# Esercizi

1. Con riferimento al database IFTS scrivi la query che trova tutti i clienti che hanno nome Mario
2. Con riferimento al database IFTS scrivi la query che trova tutte le regioni che appartengono all’area geografica Sud
3. Con riferimento al database IFTS scrivi la query che trova tutte le prenotazioni che hanno importo superiore a 200
4. Con riferimento al database IFTS scrivi la query che trova le prenotazioni con importo superiore a 150 e caparra inferiore a 50
5. ~~Con riferimento al database IFTS scrivi la query che trova la regione di provenienza dei clienti che hanno prenotazioni con importo superiore a 150 e caparra inferiore a 50~~
6. Con riferimento al database IFTS scrivi la query che trova tutti i nomi e cognomi dei clienti ordinati per età
7. Con riferimento al database IFTS scrivi la query che trova le date di arrivo per gli hotel 2 stelle ordinate in ordine cronologico dalla più recente alla più vecchia
8. Con riferimento al database IFTS scrivi la query che trova i nomi e i cognomi dei clienti che hanno prenotazioni con arrivo precedente al primo luglio 2015
9. Con riferimento al database IFTS scrivi la query che trova città di residenza, nome e cognome dei clienti che hanno prenotazioni con caparra inferiore a 50 e importo superiore a 150
10. Con riferimento al database IFTS scrivi la query che trova il nome della città di appartenenza dei clienti che hanno prenotato le camere 11, 12, 13, 20, 21, 22
11. Con riferimento al database IFTS scrivi la query che trova il nome della città di appartenenza dei clienti che hanno prenotato le camere 10, 20, 30, hanno un importo inferiore a 300 e una caparra superiore a 100
12. riferimento al database IFTS scrivi la query che per ogni tipologia di struttura trova l’importo totale e il numero di prenotazioni ricevute
13. Con riferimento al database IFTS scrivi la query che trova il nome delle regioni che hanno generato più di 10 prenotazioni di importo superiore a 200
14. Con riferimento al database IFTS scrivi la query che trova il nome della regione che ha generato il maggior numero di prenotazioni
15. Con riferimento al database IFTS scrivi la query che cancella tutti i clienti che hanno cognome Rossi
16. Con riferimento al database IFTS scrivi la query che cancella tutte le prenotazioni con caparra < 20
17. Con riferimento al database IFTS scrivi la query che cancella tutte le prenotazioni con saldo (importo – caparra) uguale a 0
18. Con riferimento al database IFTS scrivi la query che imposta a 200 l’importo di tutte le prenotazioni con caparra superiore a 50
19. Con riferimento al database IFTS scrivi la query che imposta a 20 la caparra di tutte le prenotazioni dei clienti che risiedono nella provincia di Roma
20. Con riferimento al database IFTS scrivi la query che imposta il valore della caparra uguale alla metà dell’importo di tutte le prenotazioni dei clienti che risiedono nell’area Centro
21. Con riferimento al database IFTS scrivi la query che inserisce una città chiamata Spadarolo in provincia di Rimini (sigla RN) nella regione 5
22. Con riferimento al database IFTS scrivi la query che inserisce un cliente chiamato Mario Rossi nella città 45
23. Con riferimento al database IFTS scrivi la query che trova le regioni che non hanno città
24. Con riferimento al database IFTS scrivi la query che trova le città che non hanno generato prenotazioni

# Soluzioni

## Esercizio 1

SELECT \*  
FROM clienti

WHERE nome = 'Mario'

## Esercizio 2

SELECT regione

FROM regioni

WHERE area\_geografica = 'sud'

## Esercizio 3

SELECT \*

FROM prenotazioni

WHERE importo > 200

## Esercizio 4

SELECT \*

FROM prenotazioni

WHERE importo > 150

AND caparra < 50

## Esercizio 5

--errore nel testo, l'esercizio non poteva essere risolto quando l'ho assegnato--

SELECT DISTINCT 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 6

SELECT DISTINCT nome, cognome

FROM clienti

ORDER BY dataNascita ASC

## Esercizio 7

SELECT DISTINCT arrivo

FROM prenotazioni

WHERE tipo\_struttura='2 Stelle'

ORDER BY arrivo DESC

## Esercizio 8

SELECT DISTINCT clienti.nome, clienti.cognome

FROM clienti

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

WHERE prenotazioni.arrivo < '2015-07-01'

## Esercizio 9

SELECT DISTINCT citta.citta, nome, cognome

FROM clienti

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

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

WHERE prenotazioni.caparra < 50

AND prenotazioni.importo > 150

## Esercizio 10

SELECT DISTINCT citta.citta

FROM citta

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

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

WHERE prenotazioni.camera IN ('11','12','13','20','21','22')

## Esercizio 11

SELECT DISTINCT citta.citta

FROM citta

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

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

WHERE prenotazioni.camera IN ('10','20','30')

AND prenotazioni.importo < 300

AND prenotazioni.caparra > 100

## Esercizio 12

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

FROM prenotazioni

GROUP BY tipo\_struttura

## Esercizio 13

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

## Esercizio 14

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

GROUP BY regioni.regione

ORDER BY COUNT(\*) DESC

LIMIT 0, 1

## Esercizio 15

DELETE

FROM clienti

WHERE cognome = 'Rossi'

## Esercizio 16

DELETE

FROM prenotazioni

WHERE caparra < 20

## Esercizio 17

DELETE

FROM prenotazioni

WHERE importo - caparra < 20

## Esercizio 18

UPDATE prenotazioni

SET importo = 200

WHERE caparra > 50

## Esercizio 19

UPDATE prenotazioni

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

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

SET caparra = 20

WHERE citta.nome\_provincia = 'Roma'

## Esercizio 20

UPDATE 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

SET caparra = importo / 2

WHERE regioni.area\_geografica = 'Centro'

## Esercizio 21

INSERT INTO citta (citta, nome\_provincia, provincia\_sigla, regione)

VALUES ('Spadarolo', 'Rimini', 'RN', 5)

## Esercizio 22

INSERT INTO clienti (nome, cognome, citta)

VALUES ('Mario', 'Rossi', 45)