

I concetti del Modello Relazionale - Parte 4

Prof. Francesco Gobbi

I.I.S.S. Galileo Galilei, Ostiglia

28 novembre 2024

Relazioni di riferimento

Tabella Dipendenti

ID_Dipendente	Nome	Cognome
1	Marco	Rossi
2	Laura	Bianchi
3	Sara	Verdi
4	Luca	Neri

Tabella Progetti

ID_Progetto	Nome_Progetto	ID_Dipendente
101	ProgettoAlpha	1
102	ProgettoBeta	2
103	ProgettoGamma	3
104	ProgettoDelta	NULL

Esercizio 1: Selezione

Testo: Trova tutti i dipendenti con ID maggiore di 2.

Operazione:

$$\sigma_{ID_Dipendente > 2}(Dipendenti)$$

Risultato:

ID_Dipendente	Nome	Cognome
3	Sara	Verdi
4	Luca	Neri

Esercizio 2: Selezione

Testo: Trova tutti i progetti che non hanno un dipendente assegnato.

Operazione:

$\sigma_{ID_Dipendente \text{ IS NULL}}(Progetti)$

Risultato:

ID_Progetto	Nome_Progetto	ID_Dipendente
104	<i>ProgettoDelta</i>	<i>NULL</i>

Esercizio 3: Selezione

Testo: Trova tutti i progetti con ID maggiore di 102.

Operazione:

$$\sigma_{ID_Progetto > 102}(Progetti)$$

Risultato:

ID_Progetto	Nome_Progetto	ID_Dipendente
103	<i>ProgettoGamma</i>	3
104	<i>ProgettoDelta</i>	NULL

Esercizio 1: Proiezione

Testo: Visualizza solo i nomi e i cognomi dei dipendenti.

Operazione:

$\pi_{\text{Nome, Cognome}}(\text{Dipendenti})$

Risultato:

Nome	Cognome
Marco	Rossi
Laura	Bianchi
Sara	Verdi
Luca	Neri

Esercizio 2: Proiezione

Testo: Visualizza solo i nomi dei progetti.

Operazione:

$\pi_{\text{Nome_Progetto}}(\text{Progetti})$

Risultato:

Nome_Progetto
<i>ProgettoAlpha</i>
<i>ProgettoBeta</i>
<i>ProgettoGamma</i>
<i>ProgettoDelta</i>

Esercizio 3: Proiezione

Testo: Visualizza solo gli ID dei dipendenti e dei relativi cognomi.

Operazione:

$\pi_{ID_Dipendente, Cognome}(Dipendenti)$

Risultato:

ID_Dipendente	Cognome
1	<i>Rossi</i>
2	<i>Bianchi</i>
3	<i>Verdi</i>
4	<i>Neri</i>

Esercizio 1: Equi Join con Proiezione

Testo: Trova il nome del progetto e il nome del dipendente per tutti i progetti con un dipendente assegnato.

Operazione:

$\pi_{\text{Nome_Progetto}, \text{Nome}}$

$\left(\text{Progetti} \bowtie_{\text{Progetti.ID_Dipendente} = \text{Dipendenti.ID_Dipendente}} \text{Dipendenti} \right)$

Risultato:

Nome_Progetto	Nome
<i>ProgettoAlpha</i>	<i>Marco</i>
<i>ProgettoBeta</i>	<i>Laura</i>
<i>ProgettoGamma</i>	<i>Sara</i>

Nota: Gli attributi usati nella condizione (ad esempio, ID_Dipendente) sono mantenuti nella relazione intermedia, ma non compaiono nel risultato dopo la proiezione.

Esercizio 2: Equi Join con Selezione

Testo: Trova i progetti con dipendenti assegnati, ma solo quelli in cui l'ID del dipendente è maggiore di 2.

Operazione:

$$\sigma_{\text{Dipendenti.ID_Dipendente} > 2}$$

$$\left(\text{Progetti} \bowtie_{\text{Progetti.ID_Dipendente} = \text{Dipendenti.ID_Dipendente}} \text{Dipendenti} \right)$$

Risultato:

ID_Progetto	Nome_Progetto	ID_Dipendente	Nome
103	<i>ProgettoGamma</i>	3	<i>Sara</i>

Esercizio 3: Equi Join con Selezione e Proiezione

Testo: Trova i nomi dei progetti e i nomi dei dipendenti assegnati, ma solo per i progetti il cui ID è maggiore di 102.

Operazione:

$$\pi_{\text{Nome_Progetto}, \text{Nome}} \left(\sigma_{\text{Progetti.ID_Progetto} > 102} \right.$$

$$\left. \left(\text{Progetti} \bowtie_{\text{Progetti.ID_Dipendente} = \text{Dipendenti.ID_Dipendente}} \text{Dipendenti} \right) \right)$$

Risultato:

Nome_Progetto	Nome
<i>ProgettoGamma</i>	<i>Sara</i>

Nota: L'Equi Join conserva attributi duplicati nella relazione intermedia, ma la proiezione li elimina nel risultato finale.

Esercizio 1: Join Naturale con Proiezione

Testo: Trova il nome del progetto e il nome del dipendente per tutti i progetti con un dipendente assegnato.

Operazione:

$$\pi_{\text{Nome_Progetto}, \text{Nome}} \left(\text{Progetti} \bowtie \text{Dipendenti} \right)$$

Risultato:

Nome_Progetto	Nome
<i>ProgettoAlpha</i>	<i>Marco</i>
<i>ProgettoBeta</i>	<i>Laura</i>
<i>ProgettoGamma</i>	<i>Sara</i>

Nota: Gli attributi duplicati vengono automaticamente eliminati nel risultato del join naturale.

Esercizio 2: Join Naturale con Selezione

Testo: Trova i progetti con dipendenti assegnati, ma solo quelli in cui l'ID del dipendente è maggiore di 2.

Operazione:

$$\sigma_{ID_Dipendente > 2} \left(Progetti \bowtie Dipendenti \right)$$

Risultato:

Nome_Progetto	ID_Dipendente	Nome
<i>ProgettoGamma</i>	3	<i>Sara</i>

Nota: Anche con il Join Naturale, la selezione può essere applicata ai dati risultanti in modo analogo.

Esercizio 3: Join Naturale con Selezione e Proiezione

Testo: Trova i nomi dei progetti e i nomi dei dipendenti assegnati, ma solo per i progetti il cui ID è maggiore di 102.

Operazione:

$$\pi_{\text{Nome_Progetto}, \text{Nome}} \left(\sigma_{\text{ID_Progetto} > 102} \left(\text{Progetti} \bowtie \text{Dipendenti} \right) \right)$$

Risultato:

Nome_Progetto	Nome
<i>ProgettoGamma</i>	<i>Sara</i>

Nota: Il Join Naturale elimina automaticamente gli attributi comuni duplicati, riducendo il grado della relazione risultante.

Nuove tabelle di riferimento

Tabella Clienti

ID_Cliente	Nome	Città
1	<i>Anna</i>	<i>Roma</i>
2	<i>Luca</i>	<i>Milano</i>
3	<i>Maria</i>	<i>Torino</i>
4	<i>Marco</i>	<i>Napoli</i>

Tabella Ordini

ID_Ordine	Prodotto	ID_Cliente
101	<i>Laptop</i>	1
102	<i>Smartphone</i>	2
103	<i>Tablet</i>	3
104	<i>Monitor</i>	<i>NULL</i>

Esercizio 1: Equi Join con Selezione e Proiezione

Testo: Trova i nomi dei clienti e i prodotti acquistati da chi vive a "Roma".

Operazione:

$$\pi_{\text{Nome, Prodotto}} \left(\sigma_{\text{Città} = \text{'Roma'}} \left(\text{Clienti} \bowtie_{\text{Clienti.ID_Cliente} = \text{Ordini.ID_Cliente}} \text{Ordini} \right) \right)$$

Risultato:

Nome	Prodotto
<i>Anna</i>	<i>Laptop</i>

Nota: L'operazione di selezione filtra i clienti con 'Città = 'Roma'' prima di eseguire il join.

Esercizio 2: Equi Join con Selezione e Proiezione

Testo: Trova i nomi dei clienti e i prodotti acquistati, ma solo per gli ordini con ID maggiore di 102.

Operazione:

$$\pi_{\text{Nome, Prodotto}} \left(\sigma_{\text{Ordini.ID_Ordine} > 102} \left(\text{Clienti} \bowtie_{\text{Clienti.ID_Cliente} = \text{Ordini.ID_Cliente}} \text{Ordini} \right) \right)$$

Risultato:

Nome	Prodotto
Maria	Tablet

Nota: L'operazione di selezione filtra gli ordini con 'ID_Ordine > 102' prima di eseguire il join.

Esercizio 3: Equi Join con Selezione e Proiezione

Testo: Trova i nomi dei clienti e i prodotti acquistati, ma solo per i clienti con ID maggiore di 2.

Operazione:

$$\pi_{\text{Nome, Prodotto}} \left(\sigma_{\text{Clienti.ID_Cliente} > 2} \left(\text{Clienti} \bowtie_{\text{Clienti.ID_Cliente} = \text{Ordini.ID_Cliente}} \text{Ordini} \right) \right)$$

Risultato:

Nome	Prodotto
Maria	Tablet
Marco	Monitor

Nota: La selezione filtra i clienti con 'ID_Cliente > 2' prima di eseguire il join.

Nuove tabelle di riferimento

Tabella Studenti

ID_Studente	Nome	Classe
1	<i>Marco</i>	5A
2	<i>Anna</i>	5B
3	<i>Luca</i>	4A
4	<i>Sara</i>	5A
5	<i>Maria</i>	4B

Tabella Esami

ID_Esame	Materia	ID_Studente
101	<i>Matematica</i>	1
102	<i>Inglese</i>	2
103	<i>Fisica</i>	1
104	<i>Chimica</i>	4
105	<i>Storia</i>	5

Esercizio 1: Equi Join con Selezione e Proiezione

Testo: Trova i nomi degli studenti e le materie d'esame per gli studenti della classe "5A".

Operazione:

$$\pi_{\text{Nome, Materia}} \left(\sigma_{\text{Classe} = '5A'} \right)$$

$$\left(\text{Studenti} \bowtie_{\text{Studenti.ID_Studente} = \text{Esami.ID_Studente}} \text{Esami} \right)$$

Risultato:

Nome	Materia
Marco	Matematica
Marco	Fisica
Sara	Chimica

Nota:

- ▶ La condizione $\text{Classe} = '5A'$ seleziona solo gli studenti della classe 5A.
- ▶ L'Equi Join associa ogni studente con i propri esami utilizzando $\text{ID_Studente} = \text{ID_Studente}$.

Esercizio 2: Equi Join con Selezione complessa e Proiezione

Testo: Trova i nomi degli studenti, le materie d'esame e le classi, ma solo per gli esami il cui ID è maggiore di 102 e appartengono a studenti in classi di quarta.

Operazione:

$$\pi_{\text{Nome, Materia, Classe}} \left(\sigma_{\text{ID_Esame} > 102 \text{ AND Classe LIKE '4\%'}} \left(\text{Studenti} \bowtie_{\text{Studenti.ID_Studente} = \text{Esami.ID_Studente}} \text{Esami} \right) \right)$$

Risultato:

Nome	Materia	Classe
Maria	Storia	4B

Nota:

- ▶ $\text{ID_Esame} > 102$: Filtra solo gli esami con ID maggiore di 102.
- ▶ Classe LIKE '4\%' : Seleziona solo gli studenti di classi di quarta.
- ▶ L'AND va a collegare le due condizioni per la selezione.
- ▶ Il join collega ogni studente con i propri esami tramite ID_Studente.

Esercizio 3: Equi Join con Selezione e Proiezione multipla

Testo: Trova i nomi degli studenti e le materie d'esame, ma solo per studenti che hanno sostenuto esami di "Matematica" o "Fisica".

Operazione:

$$\pi_{\text{Nome, Materia}} \left(\sigma_{\text{Materia} = \text{'Matematica'} \text{ OR } \text{Materia} = \text{'Fisica'}} \left(\text{Studenti} \bowtie_{\text{Studenti.ID_Studente} = \text{Esami.ID_Studente}} \text{Esami} \right) \right)$$

Risultato:

Nome	Materia
Marco	Matematica
Marco	Fisica

Nota:

- ▶ $\text{Materia} = \text{'Matematica'} \text{ OR } \text{Materia} = \text{'Fisica'}$.
- ▶ L'OR va a selezionare le tuple in cui almeno una delle due condizioni, connesse dall'OR appunto, solo vere.
- ▶ Il join associa ogni studente con i propri esami tramite ID_Studente.

Esercizio 4: Equi Join con Selezione complessa e Proiezione

Testo: Trova i nomi degli studenti, le materie d'esame e le classi, ma solo per gli esami di "Matematica" o "Fisica" sostenuti dagli studenti della classe "5A" o "4B", il cui ID esame è pari.

Operazione:

$$\pi_{\text{Nome, Materia, Classe}} \left(\sigma_{(\text{Materia} = \text{'Matematica'} \text{ OR } \text{Materia} = \text{'Fisica'}) \text{ AND } (\text{Classe} = \text{'5A'} \text{ OR } \text{Classe} = \text{'4B'}) \text{ AND } \text{ID_Esame} \% 2 = 0} \left(\text{Studenti} \bowtie_{\text{Studenti.ID_Studente} = \text{Esami.ID_Studente}} \text{Esami} \right) \right)$$

Risultato:

Nome	Materia	Classe
Marco	Fisica	5A

Nota:

- ▶ $\text{Materia} = \text{'Matematica'} \text{ OR } \text{Materia} = \text{'Fisica'}$: Filtra gli esami per le materie specifiche.
- ▶ $\text{Classe} = \text{'5A'} \text{ OR } \text{Classe} = \text{'4B'}$.
- ▶ $\text{ID_Esame} \% 2 = 0$.

Tabelle di riferimento (modello logico)

Tabella Docenti (Docenti)

Docenti(ID_Docente, Nome, Dipartimento)

Tabella Corsi (Corsi)

Corsi(ID_Corso, Nome_Corso, ID_Docente)

Tabella Studenti (Studenti)

Studenti(ID_Studente, Nome, Classe, ID_Corso)

Note:

- ▶ Attributi chiave: Indicano gli identificatori univoci di ogni tabella.
- ▶ **Attributi chiave esterna**: Collegano una tabella con un'altra tramite riferimenti alle chiavi primarie.

Esercizio 1: Join e Proiezione

Testo: Trova i nomi dei docenti e i nomi dei corsi che insegnano.

Operazione:

$$\pi_{\text{Nome, Nome_Corso}} \left(\text{Docenti} \bowtie_{\text{Docenti.ID_Docente} = \text{Corsi.ID_Docente}} \text{Corsi} \right)$$

Risultato:

Nome	Nome_Corso
<i>Prof.Bianchi</i>	<i>Analisi1</i>
<i>Prof.Rossi</i>	<i>Fisica1</i>
<i>Prof.Verdi</i>	<i>Informatica</i>

Nota:

- ▶ L'Equi Join associa i docenti ai corsi che insegnano utilizzando ID_Docente.
- ▶ La proiezione restituisce solo i campi richiesti: 'Nome' (docente) e 'Nome_Corso'.

Esercizio 2: Join, Selezione e Proiezione

Testo: Trova i nomi degli studenti che frequentano corsi insegnati nel dipartimento di "Fisica".

Operazione:

$$\pi_{\text{Studenti.Nome}} \left(\sigma_{\text{Dipartimento} = \text{'Fisica'}} \left(\left(\text{Studenti} \bowtie_{\text{Studenti.ID_Corso} = \text{Corsi.ID_Corso}} \left(\text{Corsi} \bowtie_{\text{Corsi.ID_Docente} = \text{Docenti.ID_Docente}} \text{Docenti} \right) \right) \right) \right)$$

Risultato:

Nome
Giulia
Luca

Nota:

- ▶ La selezione $\text{Dipartimento} = \text{'Fisica'}$ filtra i docenti del dipartimento di Fisica.
- ▶ **Due join concatenati collegano 'Studenti', 'Corsi' e 'Docenti' tramite le chiavi esterne.**

Esercizio 3: Join complesso, Selezione e Proiezione

Testo: Trova i nomi degli studenti e i nomi dei corsi frequentati, ma solo per studenti in classi di quinta che frequentano corsi insegnati dal dipartimento di "Informatica".

Operazione:

$\pi_{\text{Studenti.Nome, Corsi.Nome_Corso}} \left(\sigma_{\text{Classe LIKE '5\%' AND Dipartimento = 'Informatica'}}$

$\left(\text{Studenti} \bowtie_{\text{Studenti.ID_Corso = Corsi.ID_Corso}} \right.$

$\left. \left(\text{Corsi} \bowtie_{\text{Corsi.ID_Docente = Docenti.ID_Docente}} \text{Docenti} \right) \right)$

Risultato:

Nome	Nome_Corso
Elisa	Programmazione
Marco	BasidiDati

Esercizio 3: Join complesso, Selezione e Proiezione

Nota:

- ▶ Classe LIKE '5%': Filtra solo gli studenti delle classi di quinta.
- ▶ Dipartimento = 'Informatica': Seleziona i corsi insegnati dal dipartimento di Informatica.
- ▶ I join concatenano 'Studenti', 'Corsi' e 'Docenti' tramite le chiavi esterne.
- ▶ La proiezione restituisce solo i campi richiesti: 'Nome' dello studente e 'Nome_Corso'.