- 1. CREATE TABLE scores (studentName VARCHAR(30), score INT, maxScore INT);
- 2. INSERT INTO scores (studentName, score, maxScore) VALUES ('Kyle', 80, 200);



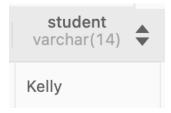
3. SELECT studentName, score FROM scores WHERE maxScore = 200;



- 4. CREATE TABLE appts (student VARCHAR(14), advisor CHAR(3), room INT);
- 5. INSERT INTO appts (student, advisor, room) VALUES ('Kelly', 'JSR', 5);



6. SELECT student FROM appts WHERE advisor = 'JSR';



7.

a. SELECT \* FROM FRIENDS WHERE LASTNAME LIKE 'M%';

LASTNAME varchar(30) →	FIRSTNAME varchar(30)	AREACODE int	PHONE varchar(255)	STATE char(2)	ZIP int \$
MEZA	AL	200	555-2222	UK	12345
MERRICK	BUD	300	555-6666	СО	80212
MAST	JD	381	555-6767	LA	23456

b. SELECT \* FROM FRIENDS WHERE STATE = 'IL' AND FIRSTNAME = 'AL';

LASTNAME varchar(30)	FIRSTNAME varchar(30)	AREACODE int	PHONE varchar(255)	STATE char(2)	ZIP int
BUNDY	AL	100	555-1111	IL	22333

- SELECT PARTNO FROM PART1 INTERSECT SELECT PARTNO FROM PART2;
- d. SELECT DISTINCT P1.PARTNO FROM PART1 P1 INNER JOIN PART2 P2 ON P1.PARTNO = P2 PARTNO;
- e. WHERE a BETWEEN 10 AND 30;
- f. SELECT FIRSTNAME FROM FRIENDS WHERE FIRSTNAME = 'AL' AND LASTNAME = 'BULHER';
  - i. It returns an empty set as there is no member in the friends table who's first name is AL and the Last Name is Bulcher.
- g. SELECT LASTNAME AS NAME, STATE AS ST FROM FRIENDS WHERE FIRSTNAME = 'AL' AND STATE = 'IL';



## Umair Dada - 20281

h. SELECT LASTNAME | | ', ' | | FIRSTNAME AS NAME, AREACODE | | '-' | | PHONE AS PHONE FROM FRIENDS WHERE LASTNAME IN ('MERRICK', 'MAST', 'BULHER') ORDER BY LASTNAME DESC;

- 8. ANSI
- 9. Projection
- 10. A Distinct / B Unique

11.

- a. SelectT \* FROM employees; (two T's in the select word)
- b. SELECT \* FROM "employees" (unless employees created with double quotes)
- c. SELECT \* FROM employees;
- 12. Right

13.

- a. Select (100\*2+50)/5 from dual OR
- d. Select ((100 \*2)+50)/5 from dual;
- 14. .
  - c. compensation: "null" commission: "Commission is "
- 15. .
  - d. Be enclosed with double goutes

16.

- a. SELECT
- b. FROM
- 17. True
- 18. Single Qoutes
- 19. 201
- 20. OR
- 21. .
- a. "\_"

```
22. .
   b. NOT
23. End of result set.
24. DESC (ORDER BY 'COLUMN1' DESC)
25. .
       a. column name or expression in the select list.
       b. a column alias in the select list.
       c. a number representing the column position in the select list.
       d. a column found in the data source but not in the select list.
26. A value for each row in the result set
27. UPPER
28.
   b. c
29. .
   c. A
30. .
   a. Select substr(46.567, 1, 2) from dual; (works because of implicit conversion-not if in
   doubles quotes)
   d. Select trunc(46.567) from dual;
31. .
   c. sysdate + 1/24
32. .
   b. TO CHAR(TO DATE('25-MAY-04','DD-MON-YY'), 'Month Ddspth, Year')
33. salary * 12
34. RR
35. Selection
36. .
   c. *
```

```
37. .
   d. multiplication, division, addition, subtraction
38. .
   a. SELECT 'Mr./Ms. '||first_name||' '||last_name ||' '||'is an employee of our
   company.' AS "Employees" FROM employees;
39. .
   c. Be enclosed with double quotes
40. .
   c. The result is null
41. .
   d. SELECT department id, last name, first name FROM employees;
42. .
   c. AND
43. .
   b. The results are sorted first numerically then alphabetically.
44. .
   c. Results for the code number column will be displayed from smallest number to
   largest number.
45. .
   b. ORDER BY last name, first name, department id, salary DESC
46. .
   a. job id = 'AD PRES' AND hire date LIKE '03-JUN-04';
47. .
   b. % and
48. .
   c. Any employee with an employee id of 103 and also any employee whose salary is
   4000.
49. .
   c. 410
```

## Umair Dada - 20281

- 50. .
  - d. SELECT last\_name FROM employees WHERE last\_name LIKE 'S%ae%';
- 51. .
  - d. Putting a column number in the SELECT list.
- 52. .
  - c. Four columns will be displayed with hire\_dates displayed with the most recent dates listed first.