# Esercizi

1. Con riferimento al database prenotazioni scrivi la query che trova tutte le regioni che appartengono all’area geografica Sud
2. Con riferimento al database prenotazioni scrivi la query che trova tutte le informazioni relative alle prenotazioni che hanno importo superiore a 200
3. Con riferimento al database prenotazioni scrivi la query che trova le prenotazioni con importo superiore a 150 e caparra inferiore a 50
4. Con riferimento al database prenotazioni scrivi la query che trova tutti i nomi e cognomi dei clienti ordinati per età
5. Con riferimento al database prenotazioni scrivi la query che trova le date di arrivo per gli hotel 2 stelle ordinate in ordine cronologico dalla più recente alla più vecchia
6. Con riferimento al database prenotazioni scrivi la query che trova gli id cliente, i nomi e i cognomi dei clienti che hanno cognome Rossi o Bianchi ordinati per cognome crescente e per data di nascita decrescente
7. Con riferimento al database prenotazioni scrivi la query che trova la regione di provenienza dei clienti che hanno prenotazioni con importo superiore a 150 e caparra inferiore a 50
8. Con riferimento al database prenotazioni scrivi la query che trova città di residenza, nome e cognome dei clienti che hanno prenotazioni con caparra compresa fra 50 e 90 e importo compreso fra 140 e 240
9. Con riferimento al database prenotazioni scrivi la query che trova il nome della città di appartenenza dei clienti che hanno prenotato le camere 11, 12, 13, 20, 21, 22
10. Con riferimento al database prenotazioni scrivi la query che trova il numero di prenotazioni relative alle strutture 4 stelle
11. Con riferimento al database prenotazioni scrivi la query che trova l’importo medio delle prenotazioni con caparra minore di 50 relative alle strutture 2 stelle
12. Con riferimento al database prenotazioni scrivi la query che trova il numero, l’importo minimo e l’importo massimo delle caparre per le alle strutture 3 stelle
13. Con riferimento al database prenotazioni scrivi la query che per ogni tipologia di struttura trova l’importo totale e il numero di prenotazioni ricevute
14. Con riferimento al database magazzini scrivi la query che per ogni categoria di articolo trova il numero di articoli e il costo unitario medio
15. Con riferimento al database magazzini scrivi la query che per ogni fornitore e categoria di articolo trova il numero di articoli e il costo unitario totale
16. Con riferimento al database prenotazioni scrivi la query che trova i tipi di struttura che hanno importo medio della caparra maggiore di 50
17. Con riferimento al database prenotazioni scrivi la query che trova i nomi delle città che hanno generato più di 10 prenotazioni per le strutture 4 stelle
18. Con riferimento al database magazzini scrivi la query che trova i nomi dei fornitori che hanno più di 10 prodotti con costo unitario maggiore di 25
19. Con riferimento al database prenotazioni scrivi la query che trova le città che non hanno generato prenotazioni
20. Con riferimento al database prenotazioni scrivi la query che trova il nome delle regioni che hanno generato più di 10 prenotazioni di importo superiore a 200
21. Con riferimento al database IFTS scrivi la query che cancella tutte le prenotazioni che hanno caparra minore di 20
22. Con riferimento al database IFTS scrivi la query che cancella tutte le prenotazioni che hanno saldo (importo – caparra) uguale a 0
23. Con riferimento al database IFTS scrivi la query che imposta a 200 l’importo di tutte le prenotazioni con caparra superiore a 50

# Soluzioni

## Esercizio 1

SELECT regione

FROM regioni

WHERE area\_geografica = 'sud'

## Esercizio 2

SELECT \*

FROM prenotazioni

WHERE importo > 200

## Esercizio 3

SELECT \*

FROM prenotazioni

WHERE importo > 150

AND caparra < 50

## Esercizio 4

SELECT nome, cognome

FROM clienti

ORDER BY dataNascita ASC

## Esercizio 5

SELECT arrivo

FROM prenotazioni

WHERE tipo\_struttura='2 Stelle'

ORDER BY arrivo DESC

## Esercizio 6

SELECT id\_cliente, nome, cognome

FROM clienti

WHERE cognome = ‘Rossi’

OR cognome = ‘Bianchi’

ORDER BY cognome ASC, dataNascita DESC

## Esercizio 7

SELECT DISTINCT regioni.regione

FROM regioni

INNER JOIN citta ON regioni.id\_regione = citta.regione

INNER JOIN clienti ON citta.id\_citta = clienti.citta

INNER JOIN prenotazioni ON clienti.id\_cliente = prenotazioni.cliente

WHERE importo > 150 AND caparra < 50

## Esercizio 8

SELECT DISTINCT citta.citta, clienti.nome, clienti.cognome

FROM regioni

INNER JOIN citta ON regioni.id\_regione = citta.regione

INNER JOIN clienti ON citta.id\_citta = clienti.citta

INNER JOIN prenotazioni ON clienti.id\_cliente = prenotazioni.cliente

WHERE caparra > 50

AND caparra < 90

AND importo > 140 AND importo < 240

## Esercizio 9

SELECT DISTINCT citta.citta

FROM citta

INNER JOIN clienti ON citta.ID\_citta = clienti.citta

INNER JOIN prenotazioni ON clienti.ID\_cliente = prenotazioni.cliente

## Esercizio 10

SELECT COUNT(\*) AS numero\_prenotazioni

FROM prenotazioni

WHERE tipo\_struttura = ‘4 stelle’

## Esercizio 11

SELECT AVG(importo) AS importo\_medio

FROM prenotazioni

WHERE tipo\_struttura = ‘2 stelle’

AND caparra < 50

## Esercizio 12

SELECT COUNT(\*) AS numero\_prenotazioni, MIN(caparra) AS caparra\_minima, MAX(caparra) AS caparra\_massima

FROM prenotazioni

WHERE tipo\_struttura = ‘3 stelle’

## Esercizio 13

SELECT tipo\_struttura, SUM(importo) AS totale, COUNT(\*) AS numero\_prenotazioni

FROM prenotazioni

GROUP BY tipo\_struttura

## Esercizio 14

SELECT categorie.nome, COUNT(\*) AS quantita, AVG(articoli.costoUnitario) AS costo\_medio

FROM categorie

INNER JOIN articoli ON categorie.ID = articoli.categoria

GROUP BY categorie.nome

## Esercizio 15

SELECT fornitorni.nome, categorie.nome, COUNT(\*) AS quantita, SUM(articoli.costoUnitario) AS costo\_unitario\_totale

FROM categorie

INNER JOIN articoli ON categorie.ID = articoli.categoria

INNER JOIN fornitori ON fornitori.ID = articoli.fornitore

GROUP BY fornitorni.nome, categorie.nome

## Esercizio 16

SELECT tipo\_struttura

FROM prenotazioni

GROUP BY tipo\_struttura

HAVING AVG(caparra) > 50

## Esercizio 17

SELECT citta.citta

FROM citta

INNER JOIN clienti ON id\_citta = clienti.citta

INNER JOIN prenotazioni ON id\_cliente = prenotazioni.cliente

WHERE tipo\_struttura = ‘4 stelle’

GROUP BY citta.citta

HAVING COUNT(\*) > 10

## Esercizio 18

SELECT fornitori.nome

FROM fornitori

INNER JOIN prodotti ON fornitori.id = prodotti.fornitore

WHERE costoUnitario > 25

GROUP BY fornitori.nome

HAVING COUNT(\*) > 10

## Esercizio 19

SELECT citta.citta

FROM citta

LEFT OUTER JOIN clienti ON citta.id\_citta = clienti.citta

LEFT OUTER JOIN prenotazioni ON clienti.id\_cliente = prenotazioni.cliente

WHERE id\_prenotazione IS NULL

## Esercizio 20

SELECT regioni.regione

FROM prenotazioni

INNER JOIN clienti ON clienti.id\_cliente = prenotazioni.cliente

INNER JOIN citta ON citta.id\_citta = clienti.citta

INNER JOIN regioni ON regioni.id\_regione = citta.regione

WHERE importo > 200

GROUP BY regioni.regione

HAVING COUNT(\*) > 10

--Soluzione con variabili

DECLARE @importoMassimo AS real

DECLARE @limitePrenotazioni AS int

SET @importoMassimo = 200

SET @limitePrenotazioni = 10

SELECT regioni.regione

FROM prenotazioni

INNER JOIN clienti ON clienti.id\_cliente = prenotazioni.cliente

INNER JOIN citta ON citta.id\_citta = clienti.citta

INNER JOIN regioni ON regioni.id\_regione = citta.regione

WHERE importo > @importoMassimo

GROUP BY regioni.regione

HAVING COUNT(\*) > @ limitePrenotazioni

--Soluzione con sottoquery

SELECT regione

FROM(

SELECT regioni.regione, COUNT(\*) AS numero\_prenotazioni

FROM prenotazioni

INNER JOIN clienti ON clienti.id\_cliente = prenotazioni.cliente

INNER JOIN citta ON citta.id\_citta = clienti.citta

INNER JOIN regioni ON regioni.id\_regione = citta.regione

WHERE importo > 200

GROUP BY regioni.regione

) AS sottoquery

WHERE numero\_prenotazioni > 10

## Esercizio 21

DELETE

FROM prenotazioni

WHERE caparra < 20

## Esercizio 22

DELETE

FROM prenotazioni

WHERE importo - caparra = 20

## Esercizio 23

UPDATE prenotazioni

SET importo = 200

WHERE caparra > 50