

SHIPMENTS

Supplier number	Part number	Project number	Quantity
SN1	PN1	PJ1	300
SN1	PN1	PJ4	400
SN2	PN3	PJ1	350
SN2	PN3	PJ2	450
SN2	PN3	PJ3	640
SN2	PN3	PJ4	320
SN2	PN3	PJ5	330
SN2	PN3	PJ6	520
SN2	PN3	PJ7	480
SN2	PN5	PJ2	460
SN3	PN3	PJ1	440
SN3	PN4	PJ2	410
SN4	PN6	PJ3	310
SN4	PN6	PJ7	320
SN5	PN2	PJ2	340
SN5	PN2	PJ4	350
SN5	PN5	PJ5	360
SN5	PN5	PJ7	370
SN5	PN6	PJ2	380
SN5	PN1	PJ4	420
SN5	PN3	PJ4	440
SN5	PN4	PJ4	440
SN5	PN5	PJ4	400
SN5	PN6	PJ4	410

13. (i) For each part supplied, get the part number and the total shipment quantity?

- (A) SELECT shipments.part-number, SUM(shipments.quantity)
FROM Shipments
GROUP BY shipments.part-number
- (B) SELECT SUM(shipments.quantity)
FROM Shipments
GROUP BY shipments.part-number
- (C) SELECT shipments.part-number, SUM(shipments.quantity)
FROM Shipments
GROUP BY shipments.quantity
- (D) SELECT shipments.part-number, SUM(shipments.part-number)
FROM Shipments
GROUP BY shipments.part-number

- (ii) Get part numbers for parts supplied by more than two suppliers?

- (A) SELECT shipments.part-number
FROM Shipments
GROUP BY shipments.part-number
HAVING COUNT(shipments.supplier-number) > 2

- (B) SELECT shipments.part-number
FROM Shipments
GROUP BY shipments.part-number
HAVING COUNT(shipments.supplier-number) >= 2

- (C) SELECT shipments.part-number
FROM Shipments
GROUP BY shipments.part-number > 2

- (D) SELECT shipments.part-number, COUNT(shipments.supplier-number) > 2
FROM Shipments
GROUP BY shipments.part-number

- (iii) Get supplier names for suppliers who supply part PN3?

- (A) SELECT DISTINCT suppliers.supplier-name
FROM Supplier
WHERE suppliers.supplier-number IN (SELECT Shipments.supplier-number
FROM Shipments
WHERE Shipments.part-number='PN3')

- (B) SELECT DISTINCT suppliers.supplier-name
FROM Supplier
WHERE suppliers.supplier-number NOT IN (SELECT Shipments.supplier-number
FROM Shipments
WHERE Shipments.part-number='PN3')

- (C) SELECT DISTINCT suppliers.supplier-name
FROM Supplier
WHERE suppliers.supplier-number EXCEPT (SELECT Shipments.supplier-number
FROM Shipments
WHERE Shipments.part-number='PN3')

- (D) SELECT DISTINCT suppliers, supplier-name
FROM Supplier
WHERE suppliers.supplier-number
UNION
SELECT Shipments.supplier-number
FROM Shipments
WHERE Shipments.part-number='PN3'

14. (i) Get supplier names for suppliers who supply at least one blue part.

- (A) SELECT DISTINCT suppliers.supplier-name
FROM Suppliers
WHERE suppliers.supplier-number
IN (SELECT Shipments.supplier-number
FROM Shipments
WHERE Shipments.part-number
IN (SELECT Parts.part-number
FROM Parts
WHERE Parts.color='Blue'))

- (B) SELECT DISTINCT suppliers.supplier-name
FROM Suppliers
WHERE suppliers.supplier-number
IN (SELECT Shipments.supplier-number
FROM Shipments

- WHERE Shipments.part-number NOT
IN(SELECT Parts.part-number
FROM Parts
WHERE Parts.color='Blue'))
- (C) SELECT DISTINCT suppliers.supplier-name
FROM Suppliers
WHERE suppliers.supplier-number NOT
IN(SELECT Shipments.supplier-number
FROM Shipments
WHERE Shipments.part-number
IN (SELECT Parts.part-number
FROM Parts
WHERE Parts.color='Blue'))
- (D) SELECT DISTINCT suppliers.supplier-name
FROM Suppliers
WHERE suppliers.supplier-number
IN (SELECT Shipments.supplier-name
FROM Shipments
WHERE Shipments.part-number
IN (SELECT Parts.part-number
FROM Parts
WHERE Parts.color='Blue'))
- (ii) Get supplier numbers for suppliers with status less than the current maximum status in the suppliers table:
- (A) SELECT Suppliers.supplier-number
FROM suppliers
WHERE Suppliers.status < (SELECT MAX
(Suppliers.status)
FROM suppliers)
- (B) SELECT Suppliers.supplier-number
FROM suppliers
WHERE Suppliers.status <= (SELECT MAX
(Suppliers.status)
FROM Suppliers)
- (C) SELECT Suppliers.supplier-number,
MAX (Suppliers.status)
FROM suppliers
WHERE Suppliers.status
- (D) SELECT Suppliers.supplier-number
FROM suppliers
WHERE Suppliers.status=MAX(Suppliers.
status)
- (iii) Get supplier names for suppliers who supply part PN2?
- (A) SELECT DISTINCT suppliers.supplier-name
FROM Suppliers
WHERE EXIST(SELECT *
FROM Shipments
WHERE Shipments.supplier-number = sup-
pliers.supplier-number
AND
Shipments.part-number='PN2')
- (B) SELECT DISTINCT suppliers.supplier-name
FROM Suppliers
WHERE NOT EXIST(SELECT *
FROM Shipments
WHERE Shipments.supplier-number = sup-
pliers.supplier-number
AND
Shipments.part-number='PN2')
- (C) SELECT DISTINCT suppliers.supplier-name
FROM Suppliers
WHERE EXIST(SELECT *
FROM Shipments
WHERE Shipments.supplier-number = sup-
pliers.supplier-number
OR
Shipments.part-number='PN2')
- (D) SELECT DISTINCT suppliers.supplier-name
FROM Suppliers
WHERE EXIST(SELECT *
FROM Shipments
WHERE Shipments.supplier-number = sup-
pliers.supplier-number
UNION
Shipments.part-number='PN2')
15. (i) Get supplier names for suppliers who do not supply part PN2.
- (A) SELECT DISTINCT suppliers.supplier-name
FROM Suppliers
WHERE NOT EXIST(SELECT *
FROM Shipments
WHERE Shipments.supplier-number = sup-
pliers.supplier-number
AND
Shipments.part-number='PN2')
- (B) SELECT DISTINCT suppliers.supplier-name
FROM Suppliers
WHERE EXIST(SELECT *
FROM Shipments
WHERE Shipments.supplier-number = sup-
pliers.supplier-number
AND
Shipments.part-number='PN2')
- (C) SELECT DISTINCT suppliers.supplier-name
FROM Suppliers
WHERE EXCEPT(SELECT *
FROM Shipments
WHERE Shipments.supplier-number = sup-
pliers.supplier-number
AND
Shipments.part-number='PN2')
- (D) SELECT DISTINCT suppliers.supplier-name
FROM Suppliers
WHERE NOT EXIST(SELECT *
FROM Shipments

- (A) I – 2, II – 1, III – 4, IV – 3
 (B) I – 3, II – 4, III – 2, IV – 1
 (C) I – 1, II – 2, III – 3, IV – 4
 (D) I – 2, II – 3, III – 4, IV – 1

20. Which one is correct for division operations for relation r and s

- (A) $r \div s$
 (B) $\pi_{R-S}(r) - \pi_{R-S}((\pi_{R-S}(r) \times s) - \pi_{R-S}, s(r))$
 (C) Temp 1 $\leftarrow \pi_{R-S}(r)$
 Temp 2 $\leftarrow \pi_{R-S}(\text{temp1} \times s) - \pi_{R-S}, s(r)$
 result = temp 1 – temp 2
 (D) All the above

Practice Problems 2

Directions for questions 1 to 20: Select the correct alternative from the given choices.

- The correct order of SQL expression is
 (A) Select, group by, where, having
 (B) Select, where, group by, having
 (C) Select, group by, having, where
 (D) Select, having, where, group by
- Which one is not a query language?
 (A) SQL (B) QBE
 (C) Data log (D) MySQL
- Like 'a b \ % c d' escape '\ ' matches all the strings
 (A) Ending with a b c d
 (B) Beginning with a b c d
 (C) Beginning with a b % c d
 (D) Beginning with a b % c d
- '_ _ _ %' matches any string of
 (A) At least three characters
 (B) At most three characters
 (C) Exactly three characters
 (D) exactly three characters ending with %
- Which of the following are set operations?
 (i) Union
 (ii) Intersection
 (iii) Set Difference
 (iv) Cartesian Product
 (A) (i), (ii), (iii)
 (B) (i), (iii), (iv)
 (C) (i), (iii), (ii), (iv)
 (D) (i), (ii), (iv)
- What is the purpose of project operation?
 (A) It selects certain columns
 (B) It selects certain rows
 (C) It selects certain strings
 (D) It selects certain integers

Common data for questions 7 and 8: Person

Id	Name	Age	Hobby
11	Anu	21	Stamp Collection
22	Kamal	32	Painting
33	Ravi	24	Dancing
44	Ram	22	Singing

- Select the persons whose hobby is either painting (or) singing.
 (A) $\sigma_{\text{Hobby} = \text{'painting'} \text{ OR } \text{'singing'}}(\text{person})$
 (B) $\sigma_{\text{Hobby} = \text{'painting'}, \text{'singing'}}(\text{person})$
 (C) $\sigma_{\text{Hobby} = \text{'painting'} \text{ OR } \text{'singing'}}(\text{person})$
 (D) All are correct
- Select the persons whose age is above 21 and below 32:
 (A) $\sigma_{\text{age} > 21 \text{ AND } \text{age} < 32}(\text{person})$
 (B) $\sigma_{21 < \text{age} < 32}(\text{person})$
 (C) $\sigma_{\text{age} > 21 \text{ OR } \text{age} < 32}(\text{person})$
 (D) $\sigma_{\text{age} < 21 \text{ AND } \text{age} > 32}(\text{person})$

Common data for questions 9 and 10: Consider the following relation: Teach

Name	course	Rating	Age
Zohar	MD	7	35
Ilkha	BDS	8	27
Zohar	MS	7	34
Ravi	MBA	9	33

- Select the teachers whose rating is above 7 and whose age is less than 32?
 (A) $s_{\text{Rating} > 7 \text{ AND } \text{Age} < 32}(\text{Teach})$
 (B) $s_{\text{Rating} \geq 7 \text{ AND } \text{Age} < 32}(\text{Teach})$
 (C) $s_{\text{Rating} > 7 \text{ AND } \text{Age} < 32}(\text{Teach})$
 (D) Both (A) and (B)
- Select the courses with rating above 7?
 (A) $\pi_{\text{course}}(\sigma_{\text{rating} > 7}(\text{Teach}))$
 (B) $\sigma_{\text{course}}(\pi_{\text{rating} > 7}(\text{Teach}))$
 (C) $\pi_{\text{name, course}}(\sigma_{\text{rating} > 7}(\text{Teach}))$
 (D) None

Common data for questions 11 and 12: Consider the following schema of a relational database employee (empno, ename, eadd) project (pno, pname) Work-on (empno, pno) Part(partno, partname, qty-on-hand, size) Use (empno, pno, partno, number)

- Display the names of the employees who are working on a project named 'VB'.
 (A) $\sigma_{\text{name}}(\text{employee} \bowtie (\sigma_{\text{pname} = \text{'VB'}}(\text{project}) \bowtie \text{worked on}))$
 (B) $\sigma_{\text{name}}(\text{employee} \bowtie (\pi_{\text{pname} = \text{'VB'}}(\text{project}) \bowtie \text{work on}))$
 (C) $\pi_{\text{name}}(\text{employee} \bowtie (\sigma_{\text{pname} = \text{'VB'}}(\text{project}) \bowtie \text{work on}))$
 (D) $\pi_{\text{name}}(\text{employee} \bowtie (\pi_{\text{pname} = \text{'VB'}}(\text{project}) \bowtie \text{work on}))$

The following query is made on the database.

$$T1 \leftarrow \pi_{CourseName}(\sigma_{StudentName='SA'}(CR))$$

$$T2 \leftarrow CR \div T1$$

The number of rows in $T2$ is _____. [2017]

35. Consider the following database table named *top_scorer*:

top_scorer		
player	country	goals
Klose	Germany	16
Ronaldo	Brazil	15
G Müller	Germany	14
Fontaine	France	13
Pelé	Brazil	12
Klinsmann	Germany	11
Kocsis	Hungary	11
Batistuta	Argentina	10
Cubillas	Peru	10
Lato	Poland	10
Lineker	England	10
T Muller	Germany	10
Rahn	Germany	10

Consider the following SQL query:

```
SELECT ta.player FROM top_scorer AS ta
WHERE ta.goals > ALL (SELECT tb.goals
FROM top_scorer AS tb
WHERE tb.country = 'Spain')
AND ta.goals < ANY (SELECT tc.goals
FROM top_scorer AS tc
WHERE tc.country = 'Germany')
```

The number of tuples returned by the above SQL query is _____. [2017]

36. Consider the following two tables and four queries in SQL.

Book (isbn, bname), Stock (isbn, copies)

Query 1: SELECT B.isbn, S.copies
FROM Book B INNER JOIN Stock S
ON B.isbn = S.isbn;

Query 2: SELECT B.isbn, S.copies
FROM Book B LEFT OUTER
JOIN Stock S
ON B.isbn = S.isbn;

Query 3: SELECT B.isbn, S.copies
FROM Book B RIGHT OUTER
JOIN Stock S
ON B.isbn = S.isbn;

Query 4: SELECT B.isbn, S.copies
FROM Book B FULL OUTER
JOIN Stock S
ON B.isbn = S.isbn;

Which one of the queries above is certain to have an output that is a superset of the outputs of the other three queries? [2018]

- (A) Query 1 (B) Query 2
(C) Query 3 (D) Query 4

37. Consider the relations $r(A, B)$ and $s(B, C)$, where $s \cdot B$ is a primary key and $r \cdot B$ is a foreign key referencing $s \cdot B$. Consider the query

$$Q: r \bowtie (\sigma_{B < 5}(S))$$

Let LOJ denote the natural left outer-join operation. Assume that r and s contain no null values.

Which one of the following queries is NOT equivalent to Q ? [2018]

- (A) $\sigma_{B < 5}(r \bowtie s)$ (B) $\sigma_{B < 5}(r \text{ LOJ } s)$
(C) $r \text{ LOJ } (\sigma_{B < 5}(s))$ (D) $\sigma_{B < 5}(r) \text{ LOJ } s$

ANSWER KEYS

EXERCISES

Practice Problems 1

1. A 2. B 3. C 4. (i) B (ii) D (iii) D 5. C 6. C 7. (i) A (ii) C
8. (i) B (ii) A 9. A 10. A 11. (i) A (ii) B 12. (i) B (ii) B (iii) A
13. (i) A (ii) A (iii) A 14. (i) A (ii) A (iii) A 15. (i) A (ii) A (iii) A 16. C
17. A 18. D 19. A 20. D

Practice Problems 2

1. B 2. D 3. D 4. A 5. C 6. A 7. A 8. A 9. A 10. A
11. C 12. C 13. C 14. C 15. C 16. B 17. C 18. D 19. B 20. A

Previous Years' Questions

1. C 2. A 3. C 4. B 5. C 6. B 7. D 8. A 9. A 10. C
11. C 12. C 13. A 14. C 15. A 16. B 17. A 18. A 19. D 20. B
21. 19 22. A 23. C 24. A 25. D 26. D 27. D 28. 2 29. C 30. A
31. B 32. 2.6 33. D 34. 4 35. 7 36. D 37. C