1. SELECT model, speed, hd FROM PC WHERE price < 500

2. SELECT DISTINCT maker FROM Product WHERE type = 'Printer'

3. SELECT Product.model, Laptop.ram, Laptop.screen

FROM Product INNER JOIN Laptop

ON Product.model = Laptop.model WHERE Laptop.price > 1000

4. SELECT \* FROM Printer WHERE color = 'y'

5. SELECT model, speed, hd

FROM PC

WHERE ((cd = '12x' OR cd = '24x') AND price < 600)

6. SELECT DISTINCT Product.maker, Laptop.speed

FROM Product INNER JOIN Laptop

ON Product.model = Laptop.model WHERE Laptop.hd >= 10

7. SELECT Laptop.model, Laptop.price

FROM Product INNER JOIN Laptop

ON Product.model = Laptop.model WHERE Product.maker = 'B'

UNION

SELECT Printer.model, Printer.price

FROM Product INNER JOIN Printer

ON Product.model = Printer.model WHERE Product.maker = 'B'

UNION

SELECT PC.model, PC.price

FROM Product INNER JOIN PC

ON Product.model = PC.model WHERE Product.maker = 'B'

8. SELECT DISTINCT maker

FROM product WHERE type='pc' and maker NOT IN

(SELECT maker FROM product WHERE type='laptop')

9. SELECT DISTINCT Product.maker

FROM Product INNER JOIN PC

ON Product.model = PC.model WHERE PC.speed >= 450

10. SELECT DISTINCT model, price

FROM Printer

WHERE price = (SELECT MAX(price)

FROM Printer

)

11. SELECT AVG(speed)

FROM PC

12. SELECT AVG(speed)

FROM Laptop WHERE price > 1000

13. SELECT AVG(speed)

FROM Product INNER JOIN PC

ON Product.model = PC.model WHERE Product.maker = 'A'

14. SELECT Classes.class, Ships.name, Classes.country

FROM Classes INNER JOIN Ships

ON Classes.class= Ships.class WHERE Classes.numGuns >= 10

15. SELECT hd FROM PC

GROUP BY (hd)

HAVING COUNT(model) >= 2

16. SELECT DISTINCT p1.model, p2.model, p1.speed, p1.ram

FROM PC as p1, PC as p2

WHERE p1.speed = p2.speed and p1.ram = p2.ram and p1.model > p2.model

17. SELECT DISTINCT Product.type, Laptop.model, Laptop.speed

FROM Product, Laptop, PC

WHERE Laptop.speed < (SELECT MIN (speed) FROM PC) AND

Product.type = 'laptop'

18. SELECT DISTINCT Product.maker, Printer.price

FROM Product INNER JOIN Printer

ON product.model = printer.model

WHERE (Printer.price = (SELECT MIN (price) FROM Printer WHERE Printer.color = 'y')) AND Printer.color = 'y'

19. SELECT Product.maker, AVG(screen) AS Avg\_screen

FROM Product, Laptop

WHERE Product.model = Laptop.model

GROUP BY Product.maker

20. SELECT maker, COUNT(model)

FROM Product

WHERE type = 'pc'

GROUP BY (maker)

HAVING COUNT(DISTINCT model) >= 3

21. SELECT Product.maker, MAX(PC.price)

FROM Product INNER JOIN PC

ON product.model = pc.model

GROUP BY (maker)

22. SELECT speed, AVG(price)

FROM PC

WHERE speed > 600

GROUP BY (speed)

23. SELECT maker

FROM Product, PC

WHERE (Product.model = PC.model AND PC.speed>=750)

INTERSECT

SELECT maker

FROM Product, Laptop

WHERE (Product.model = Laptop.model AND Laptop.speed>=750)

24. WITH allpr AS

(SELECT model, price

FROM pc

WHERE price =

(SELECT MAX(price)

FROM pc)

UNION

SELECT model, price

FROM laptop

WHERE price =

(SELECT MAX(price)

FROM laptop)

UNION

SELECT model, price

FROM printer

WHERE price =

(SELECT MAX(price)

FROM printer))

SELECT model

FROM allpr

WHERE price =

(SELECT MAX(price)

FROM allpr )

25. SELECT DISTINCT Product.maker

FROM product INNER JOIN PC

ON Product.model = pc.model

WHERE PC.ram = (select min(ram) from PC)

and PC.speed = (SELECT MAX(speed)

FROM PC WHERE ram = (SELECT MIN(ram) FROM PC))

and Product.maker in

(SELECT maker

FROM product

WHERE type = 'printer')

26. WITH allpr AS

(SELECT PC.model, PC.price, Product.maker

FROM PC INNER JOIN Product

ON PC.model = Product.model

WHERE Product.maker = 'A'

UNION ALL

SELECT Laptop.model, Laptop.price, Product.maker

FROM Laptop INNER JOIN Product

ON Laptop.model = Product.model

WHERE Product.maker = 'A'

)

SELECT AVG(price)

FROM allpr

27. SELECT Product.maker, AVG(PC.hd)

FROM Product INNER JOIN PC

ON product.model = pc.model

WHERE Product.maker in

(SELECT maker

FROM product

WHERE type = 'printer')

GROUP BY (maker)

28. SELECT COUNT(maker) FROM Product

WHERE maker IN

(

SELECT maker FROM Product

GROUP BY maker

HAVING COUNT(model) = 1

)

29. select t.point, t.date, SUM(t.inc), sum(t.out)

from(select point, date, inc, null as out from Income\_o

Union

select point, date, null as inc, Outcome\_o.out from Outcome\_o)

as t group by t.point, t.date

30. with i as (

select point, date, sum(inc) as inc

from Income

group by point, date

),

o as (

select point, date, sum(out) as out

from Outcome

group by point, date

)

select i.point, i.date, o.out, i.inc

from i left join o on i.point = o.point and i.date = o.date

union

select o.point, o.date, o.out, i.inc

from i right join o on i.point = o.point and i.date = o.date

31. SELECT class, country

FROM Classes

WHERE bore >= 16

32. SELECT res.country, cast(avg(power(res.bore,3)/2) as numeric(6,2)) as weight

From (SELECT Classes.country, Classes.bore, Ships.name

FROM Classes JOIN Ships

ON Classes.class=Ships.class

Union

SELECT Classes.country, Classes.bore, Outcomes.ship AS name

FROM Classes JOIN Outcomes

ON Classes.class=Outcomes.ship) AS res

GROUP BY res.country

33. SELECT ship

FROM Outcomes

WHERE result = 'sunk' AND battle = 'North Atlantic'

34. SELECT Ships.name

FROM Classes INNER JOIN Ships

ON Classes.class = Ships.class

WHERE Classes.type = 'bb' AND Ships.launched >= 1922 AND Classes.displacement > 35000

35. SELECT model, type

FROM Product

WHERE UPPER(model) NOT LIKE '%[^A-Z]%'

OR model NOT LIKE '%[^0-9]%'

36. SELECT name

FROM Ships

Where class = name

UNION

SELECT Outcomes.ship

FROM Outcomes, Classes

Where Outcomes.ship = Classes.class

37. SELECT Classes.class

FROM Classes JOIN

(SELECT name, class FROM Ships

UNION

SELECT ship, ship FROM Outcomes) AS res

ON Classes.class = res.class

GROUP BY Classes.class

HAVING COUNT(res.name) = 1

38. SELECT DISTINCT country FROM Classes

WHERE country IN

(

SELECT country FROM Classes

WHERE type = 'bb'

INTERSECT

SELECT country FROM Classes

WHERE type = 'bc'

)

39. SELECT DISTINCT o2.ship

FROM Battles as b1 JOIN Outcomes o1 ON o1.battle = b1.name,

Battles as b2 JOIN Outcomes o2 ON o2.battle = b2.name

WHERE b1.date < b2.date

AND o1.result = 'damaged'

AND o1.ship = o2.ship

40. SELECT maker, MAX(TYPE)

FROM product

GROUP BY maker

HAVING COUNT (model)> 1 AND MAX(TYPE)=MIN(TYPE)

41. SELECT Product.maker,

CASE WHEN MAX(CASE WHEN x.price IS NULL THEN 1 ELSE 0 END) = 0

THEN MAX(x.price) ELSE NULL END

FROM Product JOIN (SELECT model, price FROM PC

UNION

SELECT model, price FROM Laptop

UNION

SELECT model, price FROM Printer) AS x

ON Product.model = x.model

GROUP BY Product.maker

42. SELECT ship, battle

FROM Outcomes

WHERE result = 'sunk'

43. SELECT name FROM Battles

WHERE DATEPART(year, date) NOT IN

(SELECT launched from Ships

WHERE launched IS NOT NULL)

44. SELECT name

FROM Ships

WHERE UPPER(name) LIKE 'R%'

UNION

SELECT ship

FROM Outcomes

WHERE UPPER(ship) LIKE 'R%'

45. SELECT name FROM Ships

WHERE name LIKE '% % %'

UNION

SELECT ship FROM Outcomes

WHERE ship LIKE '% % %'

46. SELECT Outcomes.ship, displacement, numGuns

FROM (

SELECT name AS ship, displacement, numGuns

FROM Ships JOIN Classes ON Classes.class=Ships.class

UNION

SELECT class AS ship, displacement, numGuns

FROM Classes

) AS a RIGHT JOIN Outcomes ON Outcomes.ship=a.ship

WHERE battle = 'Guadalcanal'

47. WITH t1 AS

(SELECT COUNT(name) as co, country

FROM

(SELECT name, country

FROM Classes INNER JOIN Ships

ON Ships.class = Classes.class

UNION

SELECT ship, country

FROM Classes INNER JOIN Outcomes

ON Outcomes.ship = Classes.class) x1

GROUP BY country

),

t2 AS (

SELECT COUNT(name) as co, country

FROM

( SELECT name, country

FROM Classes INNER JOIN Ships

ON Ships.class = Classes.class

WHERE name IN

(SELECT DISTINCT ship

FROM Outcomes

WHERE result LIKE 'sunk')

UNION

SELECT ship, country

FROM Classes INNER JOIN Outcomes

ON Outcomes.ship = Classes.class

WHERE ship IN

(SELECT DISTINCT ship

FROM Outcomes

WHERE result LIKE 'sunk')) x2

GROUP BY country )

SELECT t1.country

FROM t1 INNER JOIN t2

ON t1.co = t2.co

and t1.country = t2.country

48. Select Ships.class

From Ships INNER JOIN Outcomes

ON Outcomes.ship = Ships.name

WHERE Outcomes.result = 'sunk'

Union

Select Classes.class

From Classes INNER JOIN Outcomes

ON Outcomes.ship = Classes.class

WHERE Outcomes.result = 'sunk'

49. SELECT Ships.name

FROM Classes INNER JOIN Ships

ON Ships.class = Classes.class

WHERE bore = 16

UNION

SELECT Outcomes.ship

FROM Classes INNER JOIN Outcomes

ON Outcomes.ship = Classes.class

WHERE bore = 16

50. SELECT DISTINCT battle

FROM Outcomes

JOIN Ships ON Outcomes.ship=Ships.name

WHERE Ships.class='Kongo'

51. WITH sh AS (

SELECT name, class FROM ships

UNION

SELECT ship, ship FROM outcomes

)

SELECT

name

FROM sh JOIN Classes c on sh.class=c.class

WHERE numguns >= ALL(

SELECT ci.numguns FROM Classes ci

WHERE ci.displacement=c.displacement

AND ci.class IN (SELECT sh.class FROM sh)

)

52. WITH sh AS (

SELECT name, class FROM ships

UNION

SELECT ship, ship FROM outcomes

)

SELECT

name

FROM sh JOIN Classes c on sh.class=c.class

WHERE country = 'Japan'

AND type = 'bb'

AND (displacement <= 65000

OR displacement IS NULL)

AND (bore < 19

OR bore IS NULL)

AND (numGuns >= 9

OR numGuns IS NULL)

53. SELECT cast(avg(numGuns\*1.0) as numeric(6,2))

FROM Classes

WHERE type = 'bb'

54. WITH sh AS (

SELECT name, class FROM ships

UNION

SELECT ship, ship FROM outcomes

)

SELECT

cast(avg(numGuns\*1.0) as numeric(6,2))

FROM sh JOIN Classes c on sh.class=c.class

WHERE type = 'bb'

55. WITH sh AS (

SELECT name, class, launched FROM ships

)

SELECT

c.class, MIN(launched)

FROM sh FULL JOIN Classes c on sh.class=c.class

GROUP BY c.class

56. select class, SUM(CASE WHEN result='sunk' THEN 1 ELSE 0 END)

from (

select classes.class, name from classes

left join ships on classes.class=ships.class

union

select class, ship from classes

join outcomes on class=ship

) as sh

left join outcomes on sh.name=outcomes.ship

group by class

57. select class, SUM(CASE WHEN result='sunk' THEN 1 ELSE 0 END)

from (

select classes.class, name from classes

left join ships on classes.class=ships.class

union

select class, ship from classes

join outcomes on class=ship

) as sh

left join outcomes on sh.name=outcomes.ship

group by class

having

SUM(CASE WHEN result='sunk' THEN 1 ELSE 0 END) > 0

and (select count(si.name)

from (

select ships.name, ships.class from ships

union

select outcomes.ship, outcomes.ship from outcomes) as si

where si.class = sh.class

group by si.class) >= 3

58. select distinct

maker, type, cast((

count(model) over(partition by maker, type))\*100.0/

count(model) over(partition by maker)

as NUMERIC(5,2))

from (

select

pt.maker, pt.type, p.model

from (

select distinct a.maker, b.type

from product a, product b

) pt

left join product p on pt.maker=p.maker and pt.type=p.type

)as p

59.