

(A) CONOSCENZA TERMINOLOGICA

Dare una breve descrizione dei termini introdotti:

- Vincoli intra-relazionali
- Vincoli di integrità
- Comando **CREATE**
- Comando **ALTER**
- Comando **DROP**
- Comando **SHOW TABLES**
- Comando **DESCRIBE**

(B) CONOSCENZA E COMPETENZA

Rispondere alle seguenti domande producendo anche qualche esempio

B1) Conoscenza

1. Cosa vuol dire *creare una tabella*?
2. Cosa significa *modificare una tabella*?
3. Cosa significa *eliminare una tabella*?
4. Cosa è una *chiave candidata*?
5. Cosa è una *chiave esterna*?

B2) Competenza

1. Qual è la sintassi per *creare una tabella*?
2. Qual è la sintassi per *modificare una tabella*?
3. Qual è la sintassi per *eliminare una tabella*?
4. Qual è la sintassi per *elencare le tabelle*?
5. Qual è la sintassi per *visualizzare uno schema*?

(C) ESERCIZI DI COMPrensIONE

1. Indicare nella seguente tabella, la sintassi dei comandi indicati.

Operazione	Comando
Creazione tabella	
Elenco tabelle presenti	
Eliminazione tabella	
Modifica tabella	
Schema della tabella	

2. Il comando **CREATE TABLE** *tab* (*lista attributi*) serve a definire lo di una relazione, a creare una istanza dell'entità *tab* e a specificare e Degli attributi.
3. La clausola su un attributo, nel comando **CREATE TABLE**, specifica che per esso non sono possibili valori nulli, cioè che il campo non può restare vuoto.
4. La clausola su un attributo, nel comando **CREATE TABLE**, specifica che il campo verrà usato come chiave primaria.
5. La clausola su un attributo, nel comando **CREATE TABLE**, specifica che ogni istanza dell'entità avrà quel dato valore come valore predefinito.
6. La clausola su un attributo, nel comando **CREATE TABLE**, specifica che esso può essere usato come chiave, ossia è utilizzabile come ulteriore chiave primaria.
7. Il comando **ALTER TABLE** *tab* *comando* modifica lo schema di una relazione, consentendo di o attributi o di modificarne il
8. La clausola serve ad aggiungere una colonna e il suo dominio, mentre la clausola serve ad eliminare una colonna. L'aggiunta di una colonna in una data posizione, può essere fatta con la clausola
9. La clausola consente di modificare una colonna e/o il suo dominio, mentre la clausola consente la modifica del solo
10. Il comando **DROP TABLE** serve a una tabella dal database.
11. Per visualizzare l'elenco delle tabelle presenti nel DB corrente, si usa il comando, mentre per visualizzare lo schema si usa il comando
12. Per garantire la validità di certi dati, si usa la clausola applicata ad un'espressione booleana.
13. Se un campo non può essere lasciato vuoto, va dichiarato con il vincolo non è possibile non inserire un valore.
14. Quando un campo deve essere scelto come chiave primaria, va dichiarato con il vincolo In tal modo, questo campo individua in modo la riga.
15. La chiave esterna di una tabella viene indicata con il vincolo
16. Se un campo deve avere un valore non ripetibile in altri record, occorre usare il vincolo
17. Scrivere il comando SQL per creare ciascuna delle seguenti tabelle:
 - a. tabella CLIENTI:

Campo	Tipo	Null	Key	Default
id_cliente	INT(6)	NO	PRI	
cognome	CHAR (20)	NO		
nome	CHAR (15)	NO		
indirizzo	CHAR (30)	YES		NULL

- b. tabella ACQUISTI

Campo	Tipo	Null	Key	Default
id_acquisto	INT(6)	NO	PRI	
importo	DOUBLE	YES		NULL
acquisto	DATE	YES		NULL

- c. tabella NAZIONI

Campo	Tipo	Null	Key	Default
id_nazione	INT(4)	NO	PRI	
denominazione	CHAR (20)	YES		NULL
moneta	CHAR (15)	YES		NULL
temperatura	DOUBLE	YES		NULL

- d. tabella PRODOTTI

Campo	Tipo	Null	Key	Default
id_prodotto	INT(9)	NO	PRI	
descrizione	CHAR (20)	YES		NULL
consegna	DATE	NO		
prezzo	DOUBLE	YES		NULL

- e. tabella SOCIETA'

Campo	Tipo	Null	Key	Default
id_societa	INT(3)	NO	PRI	
nome	CHAR (20)	NO		
telefono	CHAR (15)	YES		NULL

18. Scrivere il comando SQL per modificare la tabella dell'esercizio 8a. nel seguente modo:

Campo	Tipo	Null	Key	Default
id_cliente	INT(6)	NO	PRI	
cognome	CHAR (20)	NO		
nome	CHAR (15)	NO		
fatturato	DOUBLE	YES		NULL
indirizzo	CHAR (30)	YES		NULL

19. Scrivere il comando SQL per modificare la tabella dell'esercizio 8e. nel seguente modo:

Campo	Tipo	Null	Key	Default
id_societa	INT(3)	NO	PRI	
nome	CHAR (20)	NO		
indirizzo	CHAR (20)c	YES		NULL
citta	CHAR (15)	YES		NULL
telefono	CHAR (15)	YES		NULL

20. Scrivere il comando SQL che sulla tabella dell'esercizio 8c:

- elimini l'attributo *temperatura* con valore di default uguale a 0;
- aggiunga l'attributo *clima* come stringa di 10 caratteri;
- modifichi l'attributo *denominazione* consentendo valori non nulli.
- crei un indice di nome *indicenazione*;

21. Scrivere i comando SQL per creare le seguenti tabelle in relazione (le chiavi esterne sono indicate in corsivo e con la sigla
- MUL**
-):

- a. tabella CLIENTI

Campo	Tipo	Null	Key	Default
IDCliente	INT(3)	NO	PRI	
cognome	CHAR (20)	NO		
nome	CHAR (15)	NO		
indirizzo	CHAR (20)			NULL
telefono	CHAR (15)			NULL

- b. tabella ORDINI

Campo	Tipo	Null	Key	Default
IDOrdine	INT (3)	NO	PRI	
IDCliente	INT (3)	NO	MUL	
IDPacchetto	CHAR (25)	NO	MUL	
data_acquisto	DATE	NO		

c. tabella NAZIONI

Campo	Tipo	Null	Key	Default
IDNazione	CHAR (25)	NO	PRI	
denominazione	CHAR (40)	NO		
moneta	CHAR (10)			NULL
clima	CHAR (15)			NULL

d. tabella PACCHETTI

Campo	Tipo	Null	Key	Default
IDPacchetto	CHAR (25)	NO	PRI	
descrizione	CHAR (40)	NO		
modalità	CHAR (10)	NO		
prezzo	DOUBLE	NO		
<i>IDNazione</i>	CHAR (25)	NO	MUL	
<i>IDOrg</i>	INTERO (3)	NO	MUL	

e. tabella ORGANIZZAZIONI

Campo	Tipo	Null	Key	Default
IDOrg	INT (3)	NO	PRI	
nome_org	CHAR (25)	NO		
telefono	CHAR (15)			NULL

22. Scrivere 1 comando SQL per creare le seguenti tabelle in relazione (le chiavi esterne sono indicate in corsivo):

<i>IDPaziente</i>	cognome	nome	indirizzo	telefono	<i>IDMedico</i>
1	bianchi	ennio	via roma 1	56476745	100
2	pandolfi	simone	via milano 16	76567767	105
3	pacifici	eda	l.go paci 4	65565756	100
4	arduini	ada	c.so vittorio	86796866	103

<i>IDMedico</i>	cognome	nome	indirizzo	telefono	specialità
100	simoni	eduardo	via milano 35	6576545	otorino
101	carli	lucio	l.go aironi 6	5675657	odontoiatra
102	balzani	giuseppe	v.le meduse 48	8789789	neurologo
103	peri	antonio	c.so garibaldi	5685787	otorino

23. Scrivere i comando SQL per creare le seguenti tabelle in relazione (le chiavi esterne sono indicate in corsivo):

- a. tabella CORSI
b. tabella STUDENTI

<i>IDCorso</i>	nomecorso	aula
A10	matematica1	1
A21	fisica	2
A11	matematica2	3

<i>IDStudente</i>	cognome	nome	indirizzo	telefono	<i>IDCorso</i>
50223	alessi	simone	via milano 35	6576545	A10
50123	celestini	carlo	l.go aironi 6	5675657	A10
61234	monti	biagio	v.le meduse 48	8789789	A11
60003	pieri	anna	c.so garibaldi	5685787	A20

24. Scrivere un comando SQL per creare una tabella con attributi *id* non nullo avente il ruolo di chiave primaria, *nome* di 25 caratteri, che non ammetta duplicati, e *squadra* di 25 caratteri.

25. Scrivere il comando SQL per creare le seguenti tabelle in relazione tra loro (le chiavi esterne sono indicate in corsivo):

- a. tabella CLASSI
b. tabella STUDENTI

<i>IDClasse</i>	Classe	Specializzazione
13	IIIA	Elettronica
23	IIIB	Informatica
34	IVB	Informatica

<i>IDStudente</i>	Cognome	Nome	DataNascita	Classe
231	Bianchi	Emanuele	12/02/1984	13
233	Rosati	Anna	15/04/1984	13
234	Vitale	Ernesto	05/05/1983	34

c. tabella VOTI

<i>IDVoto</i>	<i>IDStudente</i>	Voto	DataVoto
151	231	30	15/04/2004
152	233	25	05/05/2003
153	234	27	12/12/2003

26. Scrivere il comando **SQL** per creare la tabella CATEGORIA a fianco. Poi, modificare la struttura nella seguente, facendo in modo che l'attributo *iva* possa assumere valori compresi tra 10 e 20:

Campo	Tipo	Null	Key	Default
id_cat	INTERO(2)	NO	PRI	
Descrizione	CHAR (40)			NULL
iva	INTERO (2)	NO		

27. Stabilire se la seguenti istruzioni **SQL** contengono errori e, in caso affermativo, riscriverle correttamente.
- CREATE TABLE** Nominativi
(
 Nome, **VARCHAR2 (25),**
 Cognome, **VARCHAR2 (25),**
 DataNascita **DATA,**
);
 - CREATE TABLE** Calciatori
(
 IDCalciatore **CHAR (2),**
 Cognome **VARCHAR2(25), NOT NULL**
 Nome **VARCHAR2(15) NOT NULL**
 PRIMARY KEY (IDCalciatore)
);
 - CREATE TABLE** Vetture
(
 IDVettura **PRIMARY KEY,**
 Modello **VARCHAR (25),**
 Marca **VARCHAR (15),**
 Targa **VARCHAR2 (10) UNIQUE**
);
 - CREATE TABLE** Dipendenti
(
 IDDipendente **INTEGER (3) PRIMARY KEY,**
 Cognome **VARCHAR2(25),**
 Nome **VARCHAR2(25),**
 NumeroFigli **INTEGER DEFAULT '0',**
);
28. Stabilire in quali dei seguenti casi la clausola **CHECK** è usata correttamente:
- cap **INTEGER CHECK (capo IN (0,1))**
 - sezione **CHAR (1) CHECK (sezione IN (A,B,C))**
 - alim **CHAR (15) CHECK (alim IN('Benzina','Diesel'))**
 - classe **CHAR (1) CHECK (classe IN (1,2,3,4,5))**
 - opzione **INTEGER(1) CHECK (opzione>=1 AND opzione <=5)**
29. Descrivere il significato della seguente dichiarazione di tabella:
- CREATE TABLE** Calciatori
(
 IDCalciatore **CHAR(3) PRIMARY KEY,**

 Nome **VARCHAR2(20),**
 Cognome **VARCHAR2(20)**
);
 - CREATE TABLE** Persona
(
 IDPersona **INTEGER(3) PRIMARY KEY,**
 Nome **VARCHAR2(25) UNIQUE,**
);
 - CREATE TABLE** Persona
(
 IDPersona **INTEGER(3) PRIMARY KEY,**
 Nome **VARCHAR(25) NOT NULL,**
 Cognome **VARCHAR(25) NOT NULL,**
 UNIQUE (Nome, Cognome)
);

(D) ESERCIZI DI APPLICAZIONE

- Esercizio risolto.** Creare un database *Cattedre* che consenta di gestire i docenti e le classi di un sistema scolastico, scrivere gli schemi necessari e i comandi SQL per la creazione delle tabelle relative. Supponiamo di considerare i seguenti schemi:
 - CLASSI (IDClasse, Classe, Sezione, Specializzazione)
 - PROFESSORI (IDProfessore, Cognome, Nome, ClasseConcorso)
 - INSEGNA (IDProfessore, IDClasse, Ore)
in cui le chiavi primarie sono sottolineate, mentre quelle esterne sono in corsivo.

a. tabella CLASSI:

Campo	Tipo	Null	Key	Default
IDClasse	INT(2)	NO	PRI	
Classe	INT (1)	NO		
Sezione	CHAR(1)	NO		
Specializzazione	CHAR(15)	NO		

b. tabella PROFESSORI

Campo	Tipo	Null	Key	Default
IDProfessore	INT(2)	NO	PRI	
Cognome	CHAR(15)	NO		
Nome	CHAR(15)	NO		
ClasseConcorso	CHAR(5)	NO		

c. tabella INSEGNA

Campo	Tipo	Null	Key	Default
IDProfessore	INT(2)	NO	PRI	
IDClasse	INT(2)	NO	PRI	
Ore	INT(2)	NO		

Supponendo di operare in un ambiente interattivo, come **MySQL**, scriviamo la seguente serie di comandi:

```
CREATE DATABASE cattedre;           /* creazione del database */
USE cattedre;
CREATE TABLE classi                /* creazione tabella CLASSI */
→ (  IDClasse          INTEGER (2)  NOT NULL,
→   Classe             INTEGER (1)  NOT NULL,
→   Sezione            CHAR (1)     NOT NULL,
→   Specializzazione   CHAR (15)    NOT NULL,
→   PRIMARY KEY (IDClasse)
→ );
CREATE TABLE professori           /* creazione tabella PROFESSORI */
→ (  IDProfessore      INTEGER (2)  NOT NULL,
→   Cognome            CHAR (15)    NOT NULL,
→   Nome               CHAR (15)    NOT NULL,
→   Classeconcorso     CHAR (5)     NOT NULL,
→   PRIMARY KEY (IDProfessore)
→ );
CREATE TABLE insegna              /* creazione tabella INSEGNA */
→ (  IDProfessore      INTEGER(2)   NOT NULL,
→   IDClasse           INTEGER(2)   NOT NULL,
→   Ore                INTEGER(2)   NOT NULL,
→   PRIMARY KEY (IDProfessore, IDClasse),
→   FOREIGN KEY (IDProfessore) REFERENCES professori (IDProfessore),
→   FOREIGN KEY (IDClasse) REFERENCES classi (IDClasse)
→ );
```

2. Scrivere il comando per mostrare l'elenco dei campi presenti nel tracciato record di una tabella di un database, dopo averlo selezionato.
3. Scrivere il comando per creare un DB che consenta di memorizzare i dati relativi a squadre di calcio e ai tifosi di ciascuna squadra.
4. Creare un DB in cui sia possibile memorizzare i dati di dipendenti di un'azienda (*cognome*, *nome* e *mansioni*). I dipendenti sono suddivisi in gruppi di lavoro caratterizzati, ciascuno da un *ID* progressivo, dal *cognome* del dipendente appartenente e da un campo *capogruppo* che se vale 1 indica che il dipendente è capogruppo, altrimenti è posto a 0.