto medio degli esami sostenuti dalla matricola M2

```
SELECT AVG(Voto)
FROM E
WHERE Matr='M2'
```

```
SELECT SUM(Voto)/COUNT(Voto)
FROM E
WHERE Matr='M2'
```



# DB Esami – Aggregazione e raggruppamento

---

- ▶ Voto massimo e minimo ottenuto per ogni studente, escludendo il corso C2

```
SELECT Matr,MAX(Voto),MIN(Voto)
FROM E
WHERE CC <> 'C2'
GROUP BY Matr
```

# DB Esami – Aggregazione e raggruppamento

---

- ▶ Voto massimo e minimo ottenuto per ogni studente, visualizzando anche il nome dello studente

```
SELECT E.Matr, S.SNome, MAX(Voto),MIN(Voto)
FROM E, S
WHERE S.MATR = E.MATR
GROUP BY E.Matr, S.Snome
```

# DB Esami – Aggregazione e raggruppamento

---

- ▶ Codice e nome di un corso, e relativo numero di esami sostenuti

```
SELECT C.CC,C.CNome,COUNT(*)  
FROM E,C  
WHERE E.CC=C.CC  
GROUP BY C.CC,C.CNome
```

# DB Esami – Aggregazione e raggruppamento

---

- ▶ Numero di esami per ogni voto compreso tra 22 e 28

```
SELECT Voto, COUNT(*)  
FROM E  
GROUP BY Voto  
HAVING Voto BETWEEN 22 AND 28  
ORDER BY Voto
```

```
SELECT Voto, COUNT(*)  
FROM E  
WHERE Voto BETWEEN 22 AND 28  
GROUP BY Voto  
ORDER BY Voto
```

# DB Esami – Aggregazione e raggruppamento

---

- ▶ Media dei voti per ogni esame sostenuto da più di due studenti

```
SELECT CC,AVG(Voto)
FROM E
GROUP BY CC
HAVING COUNT(*) > 2
```

# DB Esami – Aggregazione e raggruppamento

---

- ▶ Matricola degli studenti che hanno sostenuto almeno due esami con lo stesso voto (uno studente fa parte del risultato se ha preso lo stesso voto almeno due volte)

```
SELECT DISTINCT Matr  
FROM E  
GROUP BY Matr,Voto  
HAVING COUNT(*) >= 2
```

# DB Mobili

---

Articoli (Art\_Cod, Cat\_Cod, Art\_Descrizione, Art\_Prezzo, Art\_IVA, Art\_Spese Trasporto)

FK: Cat\_Cod REFERENCES Categorie

Categorie (Cat\_Cod, Cat\_Descrizione)

Componenti (Com\_Cod, Com\_Descrizione, Com\_Costo, Lab\_Cod)

FK: Lab\_Cod REFERENCES Laboratori

Laboratori (Lab\_Cod, Lab\_Indirizzo, Lab\_Citta, Lab\_Telefono)

Ordini (Ord\_Cod, Neg\_Cod, Ord\_Data)

FK: Neg\_Cod REFERENCES Negozi

Negozi (Neg\_Cod, Neg\_Nome, Neg\_Indirizzo, Neg\_Citta, Neg\_Telefono)

CompArt (Art\_Cod, Com\_Cod, CompArt\_Qta)

FK: Art\_Cod REFERENCES Articoli

FK: Com\_Cod REFERENCES Componenti

OrdArt (Ord\_Cod, Art\_Cod, OrdArt\_Qta)

FK: Ord\_Cod REFERENCES Ordini

FK: Art\_Cod REFERENCES Articoli

# DB Mobili

## Negozi

<u>Neg_Cod</u>
Neg_Nome
Neg_Indirizzo
Neg_Citta
Neg_Telefono

## Categorie

<u>Cat_Cod</u>
Cat_Descrizione

## Laboratori

<u>Lab_Cod</u>
Lab_Indirizzo
Lab_Citta
Lab_Telefono

## Ordini

<u>Ord_Cod</u>
Neg_Cod
Ord_Data

## OrdArt

<u>Ord_Cod</u>
<u>Art_Cod</u>
OrdArt_Qta

## Articoli

<u>Art_Cod</u>
Cat_Cod
Art_Descrizione
Art_Prezzo
Art_IVA
Art_Spese_Trasporto

## CompArt

<u>Art_Cod</u>
<u>Com_Cod</u>
CompArt_Qta

## Componenti

<u>Com_Cod</u>
Com_Descrizione
Com_Costo
Lab_Cod





# DB Mobili – Ordinamento

---

- ▶ Ordinare in base ai nomi i negozi di Roma

```
SELECT *  
FROM Negozi  
WHERE Neg_Citta = 'Roma'  
ORDER BY Neg_Nome
```

# DB Mobili – Ordinamento

---

- ▶ Ordinare i componenti per costi decrescenti e, a parità di costo, per codici crescenti

```
SELECT *
```

```
FROM Componenti
```

```
ORDER BY Com_Costo DESC, Com_Cod
```

# DB Mobili – Aggregazione e raggruppamento

---

- ▶ La minima e la massima quantità di articoli richiesti in un ordine

```
SELECT MIN(OrdArt_Qta) , MAX(OrdArt_Qta)  
FROM OrdArt
```

# DB Mobili – Aggregazione e raggruppamento

---

- ▶ La media dei prezzi dei componenti non prodotti in alcun laboratorio

```
SELECT AVG(Com_Costo)
FROM Componenti
WHERE Lab_Cod IS NULL
```

# DB Mobili – Aggregazione e raggruppamento

---

- ▶ Il prezzo più alto per ciascuna categoria

```
SELECT Cat_Cod, MAX(Art_Prezzo)  
FROM Articoli  
GROUP BY Cat_Cod
```

# DB Mobili – Aggregazione e raggruppamento

---

- ▶ Quanti laboratori sono presenti in ciascuna città, escludendo le città con un solo laboratorio

```
SELECT Lab_Citta, COUNT(Lab_Citta)
FROM Laboratori
GROUP BY Lab_Citta
HAVING COUNT(Lab_Citta) > 1
```

# DB Mobili – Aggregazione e raggruppamento

---

- ▶ Da quanti componenti è composto ciascun articolo, considerando solo gli articoli composti da almeno 10 componenti

```
SELECT Art_Cod, SUM(CompArt_Qta)
FROM CompArt
GROUP BY Art_Cod
HAVING SUM(CompArt_Qta) >= 10
```