

ORACLE

# Corso APEX – beanTech

---

**Ing. Roberto Capancioni**

Lezione 4 - 25 Settembre 2023



# Chi Sono

Ing. Roberto Capancioni  
—Sviluppo Oracle APEX

**capancioni.com**

 **Oracle ACE  
Associate**



Email: [sviluppo@capancioni.com](mailto:sviluppo@capancioni.com)  
Linkedin: <https://www.linkedin.com/in/robertocapancioni>  
Web: <https://capancioni.com>

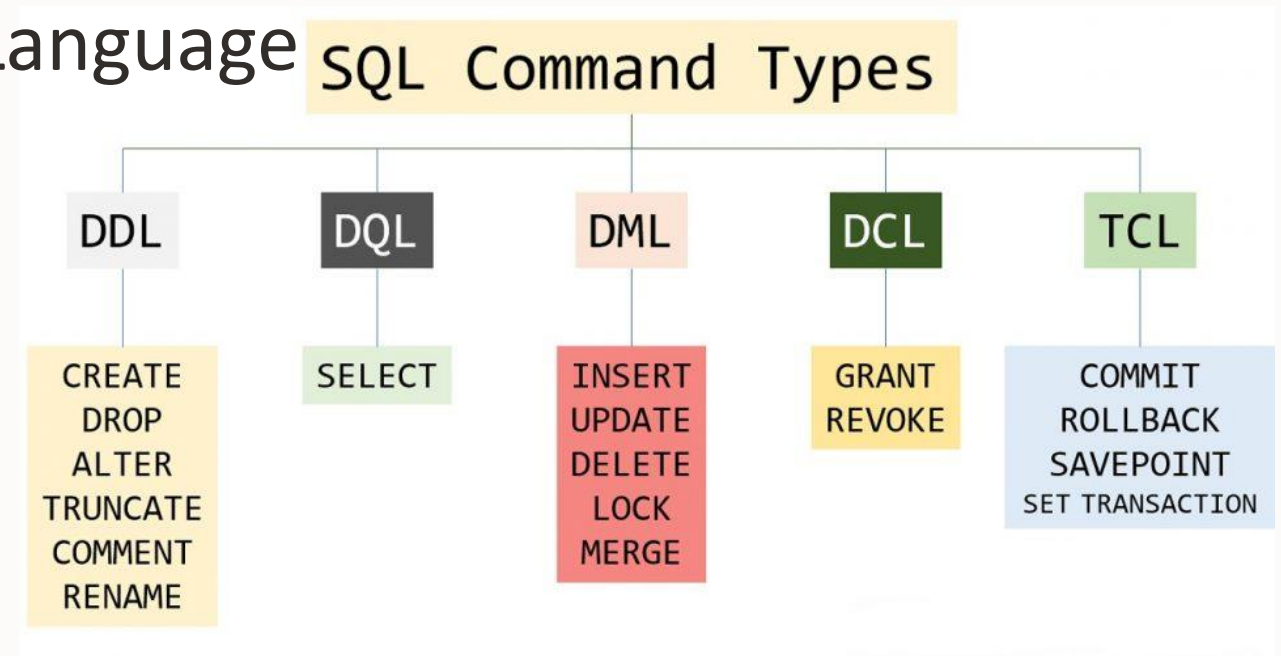


# Comandi SQL

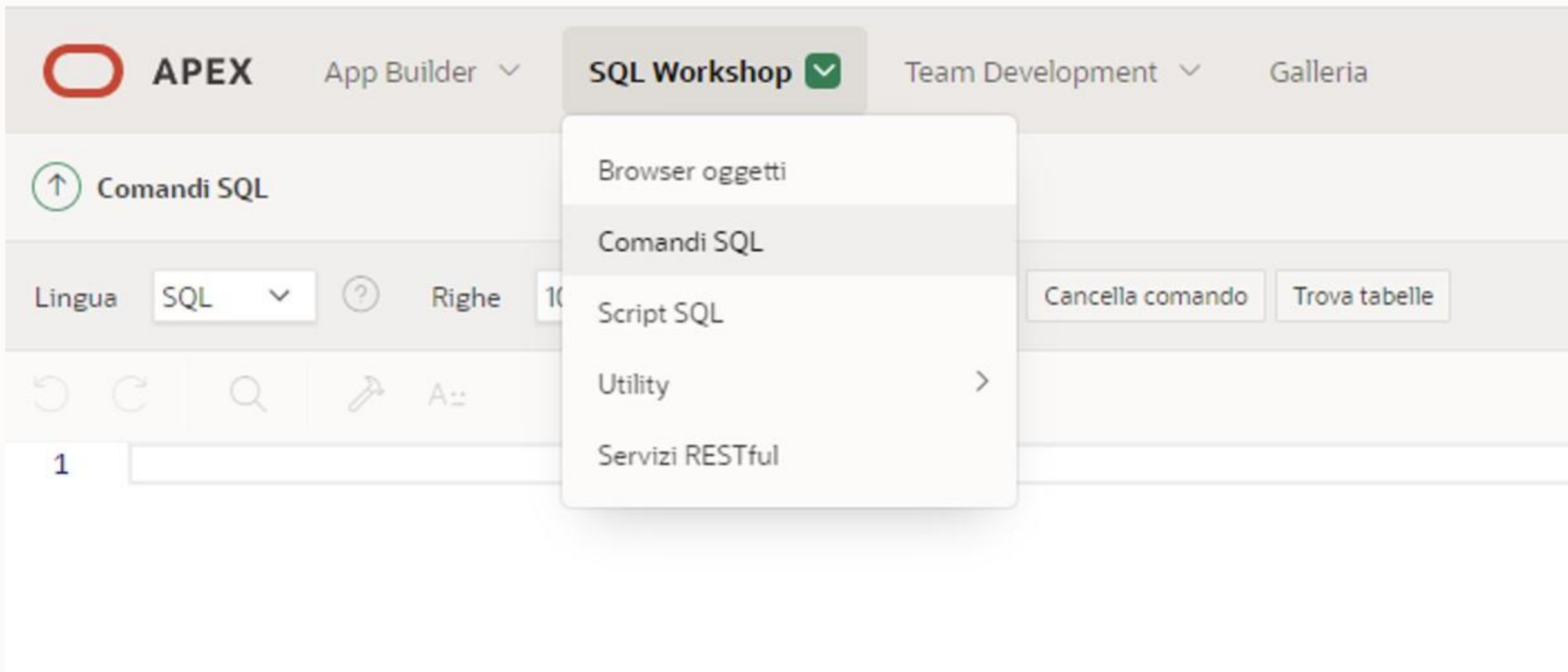
---

## Modello Dati Relazionale – Come li gestisco?

- **SQL:** Structured Query Language
- **DDL:** Data Definition Language
- **DML:** Data Manipulation Language
- **DQL:** Data Query Language
- **DCL:** Data Control Language
- **TCL:** Transaction Control Language



## SQL Workshop – Comandi SQL



The screenshot displays the APEX SQL Workshop interface. At the top, the navigation bar includes the APEX logo, 'App Builder', 'SQL Workshop' (selected with a green checkmark), 'Team Development', and 'Galleria'. Below this, the 'Comandi SQL' section is visible, featuring a dropdown menu for 'Lingua' set to 'SQL', a 'Righe' counter showing '10', and buttons for 'Cancella comando' and 'Trova tabelle'. A context menu is open over the 'Comandi SQL' section, listing the following options: 'Browser oggetti', 'Comandi SQL' (highlighted), 'Script SQL', 'Utility' (with a right arrow), and 'Servizi RESTful'. The main workspace area shows a line number '1' and an empty text input field.



# SQL Workshop – Comandi SQL

```
select *  
from D04_ATTIVITA
```

APEX

App Builder

SQL Workshop

Team Development

Galleria

Cerca

X2x250

FOEX

Comandi SQL

SchemaX250

LinguaSQL

Righe10

Cancella comando

Trova tabelle

Salva

Esegui

↶↷🔍🔗A

1  
2  
3  
4

```
select *  
from d02_attivita
```

Risultati

Spiega

Descrivi

SQL salvato

Cronologia

ID	PROGETTO	ATTIVITA	DATA_INIZIO	DATA_FINE	STATO	ASSEGNATO_A	COSTO	BUDGET
59	Manutenzione dei sistemi di supporto	Organizzare la copertura delle vacanze	09/09/2022	17/10/2022	Aperto	Simone Blocchi	300	500
60	Manutenzione dei sistemi di supporto	Aggiornamenti del software HR	23/06/2022	03/07/2022	In Attesa	Luigi Bianchi	8000	7000
61	Manutenzione dei sistemi di supporto	Applicare gli aggiornamenti del sistema di fatturazione	16/07/2022	28/07/2022	In Attesa	Andrea Verdi	9500	7000
62	Monitoraggio dei progetti software	Caricare le attività e i miglioramenti correnti	02/09/2022	12/09/2022	Chiuso	Sandra Legni	400	500
63	Monitoraggio dei progetti software	Condurre la riunione di avvio del progetto	15/08/2022	30/10/2022	Chiuso	Luigi Bianchi	100	100
64	Monitoraggio dei progetti software	Personalizza il software dei progetti software	12/07/2022	22/08/2022	Aperto	Sandra Legni	600	1000
65	Monitoraggio dei progetti software	Inserisci i dati di base (progetti, milestone, ecc.)	28/06/2022	24/09/2022	Chiuso	Sandra Legni	200	200
66	Sito Web pubblico	Sviluppare pagine web	25/08/2022	04/09/2022	In Attesa	Rina Pietra	800	2000
67	Sito Web pubblico	Acquistare licenze software aggiuntive, se necessario	18/10/2022	10/11/2022	In Attesa	Simone Blocchi	300	1000
68	Sito Web pubblico	Determinare il server host	05/08/2022	14/09/2022	Chiuso	Rina Pietra	200	200



# SQL Workshop – Comandi SQL

```
select ID,  
       PROGETTO,  
       ATTIVITA,  
       DATA_INIZIO,  
       DATA_FINE,  
       STATO,  
       ASSEGNATO_A,  
       COSTO,  
       BUDGET  
from D04_ATTIVITA
```

APEX

App Builder

SQL Workshop

Team Development

Galleria

Cerca

X2

x250

FO

EX

Comandi SQL

Schema: X250

Lingua: SQL

Righe: 10

Cancello comando

Trova tabelle

Salva

Esegui

↶

↷

🔍

🔗

A:z

4

5

6

7

8

9

10

11

12

13

14

15

select ID,

PROGETTO,

ATTIVITA,

DATA\_INIZIO,

DATA\_FINE,

STATO,

ASSEGNATO\_A,

COSTO,

BUDGET

from D04\_ATTIVITA

Resultati

Spiega

Descrivi

SQL salvato

Cronologia

ID	PROGETTO	ATTIVITA	DATA_INIZIO	DATA_FINE	STATO	ASSEGNATO_A	COSTO	BUDGET
59	Manutenzione dei sistemi di supporto	Organizzare la copertura delle vacanze	09/09/2022	17/10/2022	Aperto	Simone Blocchi	300	500
60	Manutenzione dei sistemi di supporto	Aggiornamenti del software HR	23/06/2022	03/07/2022	In Attesa	Luigi Bianchi	8000	7000
61	Manutenzione dei sistemi di supporto	Applicare gli aggiornamenti del sistema di fatturazione	16/07/2022	28/07/2022	In Attesa	Andrea Verdi	9500	7000
62	Monitoraggio dei progetti software	Caricare le attività e i miglioramenti correnti	02/09/2022	12/09/2022	Chiuso	Sandra Legni	400	500
63	Monitoraggio dei progetti software	Condurre la riunione di avvio del progetto	15/08/2022	30/10/2022	Chiuso	Luigi Bianchi	100	100
64	Monitoraggio dei progetti software	Personalizza il software dei progetti software	12/07/2022	22/08/2022	Aperto	Sandra Legni	600	1000
65	Monitoraggio dei progetti software	Inserisci i dati di base (progetti, milestone, ecc.)	28/06/2022	24/09/2022	Chiuso	Sandra Legni	200	200
66	Sito Web pubblico	Sviluppare pagine web	25/08/2022	04/09/2022	In Attesa	Rina Pietra	800	2000
67	Sito Web pubblico	Acquistare licenze software aggiuntive, se necessario	18/10/2022	10/11/2022	In Attesa	Simone Blocchi	300	1000
68	Sito Web pubblico	Determinare il server host	05/08/2022	14/09/2022	Chiuso	Rina Pietra	200	200







# SQL Workshop – Comandi SQL

```
select A.PROGETTO,  
       A.ATTIVITA,  
       A.DATA_INIZIO,  
       A.DATA_FINE,  
       A.DATA_FINE - A.DATA_INIZIO as GIORNI  
from D04_ATTIVITA A
```

ALIAS DI COLONNA  
(la parola «AS» è opzionale)

ALIAS DI TABELLA  
(la parola «AS» non va messa)

The screenshot shows the APEX SQL Workshop interface. The top navigation bar includes 'APEX', 'App Builder', 'SQL Workshop', 'Team Development', and 'Galleria'. A search bar is on the right. Below the navigation bar, the 'Comandi SQL' section is active, showing the SQL command entered in the editor. The command is: `select PROGETTO, ATTIVITA, DATA_INIZIO, DATA_FINE, DATA_FINE - DATA_INIZIO as GIORNI from D02_ATTIVITA A`. The 'Risultati' tab is selected, displaying a table with 5 columns: PROGETTO, ATTIVITA, DATA\_INIZIO, DATA\_FINE, and GIORNI. The table contains 7 rows of data.

PROGETTO	ATTIVITA	DATA_INIZIO	DATA_FINE	GIORNI
Manutenzione dei sistemi di supporto	Organizzare la copertura delle vacanze	09/09/2022	17/10/2022	38
Manutenzione dei sistemi di supporto	Aggiornamenti del software HR	23/06/2022	03/07/2022	10
Manutenzione dei sistemi di supporto	Applicare gli aggiornamenti del sistema di fatturazione	16/07/2022	28/07/2022	12
Monitoraggio dei progetti software	Caricare le attività e i miglioramenti correnti	02/09/2022	12/09/2022	10
Monitoraggio dei progetti software	Condurre la riunione di avvio del progetto	15/08/2022	30/10/2022	76
Monitoraggio dei progetti software	Personalizza il software dei progetti software	12/07/2022	22/08/2022	41
Monitoraggio dei progetti software	Inserisci i dati di base (progetti, milestone, ecc.)	28/06/2022	24/09/2022	88



# SQL Workshop – Comandi SQL

```
select A.*
from D04_ATTIVITA A
where PROGETTO = 'Sito Web pubblico'
```

Distingue Minuscole da  
Maiuscole

Apice Singolo non Doppio Apice

APEX

App Builder

SQL Workshop

Team Development

Galleria

Cerca

X2

x250

x250

FO

EX

Comandi SQL

Schema

X250

Lingua

SQL

Righe

10

Cancella comando

Trova tabelle

Salva

Esegui

1

2

3

4

select A.\*

from D02\_ATTIVITA A

where PROGETTO='Sito Web pubblico'

Risultati

Spiega

Descrivi

SQL salvato

Cronologia

ID	PROGETTO	ATTIVITA	DATA_INIZIO	DATA_FINE	STATO	ASSEGNATO_A	COSTO	BUDGET
66	Sito Web pubblico	Sviluppare pagine web	25/08/2022	04/09/2022	In Attesa	Rina Pietra	800	2000
67	Sito Web pubblico	Acquistare licenze software aggiuntive, se necessario	18/10/2022	10/11/2022	In Attesa	Simone Blocchi	300	1000
68	Sito Web pubblico	Determinare il server host	05/08/2022	14/09/2022	Chiuso	Rina Pietra	200	200
69	Sito Web pubblico	Pianificare la pianificazione dell'implementazione	16/10/2022	26/10/2022	In Attesa	Sandra Legni	0	100
70	Sito Web pubblico	Controllare le licenze software	20/11/2022	07/12/2022	Chiuso	Sandra Legni	100	100

5 righe restituite in 0.00 secondi

Gratuito



# SQL Workshop – Comandi SQL

```
select A.*  
  from D04_ATTIVITA A  
 where ATTIVITA like 'Co%'
```



INIZIA PER Co (si usa % non \*)

```
select A.*  
  from D04_ATTIVITA A  
 where ATTIVITA like '%to'
```



TERMINA PER to (si usa % non \*)

```
select A.*  
  from D04_ATTIVITA A  
 where ATTIVITA like '%comp%'
```



CONTIENE comp (si usa % non \*)

# SQL Workshop – Comandi SQL

```
select A.*  
  from D04_ATTIVITA A  
 where COSTO > BUDGET
```

```
select A.*  
  from D04_ATTIVITA A  
 where COSTO <> BUDGET
```



<> o != SIGNIFICA DIVERSO DA

```
select A.*  
  from D04_ATTIVITA A  
 where COSTO between 2000 and 3500
```



GLI ESTREMI SONO COMPRESI

## SQL Workshop – Comandi SQL

```
select A.*  
  from D04_ATTIVITA A  
 where BUDGET is null
```



```
select A.*  
  from D04_ATTIVITA A  
 where BUDGET is not null
```



Con null si usa is o is not

# SQL Workshop – Comandi SQL

```
select A.*  
  from D04_ATTIVITA A  
 where BUDGET = 500
```

```
select A.*  
  from D04_ATTIVITA A  
 where BUDGET <> 500
```



ATTENZIONE!!  
Diverso (<> o !=) NON COMPRENDE il nullo

```
select A.*  
  from D04_ATTIVITA A  
 where BUDGET <> 500  
       and BUDGET is null
```



NESSUNA RIGA!!

# SQL Workshop – Comandi SQL

```
select A.*  
  from D04_ATTIVITA A  
 where BUDGET = 500
```

```
select A.*  
  from D04_ATTIVITA A  
 where NVL(BUDGET,0) <> 500
```

```
select A.*  
  from D04_ATTIVITA A  
 where NVL(BUDGET,0) <> 500  
       and BUDGET is null
```



Soluzione: usare NVL o COALESCE



NESSUNA RIGA!!

# SQL Workshop – Comandi SQL

```
SELECT A.*  
FROM D04_ATTIVITA A  
WHERE DATA_INIZIO > DATE'2023-01-03';
```

```
SELECT A.*  
FROM D04_ATTIVITA A  
WHERE DATA_INIZIO > TO_DATE('01/03/2023', 'DD/MM/YYYY')
```

```
SELECT A.*  
FROM D04_ATTIVITA A  
WHERE DATA_INIZIO > TO_DATE('01/03/2023 13:00:00', 'DD/MM/YYYY hh24:MI:SS');
```

```
SELECT A.DATA_INIZIO,  
       A.DATA_INIZIO + 1 GIORNO_SUCCESIVO,  
       A.DATA_FINE - A.DATA_INIZIO GIORNI_ATTIVITA  
FROM D04_ATTIVITA A
```



# SQL Workshop – Comandi SQL

```
select A.*  
  from D04_ATTIVITA A  
order by PROGETTO
```



CRESCENTE

```
select A.*  
  from D04_ATTIVITA A  
order by PROGETTO ASC
```



CRESCENTE

```
select A.*  
  from D04_ATTIVITA A  
order by PROGETTO DESC
```



DESCRESCENTE

# SQL Workshop – Comandi SQL

```
select A.*  
  from D04_ATTIVITA A  
order by BUDGET DESC NULLS LAST
```

```
select A.*  
  from D04_ATTIVITA A  
order by BUDGET ASC NULLS FIRST
```



Ordinamento di nulli

```
select A.*  
  from D04_ATTIVITA A  
order by PROGETTO, ATTIVITA DESC
```



Ordinamenti Multipli

# SQL Workshop – Comandi SQL

```
select A.*  
  from D04_ATTIVITA A  
order by BUDGET  
  fetch first 5 rows only
```

```
select A.*  
  from D04_ATTIVITA A  
order by BUDGET  
  fetch next 5 rows only
```

```
select A.*  
  from D04_ATTIVITA A  
order by BUDGET ASC NULLS FIRST  
  fetch first row only
```

Solo le prime (o prossime) 5 righe

- First e next sono sinonimi
- Row e rows sono sinonimi
- Se il numero è omissso è = 1

## SQL Workshop – Comandi SQL

```
select A.*  
  from D04_ATTIVITA A  
order by BUDGET  
offset 2 rows  
fetch next 5 rows only
```

```
select A.*  
  from D04_ATTIVITA A  
order by BUDGET  
offset 1 row  
fetch next row only
```

OFFSET salta le prime n righe

# SQL Workshop – Comandi SQL

## select

```
    a.id as attivita_id,  
    a.progetto,  
    a.attivita,  
    a.data_inizio,  
    a.data_fine,  
    /*  
        altri dati  
        da mostrare  
    */  
    a.stato,  
    a.assegnato_a,  
    a.costo,  
    a.budget  
from D04_attivita a  
where progetto = 'Bug Tracker'  
order by BUDGET DESC NULLS LAST  
offset 2 rows --salta 2 righe  
fetch next 5 rows only
```

- Select
- From
- Where
- Order by
- Offset
- fetch

# SQL Workshop – Comandi SQL

```
select A.*  
  from D04_ATTIVITA A  
      -- io sono un commento a riga singola  
order by PROGETTO
```

```
select A.*  
  from D04_ATTIVITA A -- io sono un commento a riga singola  
order by PROGETTO ASC
```

```
select A.*  
      /*  
        Io sono un commnento  
        Multiriga  
      */  
  from D04_ATTIVITA A  
order by PROGETTO DESC
```

# Comuni e Province

---

# Comuni e Province

Carica dati

D12\_COMUNI.xlsx

Specificare la posizione in cui si desidera caricare i dati.

Carica in

Nuova tabella

Tabella esistente

Proprietario tabella

ATO

Nome tabella

D12\_COMUNI

Chiavi primarie

SYS\_GUID

Colonna identità

Usa i tipi di dati della colonna

Impostazioni

Intestazioni colonna

Seleziona foglio

Codifica file

Prima riga con intestazioni

COMUNI

Unicode UTF-8

Anteprima

Analizzate le prime 14 righe per effettuare il campionamento dei tipi di colonna. Nell'anteprima riportata di seguito vengono visualizzate solo le prime 2 colonne e 5 righe. Per visualizzare l'intera anteprima e configurare le impostazioni di caricamento dei dati, fare clic sul pulsante **Anteprima**.

2	COMUNE
3	Abbadia Cerreto
4	Abbadia Lariana
5	Abbategrasso

Carica dati

D12\_COMUNI.xlsx

Specificare la posizione in cui si desidera caricare i dati.

Carica in

Nuova tabella

Tabella esistente

Proprietario tabella

ATO

Nome tabella

D12\_PROVINCE

Selezionare le colonne da caricare.

Configura

Chiavi primarie

SYS\_GUID

Colonna identità

Usa i tipi di dati della colonna

Impostazioni

Intestazioni colonna

Seleziona foglio

Codifica file

Prima riga con intestazioni

PROVINCE

Unicode UTF-8

Anteprima

Analizzate le prime 13 righe per effettuare il campionamento dei tipi di colonna. Nell'anteprima riportata di seguito vengono visualizzate solo le prime 2 colonne e 5 righe. Per visualizzare l'intera anteprima e configurare le impostazioni di caricamento dei dati, fare clic sul pulsante **Anteprima**.

1	PROVINCIA_COD	PROVINCIA
2	VA	Varese
3	CO	Como
4	BS	Brescia
5	PV	Pavia

<

Annulla

Carica dati





# Comuni e Province

RELAZIONE 1 : 1

COMUNE	PROVINCIA_COD
Albano Sant'Alessandro	BG
Acquafredda	BS
Albavilla	CO
Acquanegra Cremonese	CR
Abbadia Lariana	LC
Abbadia Cerreto	LO
Agrate Brianza	MB
Abbiategrosso	MI
Acquanegra sul Chiese	MN
Alagna	PV
Albaredo per San Marco	SO
Agra	VA

1

1



PROVINCIA_COD	PROVINCIA
BG	Bergamo
BS	Brescia
CO	Como
CR	Cremona
LC	Lecco
LO	Lodi
MB	Monza e della Brianza
MI	Milano
MN	Mantova
PV	Pavia
SO	Sondrio
VA	Varese

# Comuni e Province

RELAZIONE 1 : 1

```
select c.comune,  
       c.provincia_cod,  
       p.provincia_cod,  
       p.provincia  
from d04_comune c  
join d04_provincia p  
on c.provincia_cod = p.provincia_cod
```

COMUNE	PROVINCIA_COD	PROVINCIA_COD	PROVINCIA
Agra	VA	VA	Varese
Albavilla	CO	CO	Como
Acquafredda	BS	BS	Brescia
Alagna	PV	PV	Pavia
Albaredo per San Marco	SO	SO	Sondrio
Abbiategrosso	MI	MI	Milano
Albano Sant'Alessandro	BG	BG	Bergamo
Acquanegra Cremonese	CR	CR	Cremona

# Comuni e Province

RELAZIONE 1 : 1

```
delete
  from d04_comune
 where provincia_cod in ('BG', 'BS', 'CO')
```

```
delete
  from d04_provincia
 where provincia_cod in ('MN', 'PV', 'SO', 'VA')
```

# Comuni e Province

RELAZIONE 1 : 1

COMUNE	PROVINCIA_COD
Acquanegra Cremonese	CR
Abbadia Lariana	LC
Abbadia Cerreto	LO
Agrate Brianza	MB
Abbiategrosso	MI
Acquanegra sul Chiese	MN
Alagna	PV
Albaredo per San Marco	SO
Agra	VA

No corrispondenza



1

1

No corrispondenza



PROVINCIA_COD	PROVINCIA
BG	Bergamo
BS	Brescia
CO	Como
CR	Cremona
LC	Lecco
LO	Lodi
MB	Monza e della Brianza
MI	Milano

# Comuni e Province

RELAZIONE 1 : 1

```
select c.comune,  
       c.provincia_cod,  
       p.provincia_cod,  
       p.provincia  
from d04_comune c  
join d04_provincia p  
on c.provincia_cod = p.provincia_cod
```

COMUNE	PROVINCIA_COD	PROVINCIA_COD	PROVINCIA
Abbiategrosso	MI	MI	Milano
Acquanegra Cremonese	CR	CR	Cremona
Abbadia Lariana	LC	LC	Lecco
Abbadia Cerreto	LO	LO	Lodi
Agrate Brianza	MB	MB	Monza e della Brianza

5 righe restituite in 0,00 secondi

[Scarica](#)

# Comuni e Province

JOIN o INNER JOIN

Restituisce solo righe dove i valori di PROVINCIA\_COD sono presenti in **TUTTE E 2** le tabelle

COMUNE	PROVINCIA_COD	PROVINCIA_COD	PROVINCIA
Abbiategrosso	MI	MI	Milano
Acquanegra Cremonese	CR	CR	Cremona
Abbadia Lariana	LC	LC	Lecco
Abbadia Cerreto	LO	LO	Lodi
Agrate Brianza	MB	MB	Monza e della Brianza

5 righe restituite in 0,00 secondi

[Scarica](#)

# Comuni e Province

```
select c.comune,  
       c.provincia_cod,  
       p.provincia_cod,  
       p.provincia  
from d04_comune c  
left  
join d04_provincia p  
on c.provincia_cod = p.provincia_cod
```

COMUNE	PROVINCIA_COD	PROVINCIA_COD	PROVINCIA
Abbiategrosso	MI	MI	Milano
Acquanegra Cremonese	CR	CR	Cremona
Abbadia Lariana	LC	LC	Lecco
Abbadia Cerreto	LO	LO	Lodi
Agrate Brianza	MB	MB	Monza e della Brianza
Acquanegra sul Chiese	MN	-	-
Agra	VA	-	-
Alagna	PV	-	-
Albaredo per San Marco	SO	-	-



Valori Nulli

9 righe restituite in 0,01 secondi   Scarica



# Comuni e Province

## LEFT JOIN o LEFT OUTER JOIN

Restituisce TUTTE le righe della tabella di sinistra (la prima)  
e SOLO le righe della tabella di destra (la seconda) dove i valori sono presenti

COMUNE	PROVINCIA_COD	PROVINCIA_COD	PROVINCIA
Abbiategrosso	MI	MI	Milano
Acquanegra Cremonese	CR	CR	Cremona
Abbadia Lariana	LC	LC	Lecco
Abbadia Cerreto	LO	LO	Lodi
Agrate Brianza	MB	MB	Monza e della Brianza
Acquanegra sul Chiese	MN	-	-
Agra	VA	-	-
Alagna	PV	-	-
Albaredo per San Marco	SO	-	-



Valori Nulli

9 righe restituite in 0,01 secondi

[Scarica](#)



# Comuni e Province

```
select c.comune,  
       c.provincia_cod,  
       p.provincia_cod,  
       p.provincia  
from d04_comune c  
right  
join d04_provincia p  
on c.provincia_cod = p.provincia_cod
```

COMUNE	PROVINCIA_COD	PROVINCIA_COD	PROVINCIA
Abbadia Cerreto	LO	LO	Lodi
Abbadia Lariana	LC	LC	Lecco
Abbiategrosso	MI	MI	Milano
Acquanegra Cremonese	CR	CR	Cremona
Agrate Brianza	MB	MB	Monza e della Brianza
-	-	BS	Brescia
-	-	BG	Bergamo
-	-	CO	Como



Valori Nulli



8 righe restituite in 0,01 secondi Scarica



# Comuni e Province

## RIGHT JOIN o RIGHT OUTER JOIN

Restituisce TUTTE le righe della tabella di destra (la seconda)  
e SOLO le righe delle tabella di sinistra (la prima) dove i valori sono presenti

COMUNE	PROVINCIA_COD	PROVINCIA_COD	PROVINCIA
Abbadia Cerreto	LO	LO	Lodi
Abbadia Lariana	LC	LC	Lecco
Abbiategrosso	MI	MI	Milano
Acquanegra Cremonese	CR	CR	Cremona
Agrate Brianza	MB	MB	Monza e della Brianza
-	-	BS	Brescia
-	-	BG	Bergamo
-	-	CO	Como

Valori Nulli

8 righe restituite in 0,01 secondi

[Scarica](#)

# Comuni e Province

```
select c.comune,  
       c.provincia_cod,  
       p.provincia_cod,  
       p.provincia  
from d04_comune c  
full  
join d04_provincia p  
on c.provincia_cod = p.provincia_cod
```

COMUNE	PROVINCIA_COD	PROVINCIA_COD	PROVINCIA
-	-	CO	Como
-	-	BS	Brescia
Abbiategrosso	MI	MI	Milano
-	-	BG	Bergamo
Acquanegra Cremonese	CR	CR	Cremona
Abbadia Lariana	LC	LC	Lecco
Abbadia Cerreto	LO	LO	Lodi
Agrate Brianza	MB	MB	Monza e della Brianza
Acquanegra sul Chiese	MN	-	-
Agra	VA	-	-
Alagna	PV	-	-
Albaredo per San Marco	SO	-	-

12 righe restituite in 0,01 secondi

[Scarica](#)

# Comuni e Province

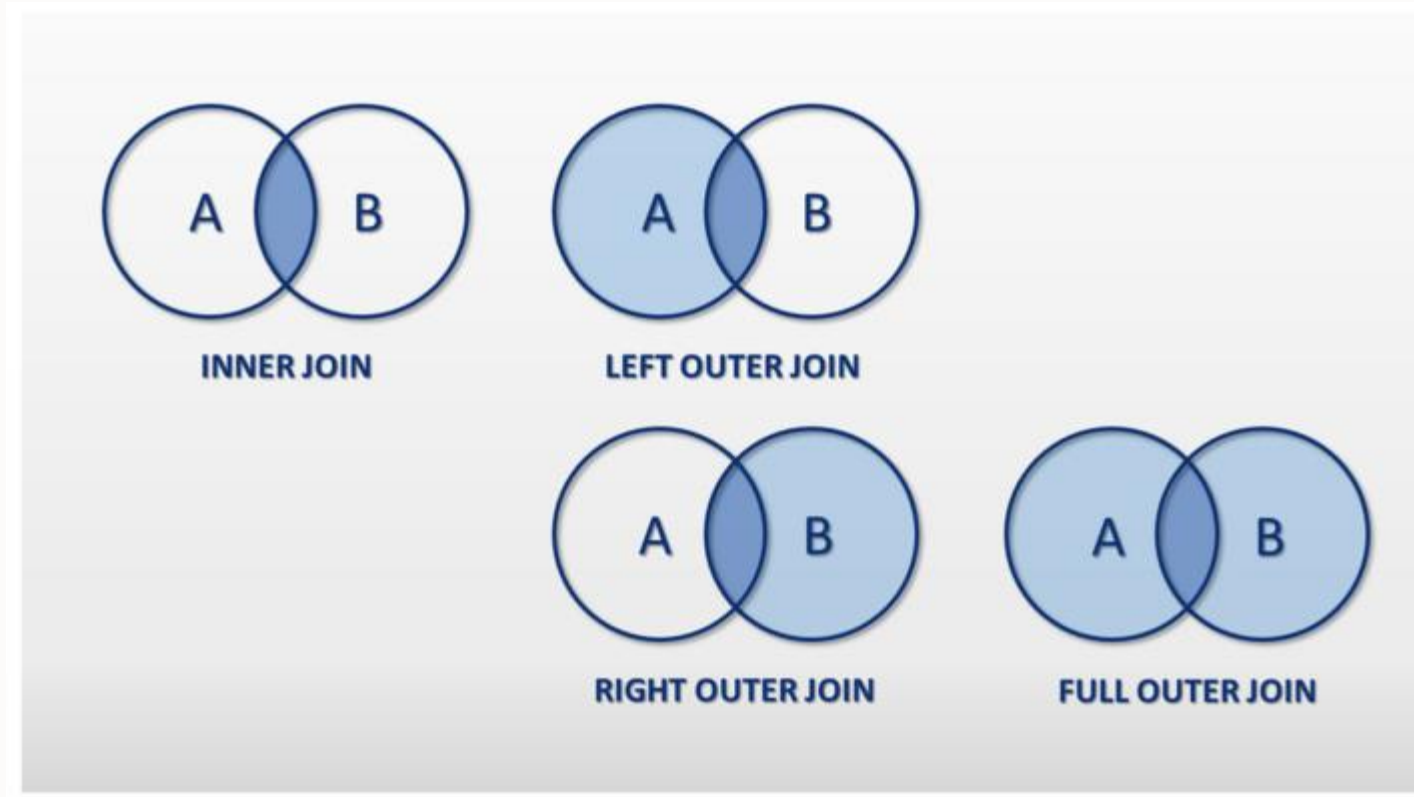
FULL JOIN o FULL OUTER JOIN  
Restituisce TUTTE le righe di entrambe le tabelle

COMUNE	PROVINCIA_COD	PROVINCIA_COD	PROVINCIA
-	-	CO	Como
-	-	BS	Brescia
Abbiategrosso	MI	MI	Milano
-	-	BG	Bergamo
Acquanegra Cremonese	CR	CR	Cremona
Abbadia Lariana	LC	LC	Lecco
Abbadia Cerreto	LO	LO	Lodi
Agrate Brianza	MB	MB	Monza e della Brianza
Acquanegra sul Chiese	MN	-	-
Agra	VA	-	-
Alagna	PV	-	-
Albaredo per San Marco	SO	-	-

12 righe restituite in 0,01 secondi

[Scarica](#)

# Join e Insiemi



# Comuni e Province

RELAZIONE M : 1

```
insert
  into d04_comune
    (comune,provincia_cod)
values ('Arese', 'MI')
```

# Comuni e Province

## RELAZIONE M : 1

COMUNE	PROVINCIA_COD
Acquanegra Cremonese	CR
Abbadia Lariana	LC
Abbadia Cerreto	LO
Agrate Brianza	MB
<b>Abbiategrasso</b>	<b>MI</b>
<b>Arese</b>	<b>MI</b>
Acquanegra sul Chiese	MN
Alagna	PV
Albaredo per San Marco	SO
Agra	VA

No corrispondenza



M

1

No corrispondenza



PROVINCIA_COD	PROVINCIA
BG	Bergamo
BS	Brescia
CO	Como
CR	Cremona
LC	Lecco
LO	Lodi
MB	Monza e della Brianza
<b>MI</b>	<b>Milano</b>

# Comuni e Province

RELAZIONE M : 1

```
select c.comune,  
       c.provincia_cod,  
       p.provincia_cod,  
       p.provincia  
from d04_comune c  
join d04_provincia p  
on c.provincia_cod = p.provincia_cod
```

COMUNE	PROVINCIA_COD	PROVINCIA_COD	PROVINCIA
Abbiategrosso	MI	MI	Milano
Arese	MI	MI	Milano
Acquanegra Cremonese	CR	CR	Cremona
Abbadia Lariana	LC	LC	Lecco
Abbadia Cerreto	LO	LO	Lodi
Agrate Brianza	MB	MB	Monza e della Brianza

6 righe restituite in 0,00 secondi

[Scarica](#)



# Comuni e Province

RELAZIONE M : N

```
insert
  into d04_provincia
    (provincia_cod,provincia)
values ('MI','Milano (doppio)')
```

# Comuni e Province

RELAZIONE M : N

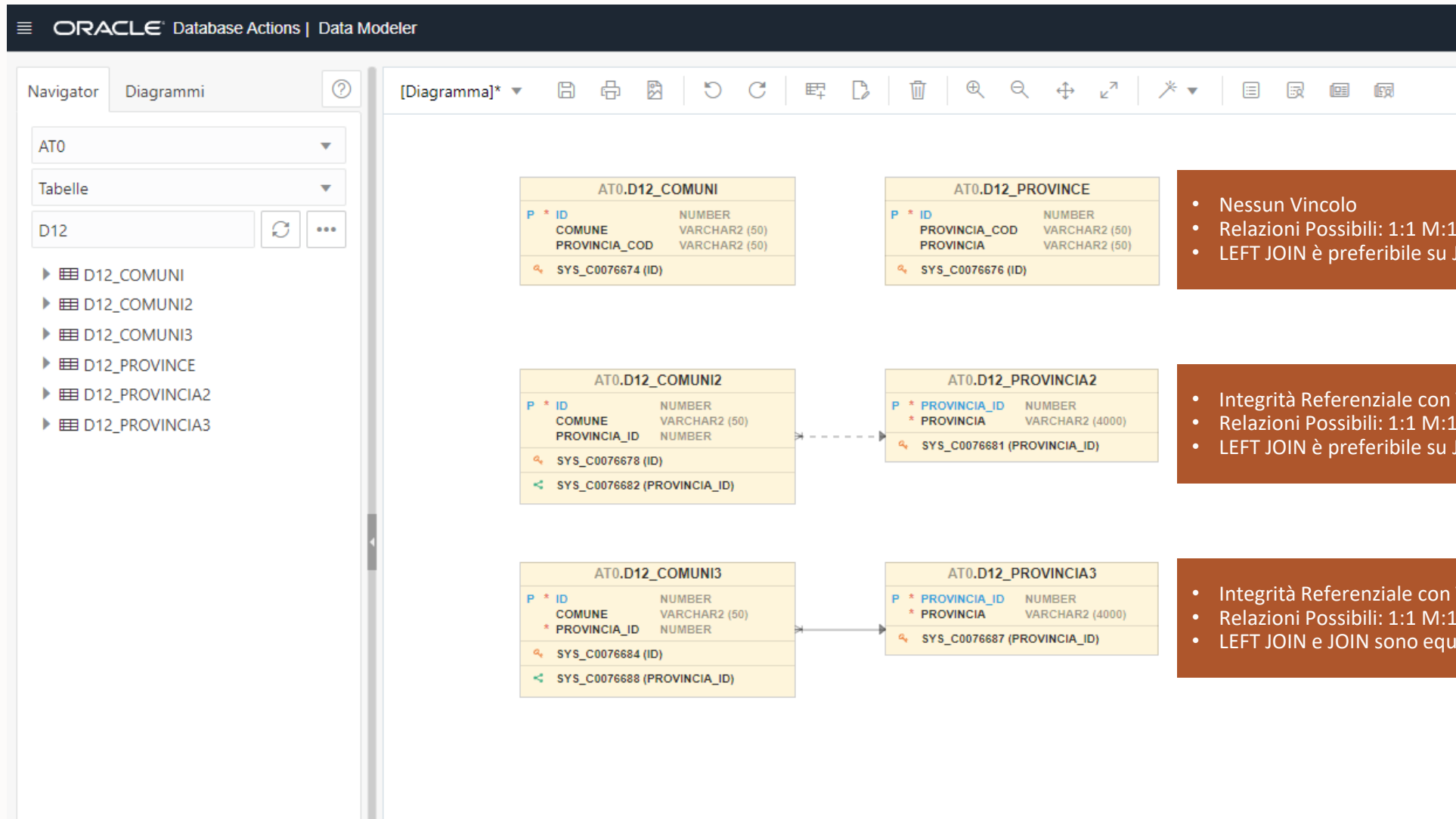
```
select c.comune,  
       c.provincia_cod,  
       p.provincia_cod,  
       p.provincia  
from d04_comune c  
join d04_provincia p  
on c.provincia_cod = p.provincia_cod
```

COMUNE	PROVINCIA_COD	PROVINCIA_COD	PROVINCIA
Abbiategrosso	MI	MI	Milano (doppio)
Arese	MI	MI	Milano (doppio)
Abbiategrosso	MI	MI	Milano
Arese	MI	MI	Milano
Acquanegra Cremonese	CR	CR	Cremona
Abbadia Lariana	LC	LC	Lecco
Abbadia Cerreto	LO	LO	Lodi
Agrate Brianza	MB	MB	Monza e della Brianza

8 righe restituite in 0,00 secondi

[Scarica](#)

# Comuni e Province: Schema E/R nei 3 casi



# Consigli


- **La relazione M:N (molti a molti) E' DA EVITARE!!**
- **Non mischiare LEFT JOIN e RIGHT JOIN (io preferisco usare LEFT)**
- **Usare il più possibile le Integrità Referenziali**

# L'Applicativo

---

# Creazione Applicativo

## Crea un'applicazione




Nome


D12 Comuni

Aspetto






















Vita, Menu laterale



### Pagine



Aggiungi pagina

	 Home	Vuota	<div>Modifica </div>
	 Comuni	Report interattivo con form ( d12_comuni )	<div>Modifica </div>
	 Province	Report interattivo con form ( d12_province )	<div>Modifica </div>
	 Comuni2	Report interattivo con form ( d12_comuni2 )	<div>Modifica </div>
	 Provincia2	Report interattivo con form ( d12_provincia2 )	<div>Modifica </div>
	 Comuni3	Report interattivo con form ( d12_comuni3 )	<div>Modifica </div>
	 Provincia3	Report interattivo con form ( d12_provincia3 )	<div>Modifica </div>

Annulla

Crea applicazione



## D04\_COMUNI Aggiungo Provincia che non c'è in tabella Padre

### Comuni

	Comune ↑	Provincia Cod
	Abbadia Cerreto	LO
	Abbadia Lariana	LC
	Abbiategrosso	MI
	Acquafredda	BS
	Acquanegra Cremona	CR
	Acquanegra sul Chiese	MN
	Agra	VA
	Agrate Brianza	MB
	Alagna	PV
	Albano Sant'Alessandro	BG
	Albaredo per San Martino	SO
	Albavilla	CO

Comuni

Comune

Roma

Provincia Cod

RM

Annulla













Crea



## D04\_PROVINCIA Elimino Provincia esistente in tabella Figlia

Province

Q ▾ Vai Azioni ▾

	Provincia Cod ↑≡	Provincia
	BG	Bergamo
	BS	Brescia
	CO	Como
	CR	
	LC	
	LO	
	MB	
	MI	
	MN	
	PV	
	SO	
	VA	Varese

Province

Provincia Cod  
BG

Provincia  
Bergamo

Annulla Elimina

Eseguire questa azione di eliminazione?

Annulla Elimina

Applica modifiche

















## D04\_COMUNE2 Posso mettere un valore Nullo in Provincia

Comuni2

Q ▾ Vai Azioni ▾

	Comune ↑	Provincia
	Abbadia Cerreto	Lodi
	Abbadia Lariana	Lecco
	Abbiategrosso	Milano
	Acquafredda	
	Acquanegra Cremonese	
	Acquanegra sul Chiese	
	Agra	
	Agrate Brianza	inza
	Alagna	
	Albano Sant'Alessandro	
	Albaredo per San Marco	
	Albavilla	Como

Comuni2

Comune  
Abbadia Cerreto

Provincia ▾













Annulla Elimina Applica modifiche



## D03\_PROVINCIA2 Non posso eliminare una Provincia presente nella tabella Figlia

Provincia2

Q ▾ Vai Azioni ▾

	Provincia ↑
	Bergamo
	Brescia
	Como
	
	
	
	
	
	
	
	
	Varese

Provincia2

**Si è verificato 1 errore**

- ORA-02292: restrizione di integrità violata (AT0.SYS\_C0076682) - chiave figlia trovata













Annula Elimina Applica modifiche



## D04\_COMUNE3 Non Posso mettere un valore Nullo in Provincia

Comuni3

Q ▾ Vai Azioni ▾ Crea

	Comune ↑≡	Provincia
	Abbadia Cerreto	Lodi
	Abbadia Lariana	Lecco
	Abbiategrosso	
	Acquafredda	
	Acquanegra Cremonese	
	Acquanegra sul Chiese	
	Agra	
	Agrate Brianza	
	Alagna	
	Albano Sant'Alessandro	
	Albaredo per San Marco	
	Albavilla	Como

Comuni3

Comune  
Abbadia Cerreto

Provincia  
▾

Specificare un valore per Provincia.

Annula Elimina Applica modifiche

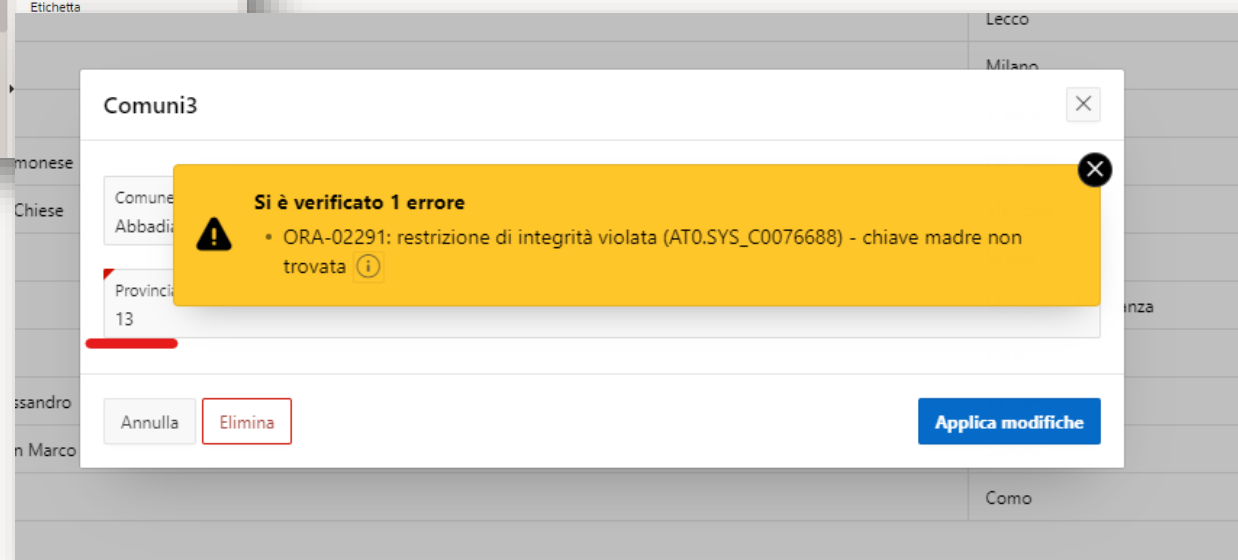
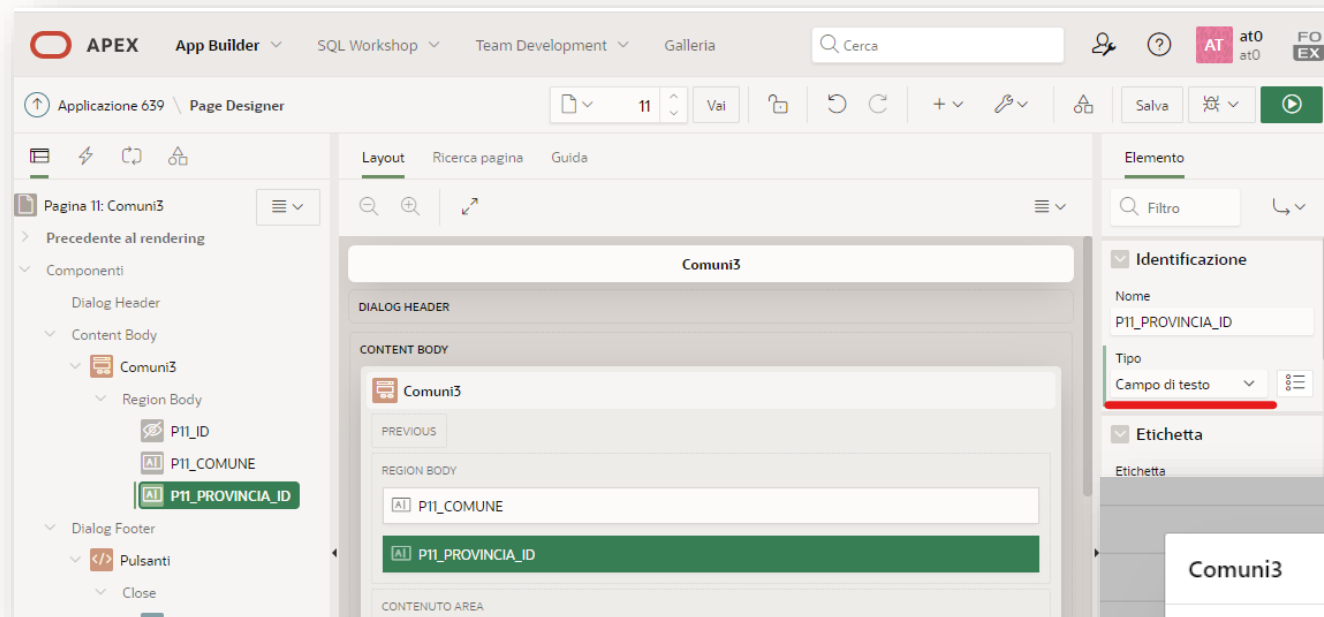
Si è verificato 1 errore

- Specificare un valore per Provincia.

1 - 12



## D04\_COMUNE3 Non Posso mettere un valore che non esiste nella tabella Padre



# Quick SQL – do3 Gestionale

---

# Overview

1. CAPIRE il problema
2. Creare gli OGGETTI relativi ai dati
3. Creare applicativo APEX

# Capire il problema

---

## CLIENTE

CLIENTE

ZONA

---

---

## VENDITA

PRODOTTO

CLIENTE

DATA

QUANTITA

---

---

## PRODOTTO

PRODOTTO

TIPO PRODOTTO

PREZZO ACQUISTO

PREZZO VENDITA

---

---

## TIPO PRODOTTO

TIPO PRODOTTO

---

---

## FORNITORE

FORNITORE

ZONA

---

---

## ACQUISTO

PRODOTTO

FORNITORE

DATA

QUANTITA

---

# Quick SQL

## Informazioni su Quick SQL

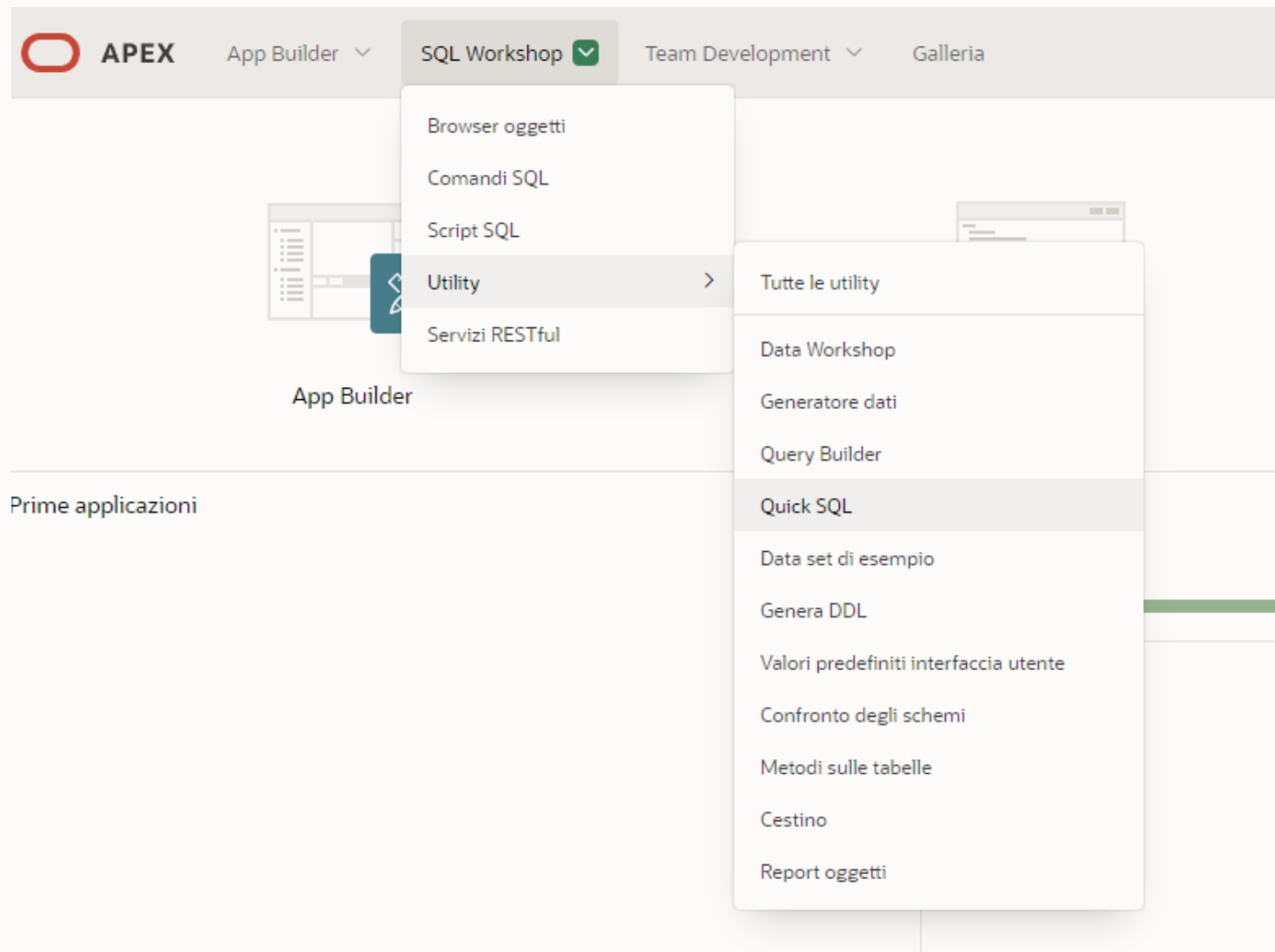
- Consente di generare rapidamente il codice SQL necessario alla creazione di un modello dati relazionale **da un documento di testo indentato**.

## Casi d'uso principali

- Creazione rapida di modelli dati potenti
- Salvataggio della sintassi abbreviata Quick SQL
- Generazione facilitata di dati casuali
- Apprendimento della sintassi di creazione tabelle, selezione, inserimento, indicizzazione, trigger, package PL/SQL e visualizzazione SQL con gli esempi forniti



# Quick SQL



# Quick SQL

APEX

App Builder

SQL Workshop

Team Development

Galleria

Cerca

at0

at0

FO

EX

Utility

Quick SQL

Quick SQL

Cancella

Carica

Guida

Salva modello

Genera SQL

1

2

3

4

5

6

7

Persona

nome vc20

cognome vc20

data nascita d

email vc50

telefono vc20

Sintassi Abbreviata Quick SQL

SQL

Impostazioni

Scarica

Salva script SQL

Riesamina ed esegui

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

-- create tables

create table persona (

id

nome

cognome

data\_nascita

email

telefono

)

;

-- load data

-- Generated by Q

/\*

Persona

nome vc20

cognome vc20

data nascita d

email vc50

telefono vc20

# settings = { semantics: "CHAR", language: "EN", APEX: true }

\*/

Codice DDL creato automaticamente



# Quick SQL

APEX App Builder SQL Workshop Team Development Galleria

Cerca

Utility \ Quick SQL

Quick SQL Cancella Carica Guida Salva modello Genera SQL SQL Impostazioni Scarica Salva script SQL Riesamina ed esegui

```
1 Persona
2 nome vc20
3 cognome vc20
4 data nascita d
5 email vc50
6 telefono vc20
7
```

```
1 -- create tables
2 create table persona (
3     id number generated by default on null as identity
4     constraint persona_id_pk primary key,
5     nome varchar2(20 char),
6     cognome varchar2(20 char),
7     data_nascita date,
8     email varchar2(50 char),
9     telefono varchar2(20 char)
```

Tipi di dati abbreviati

Tipi di dati	
num, number	NUMBER
int, integer	INTEGER
d, date	DATE
ts, timestamp	TIMESTAMP
tstz, tswtz, timestamp with local time zone	TIMESTAMP WITH LOCAL TIMEZONE
char, vc, varchar, varchar2, string	VARCHAR2(4000)
vcNNN	VARCHAR2(NNN) NNN identifica un numero compreso tra 1 e 32767.
vc(NNN)	VARCHAR2(NNN) NNN identifica un numero compreso tra 1 e 32767.
vc32k	VARCHAR2(32767)

ore 17, 2022 21:36:30

age: "EN", APEX: true }



# Quick SQL

Quick SQL

Cancella Carica Guida Salva modello Genera SQL

1 Persona

2 nome vc20 /nn

3 cognome vc20

4 codice\_fiscale /nn /unique

5 data\_nascita d

6 email vc50

7 telefono vc20

8

SQL

Impostazioni Scarica Salva script SQL Riesamina ed esegui

1 -- create tables

2 create table persona (

3 id number generated by default on null as identity

4 constraint persona\_id\_pk primary key,

5 nome varchar2(20 char) not null,

6 cognome varchar2(20 char),

7 codice\_fiscale varchar2(4000 char)

8 constraint persona\_codice\_fiscale\_unq unique not null,

9 data\_nascita date,

10 email varchar2(50 char),

11 telefono varchar2(20 char)

Informazioni

Tipi di dati

Direttive di tabella

Direttive di colonna

Viste

Impostazioni

Esempi

Direttive di colonna

## Quick SQL

```
tipo_prodotto  
tipo_prodotto vc50 /nn /unique  
gruppo vc50 /nn
```

```
create table tipo_prodotto (  
    id number generated by default on null as identity  
    constraint tipo_prodotto_id_pk primary key,  
    tipo_prodotto varchar2(50 char)  
    constraint tipo_prodotto_tipo_prodotto_unq unique not null,  
    gruppo varchar2(50 char) not null  
);
```

- vc50 -> varchar2(50)
- /nn -> not null
- /unique -> indice univoco

## Quick SQL

tipo prodotto

```
tipo_prodotto vc50 /nn /unique
gruppo vc50 /nn
```

prodotto

```
prodotto vc50 /nn /unique
tipo_prodotto_id /nn
prezzo acquisto num /nn
prezzo vendita num /nn
```

## Vincolo di Integrità Referenziale

```
create table tipo_prodotto (  
    id  
        |  
        |  
        |  
        |  
        |  
        |  
        |  
        |  
        |  
        |  
    tipo_prodotto  
        |  
        |  
        |  
        |  
        |  
        |  
        |  
        |  
        |  
        |  
    gruppo  
  
);  
;  
  
create table prodotto (  
    id  
        |  
        |  
        |  
        |  
        |  
        |  
        |  
        |  
        |  
        |  
    tipo_prodotto_id  
        |  
        |  
        |  
        |  
        |  
        |  
        |  
        |  
        |  
        |  
    prodotto  
        |  
        |  
        |  
        |  
        |  
        |  
        |  
        |  
        |  
        |  
    prezzo_acquisto  
    prezzo_vendita  
  
);  
;
```

number generated by default on null as identity  
constraint tipo\_prodotto\_id\_pk primary key,  
varchar2(50 char)  
constraint tipo\_prodotto\_tipo\_prodott\_unq unique not null,  
varchar2(50 char) not null

number generated by default on null as identity  
constraint prodotto\_id\_pk primary key,  
number  
constraint prodotto\_tipo\_producto\_id\_fk  
references tipo\_producto on delete cascade not null,  
varchar2(50 char)  
constraint prodotto\_producto\_unq unique not null,  
number not null,  
number not null

Creato un indice per la chiave esterna `tipo_prodotto_id`

```
-- table index
create index prodotto_i1 on prodotto (tipo prodotto id);
```

# Quick SQL

## cliente

```
cliente vc50 /nn /unique  
zona vc10 /nn
```

## fornitore

```
fornitore vc50 /nn /unique  
zona vc50 /nn
```

```
create table cliente (
```

```
  id
```

```
  cliente
```

```
  zona
```

```
)
```

```
;
```

number generated by default on null as identity

constraint cliente\_id\_pk primary key,

varchar2(50 char)

constraint cliente\_cliente\_unq unique not null,

varchar2(10 char) not null

```
create table fornitore (
```

```
  id
```

```
  fornitore
```

```
  zona
```

```
)
```

```
;
```

number generated by default on null as identity

constraint fornitore\_id\_pk primary key,

varchar2(50 char)

constraint fornitore\_fornitore\_unq unique not null,

varchar2(50 char) not null

# Quick SQL

```
acquisto
  data d /nn
  fornitore_id /nn
  prodotto_id /nn
  quantita num /nn
```

```

vendita
  data d /nn
  cliente_id /nn
  prodotto_id /nn
  quantita num /nn

```

```

create table acquisto (
    id                number generated by default on null as identity
                    constraint acquisto_id_pk primary key,
    fornitore_id      number
                    constraint acquisto_fornitore_id_fk
                    references fornitore on delete cascade not null,
    prodotto_id       number
                    constraint acquisto_prodotto_id_fk
                    references prodotto on delete cascade not null,
    data              date not null,
    quantita          number not null
)
;

-- table index
create index acquisto_i1 on acquisto (fornitore_id);
create index acquisto_i142 on acquisto (prodotto_id);

create table vendita (
    id                number generated by default on null as identity
                    constraint vendita_id_pk primary key,
    cliente_id        number
                    constraint vendita_cliente_id_fk
                    references cliente on delete cascade not null,
    prodotto_id       number
                    constraint vendita_prodotto_id_fk
                    references prodotto on delete cascade not null,
    data              date not null,
    quantita          number not null
)
;

-- table index
create index vendita_i1 on vendita (cliente_id);
create index vendita_i192 on vendita (prodotto id);

```





# Quick SQL

## tipo\_prodotto

```
tipo_prodotto vc50 /nn /unique  
gruppo vc50 /nn
```

## prodotto

```
prodotto vc50 /nn  
tipo_prodotto_id /nn  
prezzo_acquisto num /nn  
prezzo_vendita num /nn
```

## cliente

```
cliente vc50 /nn  
zona vc10 /nn
```

## fornitore

```
fornitore vc50 /nn  
zona vc50 /nn
```

## acquisto

```
data d /nn  
fornitore_id /nn  
prodotto_id /nn  
quantita num /nn
```

## vendita

```
data d /nn  
cliente_id /nn  
prodotto_id /nn  
quantita num /nn
```

```
-- create tables  
create table d14_tipo_prodotto (  
  id number generated by default on null as identity  
  tipo_prodotto varchar2(50 char) constraint d14_tipo_prodotto_id_pk primary key,  
  gruppo varchar2(50 char) constraint d14_tipo_prodotto_tipo_prodo_unq unique not null,  
  )  
  
create table d14_prodotto (  
  id number generated by default on null as identity  
  tipo_prodotto_id number  
  prodotto varchar2(50 char) constraint d14_prodotto_tipo_prodotto_fk  
  references d14_tipo_prodotto (tipo_prodotto_id) not null,  
  prezzo_acquisto number not null, constraint d14_prodotto_prodotto_unq unique not null,  
  prezzo_vendita number not null  
  )  
  
-- table index  
create index d14_prodotto_i1 on d14_prodotto (tipo_prodotto_id);  
  
create table d14_cliente (  
  id number generated by default on null as identity  
  cliente varchar2(50 char) constraint d14_cliente_id_pk primary key,  
  zona varchar2(10 char) constraint d14_cliente_cliente_unq unique not null,  
  )  
  
create table d14_fornitore (  
  id number generated by default on null as identity  
  fornitore varchar2(50 char) constraint d14_fornitore_id_pk primary key,  
  )  
  constraint d14_fornitore_fornitore_unq unique not null,
```

- **d14**: prefisso oggetti
- **Limita**: evita che cancellando la testata vengano eliminati tutti i dettagli

### Impostazioni

Tabella

Prefisso oggetti

d14

?

Schema

?

Su eliminazione

A catena

Limita

Imposta su null

?

Compressione

?

Genera inserimenti

?

Chiave primaria

Aggiungi chiave primaria

?

Metodo inserimento dati

Colonna identità

?

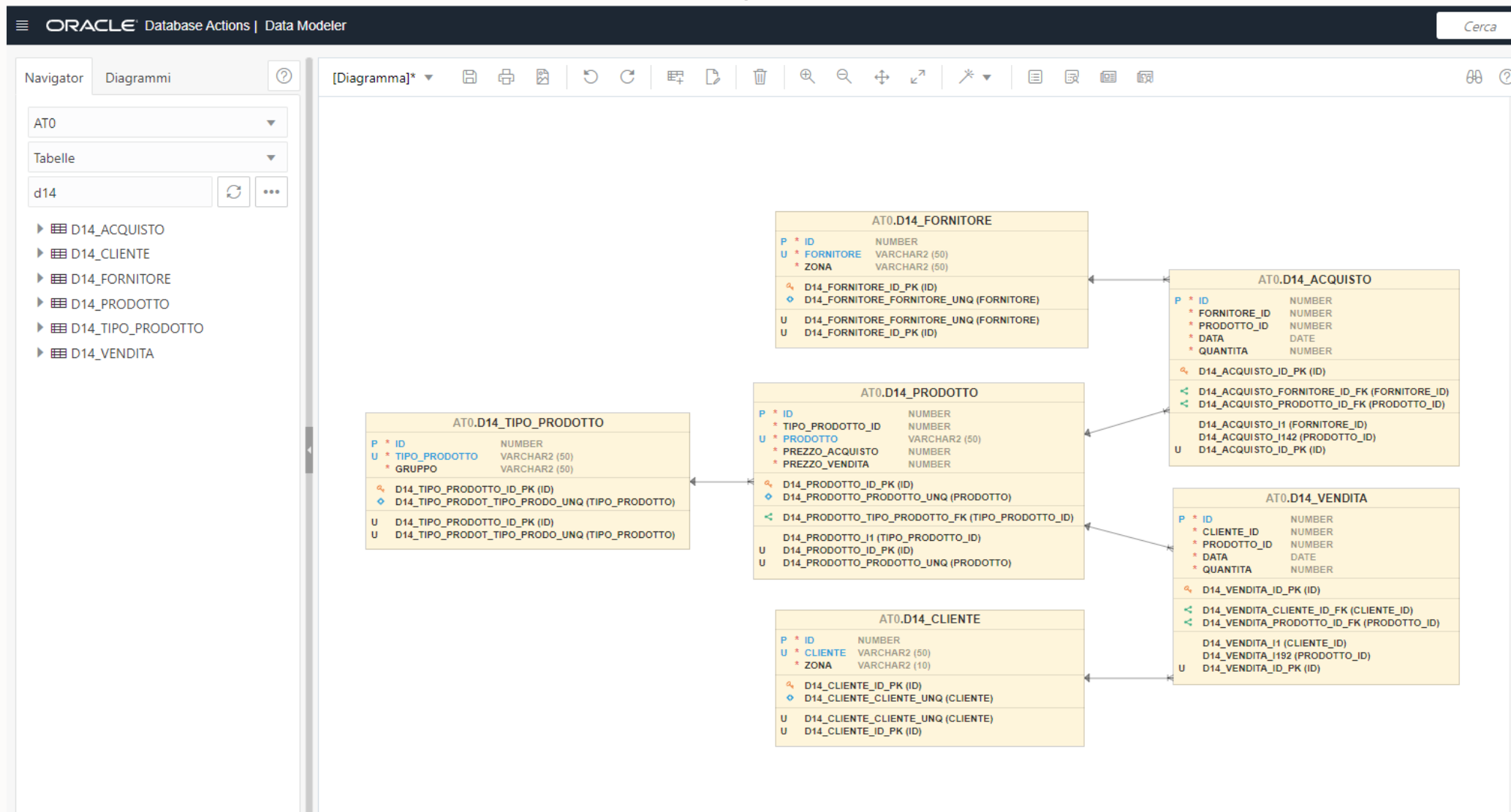
Annulla

Reimposta

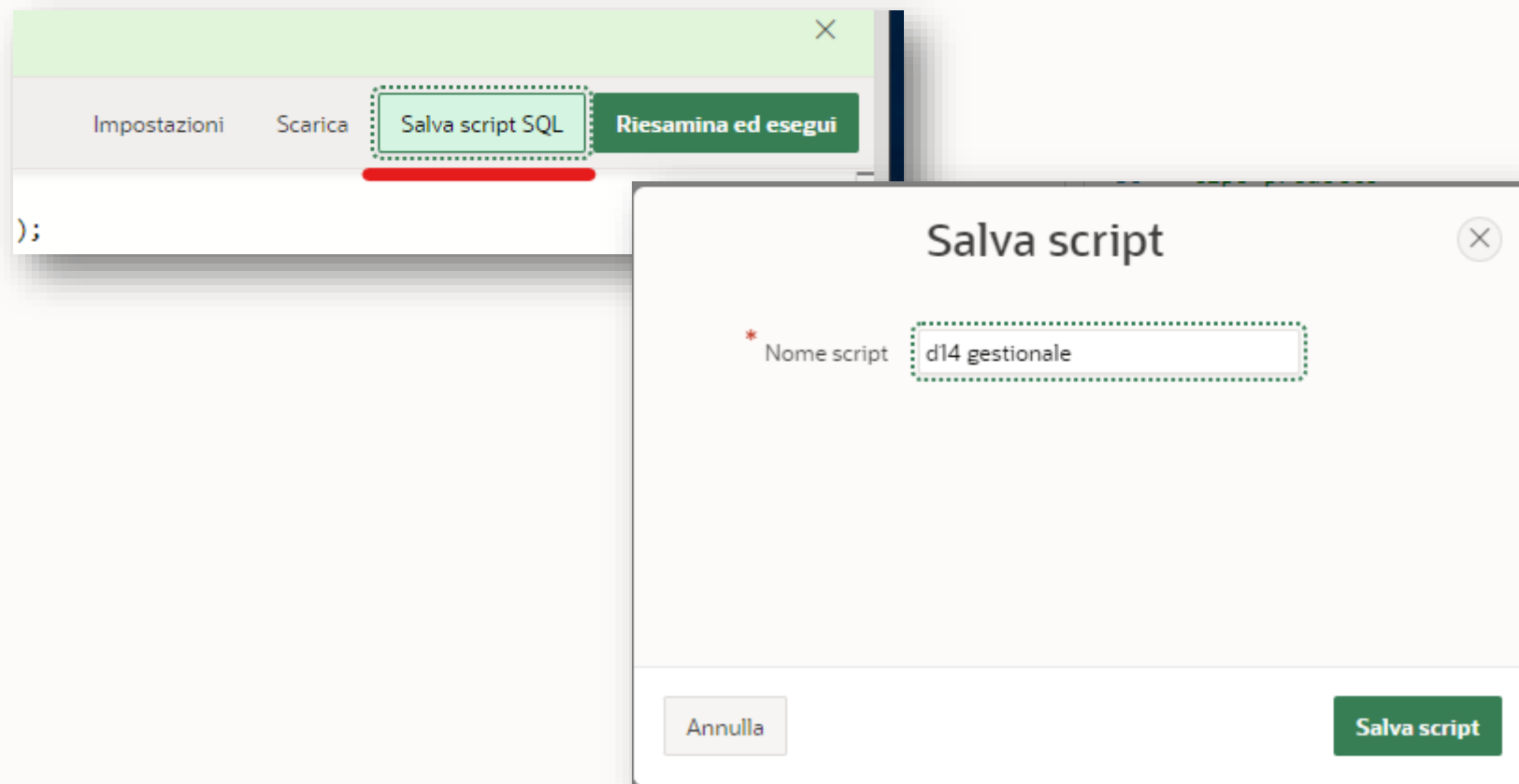
Salva modifiche

```
# settings = { prefix: "d14", onDelete: "RESTRICT", semantics: "CHAR", language: "EN", APEX: true }
```

# Schema E/R



# Quick SQL



# Script SQL

Modifica	Proprietario	Nome	Data creazione	Autore aggiornamento	Aggiornato ↓	Byte	Risultati	Esegui
	ATO	d14 gestionale	8 minuti fa	ATO	85 secondi fa	4.168	0	

Script SQL \ Risultati

Script: d14 gestionale

Stato: Completato

Crea applicazione

Modifica script

Vista: 

Dettaglio

Riepilogo

Righe: 15

Vai

Numero ↑	Tempo trascorso	Istruzione	Feedback	Righe
	0,17	create table d14_tipo_prodotto ( id	Tabella creata.	0
	0,02	create table d14_prodotto ( id	Tabella creata.	0
	0,03	create index d14_prodotto_i1 on d14_prodotto (tipo_prodotto_	Indice creato.	0
	0,01	create table d14_cliente ( id	Tabella creata.	0
	0,02	create table d14_fornitore ( id	Tabella creata.	0
	0,01	create table d14_acquisto ( id	Tabella creata.	0
	0,01	create index d14_acquisto_i1 on d14_acquisto (fornitore_id)	Indice creato.	0
	0,00	create index d14_acquisto_i142 on d14_acquisto (prodotto_id)	Indice creato.	0
	0,01	create table d14_vendita ( id	Tabella creata.	0
	0,00	create index d14_vendita_i1 on d14_vendita (cliente_id)	Indice creato.	0
	0,00	create index d14_vendita_i192 on d14_vendita (prodotto_id)	Indice creato.	0

carica



# Creazione Applicativo

Crea applicazione		Modifica script
Feedback	Righe	
Tabella creata.	0	

## Crea applicazione da script



\* Schema  ?

Nome script **d14 gestionale**

Le tabelle riportate di seguito in questo script verranno utilizzate per le pagine di report e form predefinite nella Creazione guidata applicazione.

### Nome tabella

D14\_ACQUISTO

D14\_CLIENTE

D14\_FORNITORE

D14\_PRODOTTO

D14\_TIPO\_PRODOTTO

D14\_VENDITA

Annulla

Crea applicazione >

