1. CREATE TABLE user1(id INT PRIMARY KEY, fnam VARCHAR(50) NOT NULL, lnam VARCHAR(50) NOT NULL, address VARCHAR(50), fee INT NOT NULL, DOB DATE NOT NULL);
2. SELECT fnam || lnam AS fullname FROM user1;
3. SELECT fnam, name FROM user1, member WHERE user1.lnam='sher' AND member.id=1;
4. SELECT * FROM actor ORDER BY first_name LIMIT 10 OFFSET 10;
   SELECT * FROM ACTOR LIMIT 10 OFFSET 5;
5. SELECT DISTINCT district FROM address;
6. SELECT * FROM payment WHERE amount BETWEEN 2.99 AND 7.99;
7. SELECT COUNT(*) AS total FROM payment WHERE amount > 5.99;
8. SELECT customer_id, SUM (amount)FROM payment GROUP BY customer_id;
9. SELECT COUNT(staffNo) AS myCount, SUM(salary) AS mySum FROM Staff WHERE position = 'Manager';
10. SELECT staffNo, fName, lName, salary FROM Staff ORDER BY salary DESC;
11. SELECT MIN(salary) AS Minimum, MAX(salary) AS Maximum, AVG(salary) AS Aveage FROM Staff;
12. SELECT branchNo, COUNT(staffNo) AS TotalStaff, SUM(salary) AS Sum FROM Staff GROUP BY branchNo HAVING COUNT(staffNo) > 1 ORDER BY branchNo;
13. SELECT DISTINCT CITY FROM STATION WHERE ID%2 =0;
14. SELECT (COUNT(CITY) - COUNT(DISTINCT CITY)) AS DIFF FROM STATION;
15. select CITY,LENGTH(CITY) from STATION order by Length(CITY) asc, CITY limit 1;
16. select CITY,LENGTH(CITY) from STATION order by Length(CITY) desc, CITY limit 1;
17. SELECT staffNo, fName, lName, position, salary FROM Staff WHERE (SELECT AVG(salary) FROM Staff) < salary;
18. SELECT staffNo, fName, lName, position FROM Staff WHERE position IN ('Manager', 'Supervisor');
19. SELECT first_name, last_name FROM customer WHERE EXISTS (SELECT 1 FROM payment WHERE payment.customer_id = customer.customer_id);
20. ALTER TABLE user1 RENAME TO staff;
21. alter table staff rename column fee to salary;
22. UPDATE staff SET position='manager' WHERE id=1;

23. update member SET name=staff.fnam FROM staff;

24. ALTER TABLE staff ADD CONSTRAINT staff_salary_check CHECK (salary >= 15000);

25. CREATE TABLE item (item_id INT GENERATED ALWAYS AS IDENTITY, item_name VARCHAR NOT NULL);

26. SELECT staffno, fnam, lnam FROM staff WHERE address IS NULL;

27. SELECT fnam, lnam, email FROM staff INNER JOIN member ON staff.staffno = member.id;

28. SELECT film.film_id, title, inventory_id FROM film LEFT JOIN inventory ON inventory.film_id = film.film_id ORDER BY title;

29. DELETE FROM member WHERE id=2;

30. DELETE FROM member USING staff WHERE member.id = staff.staffno;

31. ALTER TABLE staff DROP COLUMN dob;

32. SELECT staffno, fnam, lnam, position FROM Staff WHERE position IN (‘Manager’, ‘Supervisor’);

33. SELECT fnam, lnam, salary FROM staff WHERE address LIKE ‘%SIALKOT%’;

34. SELECT staffno, fnam, lnam, salary FROM Staff ORDER BY salary DESC;

35. SELECT staffno, COUNT(salary) FROM Staff;

36. SELECT COUNT(staffNo) AS totalstaff, SUM(salary) AS totalsalary FROM Staff WHERE position = ‘Manager’;

37. SELECT staffno, fnam, lnam, position, salary – 17000 AS newsalary FROM staff WHERE salary > 17000;

38. SELECT staffno, fnam, lnam, position, salary FROM staff WHERE (SELECT AVG(salary) FROM Staff) < salary;

39. SELECT name FROM Employee WHERE salary >2000 AND months < 10 ORDER BY employee_id ASC;

40. select COUNT(NAME) FROM CITY WHERE POPULATION > 100000;

41. SELECT SUM(POPULATION) FROM CITY WHERE

DISTRICT='CALIFORNIA';

42. SELECT AVG(POPULATION) FROM CITY WHERE DISTRICT='CALIFORNIA';

43. SELECT ROUND(AVG(POPULATION)) FROM CITY;

44. SELECT DISTINCT CITY FROM STATION WHERE CITY REGEXP '[aeiou]$';

45. SELECT SUM(POPULATION) FROM CITY WHERE COUNTRYCODE='JPN';

46. SELECT MAX(POPULATION)-MIN(POPULATION) FROM CITY;

47. SELECT DISTINCT CITY FROM STATION WHERE CITY REGEXP '[^aeiou]$';

48. SELECT DISTINCT CITY FROM STATION WHERE CITY REGEXP '^[^aeiou]|[^aeiou]$';

49. SELECT ROUND(MAX(LAT_N), 4) FROM STATION WHERE LAT_N < 137.2345;

50. CREATE DATABASE newdb WITH ENCODING='UTF8'OWNER=sana1 CONNECTION LIMIT=25;

51. ALTER DATABASE newdb OWNER TO postgres;

52. ALTER DATABASE newdb RENAME TO mydb;

53. DROP DATABASE mydb;

54. TRUNCATE TABLE member, user;

55. SELECT ROUND(LONG_W, 4) FROM STATION WHERE LAT_N < 137.2345 ORDER BY LAT_N DESC LIMIT 1;

56. SELECT ROUND(SUM(LAT_N), 2), ROUND(SUM(LONG_W), 2) FROM STATION;

57. CREATE TABLE user (id INT NOT NULL , s_id INT REFERENCES staff(staffno), name VARCHAR(50) NOT NULL PRIMARY KEY(id,s_id));

58. CREATE TABLE user1(id INT PRIMARY KEY, fnam VARCHAR(50) NOT NULL, lnam VARCHAR(50) NOT NULL, address VARCHAR(50), fee INT NOT NULL, DOB DATE NOT NULL, UNIQUE(address));

59. CREATE SCHEMA school;

60. CREATE TABLE school.user1(id INT PRIMARY KEY, fnam VARCHAR(50) NOT NULL, lnam VARCHAR(50) NOT NULL, address VARCHAR(50), fee INT NOT NULL, DOB DATE NOT NULL, UNIQUE(address));

61. SELECT * FROM schhol.user;

62. ALTER SCHEMA school RENAME TO office;

63. ALTER SCHEMA office OWNER TO sana1;

64. CREATE TEMPORARY TABLE user1(id INT PRIMARY KEY, fnam VARCHAR(50) NOT NULL, lnam VARCHAR(50) NOT NULL, address VARCHAR(50), fee INT NOT NULL, DOB DATE NOT NULL, UNIQUE(address));

65. SELECT SUM (CASE rental_rate WHEN 0.99 THEN 1 ELSE 0 END) AS "Mass", SUM (CASE rental_rate WHEN 2.99 THEN 1 ELSE 0 END) AS "Economic", SUM (CASE rental_rate WHEN 4.99 THEN 1 ELSE 0 END) AS "Luxury"FROM film;

66. SELECT staffno, fnam, lnam, position, salary FROM Staff WHERE salary > ALL (SELECT salary FROM Staff WHERE branchNo = 'B003');

67. SELECT Email FROM Person GROUP BY Email HAVING COUNT(*) > 1;

68. SELECT Name AS Customers FROM Customers WHERE Id NOT IN (SELECT CustomerId FROM Orders);

69. SELECT fnam, lnam FROM staff ORDER BY position FETCH FIRST 5 ROW ONLY;

70. INSERT INTO price VALUES (1, 1000, 20), (2, 15000, 200), (3, 20000, 500);

71. SELECT pid, (price – COALESCE(discount,0)) AS netprice FROM price;

72. SELECT pid, (price - CASE WHEN discount IS NULL THEN 0 ELSE discount END) AS netprice FROM price;

73. SELECT LEFT(address, 8) AS shortaddress FROM staff;

74. SELECT DISTINCT CITY FROM STATION WHERE CITY REGEXP '[^aeiou]$';

75. SELECT fnam, concat ('Your first name has',LENGTH(fnam), ' characters')FROM staff;

76. SELECT POSITION('is' IN 'This is a cat');

77. SELECT split_part(payment_date::TEXT,'-', 1) y, split_part (payment_date::TEXT,'-', 2) m, amount FROM payment;

78. UPDATE staff SET lnam = REPLACE (lnam,'sher','ch');

79. SELECT fnam,SUBSTRING(lnam, 1, 1) AS initial FROM staff

ORDER BY lnam;

80. SELECT TRANSLATE('apple,orange,banana', ',', ';');

81. SELECT REVERSE(fnam) FROM staff;

82. SELECT TRIM(" psql practice ") AS TrimmedString;

83. SELECT payment_id, payment_date, TO_CHAR(payment_date,'MON-DD-YYYY HH12:MIAM') payment_time FROM payment;

84. SELECT UCASE(fnam) AS CapitalLetter FROM staff;

85. SELECT LCASE(fnam) AS SmallLetter FROM staff;

86. SELECT LPAD('PostgreSQL',15,'*');

87. SELECT MID(fnam, 2, 4) AS ExtractString FROM staff;

88. SELECT CEIL(SUM(amount)) amt FROM payment;

90. SELECT DISTINCT CITY FROM STATION WHERE mod(ID,2)=0;

91. SELECT Name FROM STUDENTS WHERE Marks > 75 ORDER BY RIGHT(Name, 3),ID;

92. SELECT FORMAT('%s, %s',lnam ,fnam) full_name FROM staff ORDER BY full_name;

93. SELECT name FROM Employee ORDER BY name;

94. SELECT CEIL(AVG(Salary) - AVG(REPLACE(Salary, '0', ''))) FROM EMPLOYEES;

95. SELECT (months*salary) as earnings, COUNT(*) FROM Employee GROUP BY earnings ORDER BY earnings DESC LIMIT 1;

96. SELECT SUM(i.POPULATION) FROM CITY AS i JOIN COUNTRY AS o ON i.COUNTRYCODE=o.CODE WHERE o.CONTINENT='Asia';

97. SELECT fnam, CONCAT(lnam, ' is ', position) AS text FROM staff;

98. SELECT con.CONTINENT, FLOOR(AVG(c.POPULATION)) FROM CITY AS c JOIN COUNTRY AS con ON c.COUNTRYCODE=con.CODE GROUP BY con.CONTINENT;

99. SELECT ROUND(SUM(LAT_N), 4) FROM STATION WHERE LAT_N > 38.7880 AND LAT_N < 137.2345;

100. SELECT CONCAT(Name, '(', LEFT(Occupation,1),')') FROM OCCUPATIONS ORDER BY Name;

SELECT CONCAT('There are a total of ', COUNT(Occupation), ' ', LOWER(Occupation), 's.') FROM OCCUPATIONS GROUP BY Occupation ORDER BY COUNT(Occupation), Occupation;