

<p style="text-align: center;">BSCCS2001: Practice Assignment with Solutions Week 1</p>

1. A programmer is working on the data structures that store data internally in a database. At what level of abstraction is the programmer working?
- ☐ Logical Level
 - ☐ View Level
 - ☐ Programming Level
 - ✓ ☒ Physical Level

Solution:

Logical Level represents the relational model and conceptual schema of the data.

View Level represents the user's view of the data i.e how the user sees the data

Physical level deals with the actual storage structure of the data internally in the form of trees and other data structures.

Programming level is not a level of abstraction.

2. From among the given types of applications, choose the ones for which DBMS will be a preferred choice over filesystems.
- ✓ ☒ Applications with large datasets.
 - ✓ ☒ Applications with concurrent transactions.
 - ☐ Applications with small datasets.
 - ☐ Applications with no dedicated database administrators.

Solution:

For applications with small datasets, the overhead in installing the DBMS will be much more than the advantage obtained due to reduced retrieval time.

If there is no dedicated personnel for maintaining a database, the performance will begin to deteriorate after a period of time.

3. By the concept of Logical Data Independence, a change in the logical level of DBMS should not affect which other level(s) of abstraction?
- ✓ ☒ View Level
 - ☐ Physical Level

- ☐ Both Physical and View Level
- ☐ None of the above

Solution:

Logical Data Independence: A change in Logical Level of DBMS should not affect the View Level.

4. By the concept of Physical Data Independence, a change in the physical level of DBMS should not affect which other level(s) of abstraction?
- ☐ View Level
 - ☐ Logical Level
 - ✓ ☒ Both Logical and View Level
 - ☐ None of the above

Solution:

Physical Data Independence: A change in Physical Level of DBMS should not affect either the View Level or the Logical Level.

5. Which model is widely used during the planning and designing phase of a database system?
- ☐ View Model
 - ✓ ☒ Entity-Relationship Model
 - ☐ Object Model
 - ☐ Relational Model

Solution: Entity-Relationship Model is used to define a high level view of the data entities and the relationships between them. It is used mainly for designing and planning the database structure.

6. Which of the following are advantages of DBMS over file based data management applications?
- ✓ ☒ Easy recovery of data
 - ✓ ☒ Consistency of data
 - ✓ ☒ Efficiency of operation
 - ☐ Ease of initial setup

Solution: Initial system setup is costly in the case of DBMS, whereas in file based systems, it is relatively easy and economical.

7. Which among the following is a good option for exchanging data among different systems over the internet?

- ☐ HTML
- ☐ MS Access
- ☐ SQL
- ✓ ☒ XML

Solution: HTML is a markup language used mainly for data presentation, whereas XML is widely used for sharing data over different systems over the web. The other two are not meant for information exchange over the internet.

8. Planning what attributes should be placed in which table of a database is a part of

- ☐ Attribute Design
- ✓ ☒ Logical Design
- ☐ Physical Design
- ☐ Subsystem Design

Solution: The design of structure of tables is a part of logical design.

9. Which of the following options are examples of data transactions?

- ✓ ☒ Transferring funds using e-wallets
- ✓ ☒ Booking a reservation in railways
- ☐ Increasing the data storage capacity
- ✓ ☒ Updating KYC in bank.

Solution: Any event which modifies the current state (value) of the data is a data transaction.

Increasing storage capacity does not make any changes in the current data.

10. Which data model aptly satisfies the need of a system which maintains large set of complex interconnected data, where the semantics of interconnection changes dynamically (like a social networking site)?

- ☐ Relational Model
- ☐ XML Model
- ☐ Object-Relational Model
- ☒ Graph Model

Solution: Refer Lecture 1.4.

<p>BSCCS2001: Practice Assignment with Solutions</p> <p>Week 2</p>
--

Modules covered:

1. Attribute Types, Relation Schema and Instance, Keys, Relational Query Languages
2. Operations, Select, Project, Union, Difference, Intersection, Cartesian Product
3. Natural Join, Aggregate Operations
4. Introduction to SQL - History of SQL, Data Definition Language (DDL), Basic Query Structure (DML)
5. Additional Basic Operations, Set Operations, Null Values, Aggregate Functions

1. Consider the table **taskAssignment** (in Figure 1) which represents tasks assigned to each employee for a given day. [MCQ: 2 points]

taskAssignment					
employee_num	task_num	task_duration	date_of_assignment	supervisor_num	location
101	P103	7	10-01-2020	112	Block-C
102	P103	5	10-01-2020	112	Block-C
101	P103	4	11-01-2020	112	Block-C
101	P103	6	12-01-2020	112	Block-C
104	P102	6	10-01-2020	111	Block-B
105	P101	7	10-01-2020	110	Block-A
104	P101	7	11-01-2020	110	Block-A
105	P101	6	11-01-2020	110	Block-A
102	P102	6	11-01-2020	111	Block-B

Figure 1: Table **taskAssignment**

Select the appropriate compound key for the table.

- ☐ { *employee_num* }
- ☐ { *employee_num*, *task_num* }
- ☒ { *employee_num*, *task_num*, *date_of_assignment* }
- ☐ { *employee_num*, *supervisor_num* }

Solution: A compound key is a set of more than one attribute which uniquely identifies all rows of a relation.

The entries for { *employee_num* } are not unique for all the tuples. It is neither a set of more than one attribute nor uniquely identifies all rows of the given relation. Thus, it cannot be a compound key.

The entries for { *employee_num*, *task_num* } are not unique for all the tuples. Thus, it cannot be a compound key.

The entries for { *employee_num*, *task_num*, *date_of_assignment* } are unique for all the rows. So, **it is a valid candidate key for the given relation.**

The entries for { *employee_num*, *supervisor_num* } are not unique for all the tuples. Thus, it cannot be a compound key.

2. Consider the following relational schema on students of a school. [MCQ: 2 points]
studentInfo (*enrollment_num*, *class*, *section*, *roll*, *name*).
 $\{enrollment_num\}$ and $\{class, section, roll\}$ are two possible candidate keys. What is the maximum number of possible superkeys of **studentInfo**?

- ☐ 16
☒ 18
☐ 20
☐ 22

Solution: This question will be discussed in the *Solve with the Instructor* session.

3. Consider the tables **vendor** and **component** as shown in Figure 2. The table **component** has attribute *vendor_num* which is a foreign key that refers to table **vendor**(*vendor_num*).

vendor		
vendor_num	vendor_name	vendor_location
10	YADAV	CHENNAI
11	AKHTAR	KOLKATA
12	PRASAD	TRICHY
13	SHARMA	BENGALURU

component			
item_num	name	cost	vendor_num
1011	RAM	2500.00	11
1012	CPU	8000.50	12
1013	MONITOR	5000.00	10
1014	KEYBOARD	500.50	13
1013	MONITOR	2250.00	13
1014	KEYBOARD	450.50	11
1011	RAM	3300.00	10

Figure 2: Tables **vendor** and **component**

Identify the appropriate “CREATE TABLE” statement for table **component**. [MCQ: 2 points]

- ☐ CREATE TABLE component(
item_num int NOT NULL,
name varchar(20),
cost numeric(6, 2) NOT NULL,
vendor_num int NOT NULL,
PRIMARY KEY (item_num),
FOREIGN KEY (vendor_num) REFERENCES vendor(vendor_num));
- ☐ CREATE TABLE component(
item_num int NOT NULL,
name varchar(20),
cost numeric(6, 2) NOT NULL,
vendor_num int NOT NULL,
PRIMARY KEY (item_num, name),
FOREIGN KEY (vendor_num) REFERENCES vendor(vendor_num));
- ☐ CREATE TABLE component(
item_num int NOT NULL,
name varchar(20),
cost numeric(6, 2) NOT NULL,
vendor_num int NOT NULL,
PRIMARY KEY (item_num, vendor_num),
FOREIGN KEY (vendor_num) REFERENCES vendor(item_num, vendor_num));
- ☒ CREATE TABLE component(
item_num int NOT NULL,
name varchar(20),


```
cost numeric(6, 2) NOT NULL,  
vendor_num int NOT NULL,  
PRIMARY KEY (item_num, vendor_num),  
FOREIGN KEY (vendor_num) REFERENCES vendor(vendor_num));
```

Solution:

Option 1: since the table **component** has duplicate values in the column *item_num*, the attribute set $\{item_num\}$ cannot be a primary key. Hence, option 1 is incorrect.

Option 2: since table **component** has duplicate values in columns $\{item_num, name\}$, the attribute set $\{item_num, name\}$ cannot be a primary key. Hence, option 2 is incorrect.

Option 3: since the number of columns in the foreign key does not match the number of columns in the referenced table, option 3 is incorrect.

Option 4: since table **component** has unique entries in columns $\{item_num, vendor_num\}$, attribute set $\{item_num, vendor_num\}$ is the proper primary key and hence, option 4 is the correct option.

4. Identify the appropriate “ALTER TABLE” statement for table **vendor** such that we can add a column named *vendor_phone* to store the phone number of the vendors. [MSQ: 2 points]

- ☐ ALTER TABLE vendor APPEND COLUMN vendor_phone numeric(10);
- ✓ ☒ ALTER TABLE vendor ADD COLUMN vendor_phone numeric(10);
- ✓ ☒ ALTER TABLE vendor ADD vendor_phone numeric(10);
- ☐ ALTER TABLE vendor INSERT COLUMN vendor_phone numeric(10);

Solution: To add a new column in the given table, we use “**ADD**” or “**ADD COLUMN**” commands. Thus, option 2 and option 3 are correct.

5. Identify the correct statement(s) about a Foreign Key. [MSQ: 2 points]

- ☐ A FOREIGN KEY is a data element/attribute within a data field of a data record that is not unique, and cannot be used to distinguish one data record in a database from another data record within a database table.
- ☐ A FOREIGN KEY constraint prevents data from being inserted into the foreign key column.
- ✓ A FOREIGN KEY is a data element/attribute within a data field of a data record within a database table that refers to either a primary key or an attribute with unique constraint.
- ✓ A FOREIGN KEY can be added at the time of ALTER TABLE query.

Solution: Option 1 - A FOREIGN KEY is a data element/attribute within a data field of a data record that is unique, and this can be used to distinguish one data record in a database from another data record within a database table.

Option 2 - The FOREIGN KEY constraint prevents invalid data from being inserted into the foreign key column.

Option 3, 4 are factual statements about Foreign Key.

6. Consider the table **Employee** given in Figure 3.

Employee		
ID	Name	Salary
1	MIKE	35
2	KYLE	50
3	JAMES	50
4	JONES	NULL
5	LIMA	70
6	PACY	50

Figure 3: Table Employee

Determine the correct relation based on the three queries given below: [MCQ: 2 points]

- `SELECT COUNT(DISTINCT Salary) as A FROM Employee;`
- `SELECT COUNT(*) as B FROM Employee;`
- `SELECT COUNT(Salary) as C FROM Employee;`
 - ☐ value of $B > A$ and $B = C$
 - ☐ value of $A > B$ and $B > C$
 - ☐ value of $A > B$ and $B = C$
 - ☒ value of $B > A$ and $B > C$

Solution: This question will be discussed in the *Solve with the Instructor* session.

7. Consider the table **Employee** given in Figure 4.

Employee		
ID	Name	Salary
1	MIKE	35
2	KYLE	50
3	JAMES	50
4	JONES	NULL
5	LIMA	70
6	PACY	50

Figure 4: Table Employee

Determine the suitable query that returns the following table:

[MCQ: 2 points]

AVG_SAL
55

- ☐ SELECT AVG(Salary) AS AVG_SAL
FROM Employee WHERE Salary IS NOT NULL AND ID>5;
- ☐ SELECT AVG(Salary) AS AVG_SAL
FROM Employee WHERE Salary IS NOT NULL AND Salary>50;
- ☐ SELECT AVG(Salary) AS AVG_SAL
FROM Employee WHERE Salary IS NOT NULL AND ID>3;
- ✓ ☒ SELECT AVG(Salary) AS AVG_SAL
FROM Employee WHERE Salary IS NOT NULL AND Salary>35;

Solution: From the table **Employee**, NOT NULL values that are greater than 35 are 50, 50, 70, 50. The average of 50, 50, 70, 50 is 55.

8. Identify the output for the following SQL statement.

[MCQ: 2 points]

```
SELECT max(temperature) - min(temperature)
FROM weatherReport
WHERE state='Karnataka';
```

☐ 4

☒ 5

☐ 2

☐ 0

Solution: The part of the statement:

```
WHERE state='Karnataka';
```

extracts all rows having *state* as “Karnataka”.

`max(temperature)` returns the maximum *temperature* from among the given cities in “Karnataka”, i.e. 36 (for “Bellary”).

`min(temperature)` returns the minimum *temperature* from among the given cities in “Karnataka”, i.e. 31 (for “Bengaluru”).

Therefore, the output is $36 - 31 = 5$;

9. Using the table **Citizen** given in Figure 5, answer the question that follows.

Citizen				
ID	profession	lastname	firstname	
23	clerk	Holmes	Mark	
45	firefighter	Singh	Vikram	
23	police	Samson	Rana	
31	clerk	Butler	Jones	
67	gardener	Holmes	John	

Figure 5: Table Citizen

Identify the correct SQL statement(s) to create the given table. [MSQ: 2 points]

- ☐ CREATE TABLE Citizen (ID int NOT NULL, profession varchar(255), lastname varchar(255) NOT NULL, firstname varchar(255), PRIMARY KEY (ID));
- ✓ ☒ CREATE TABLE Citizen (ID int NOT NULL, profession varchar(255), lastname varchar(255) NOT NULL, firstname varchar(255), PRIMARY KEY (ID, lastname));
- ✓ ☒ CREATE TABLE Citizen (ID int, profession varchar(255), lastname varchar(255) NOT NULL, firstname varchar(255), PRIMARY KEY (firstname, lastname));
- ✓ ☒ CREATE TABLE Citizen (ID int NOT NULL, profession varchar(255), lastname varchar(255) NOT NULL, firstname varchar(255), PRIMARY KEY (ID, profession));
- ☐ CREATE TABLE Citizen (ID int NOT NULL, profession varchar(255), lastname varchar(255) NOT NULL, firstname varchar(255), PRIMARY KEY (profession));

Solution: Only an attribute, or a set of attributes that can uniquely identify a row, can be chosen as the PRIMARY KEY. Here (ID, lastname), (firstname, lastname) and (ID, profession) can do so. Hence, options 2, 3, 4 are correct.

10. Using the table **Citizen** given in Figure 6, answer the question that follows.

Citizen			
ID	profession	lastname	firstname
23	clerk	Holmes	Mark
45	firefighter	Singh	Vikram
23	police	Samson	Rana
31	clerk	Butler	Jones
67	gardener	Holmes	John

Figure 6: Table Citizen

Let the table **Citizen** be created using the SQL statement given below: [MSQ: 2 points]

- `CREATE TABLE Citizen (ID int, profession varchar(255), lastname varchar(255) NOT NULL, firstname varchar(255));`

Identify the correct INSERT INTO statement for this table.

- ✓ `INSERT INTO Citizen (ID, profession, lastname, firstname) VALUES (23,'clerk', 'Holmes', 'Mark');`
- `INSERT INTO Citizen TABLE VALUES (23,'clerk', 'Holmes', 'Mark');`
- ✓ `INSERT INTO Citizen VALUES (23,'clerk', 'Holmes', 'Mark');`
- `INSERT INTO Citizen VALUES ('23','clerk', 'Holmes', 'Mark');`

Solution: To insert values in a table, the generic format is -
`INSERT INTO table_name (column1, column2, ...) VALUES (value1, value2, ...);`
OR
`INSERT INTO table_name VALUES (value1, value2, ...);`
Also, varchar type attributes need to be in ' ' and INT type attributes without ' '.

11. Using the table **Citizen** given in Figure 7, answer the question that follows.

Citizen				
ID	profession	lastname	firstname	
23	clerk	Holmes	Mark	
45	firefighter	Singh	Vikram	
23	police	Samson	Rana	
31	clerk	Butler	Jones	
67	gardener	Holmes	John	

Figure 7: Table Citizen

Let the table **Citizen** be created with *firstname* as the primary key, and the values be inserted as per the schema in the table. [MCQ: 1 points]

Choose the SQL statement to remove the primary key constraint Citizen_pkey of the table **Citizen**.

- ☐ ALTER TABLE Citizen
DROP Citizen CONSTRAINT Citizen_pkey;
- ☐ MODIFY TABLE Citizen
DROP CONSTRAINT Citizen_pkey;
- ☐ MODIFY TABLE Citizen
DROP Citizen CONSTRAINT;
- ☒ ALTER TABLE Citizen
DROP CONSTRAINT Citizen_pkey;

Solution: The syntax to remove an existing primary key constraint is -
ALTER TABLE table_name DROP CONSTRAINT primary_key_constraint_name;
Hence option 4 is correct.
ALTER TABLE Citizen DROP CONSTRAINT Citizen_pkey;

12. Let $\{sup_num\}$ be the primary key of table **suppliers** and $\{part_num, sup_num\}$ be the primary key of table **parts**. [NAT: 2 points]
Consider the SQL query given below:

```
SELECT s.sup_num, sum(p.part_qty)
FROM suppliers s, parts p WHERE s.sup_num = p.sup_num
GROUP BY s.sup_num
HAVING SUM(p.part_qty) > 70
```

How many rows will be returned by the above SQL query?

Answer: 2

Solution: As per the given SQL statement, it first performs a Cartesian product between **suppliers** and **parts**, which output all possible combinations from both the tables.

The part of the statement:

s.sup_num = p.sup_num

eliminates the rows which do not satisfy the condition. The output is as shown below:

s.sup_num	s.sup_name	p.part_num	p.sup_num	p.part_qty
1001	Able	301	1001	32
1001	Able	302	1001	16
1002	Peter	301	1002	41
1002	Peter	302	1002	11
1002	Peter	304	1002	35
1003	Molina	302	1003	36
1003	Molina	304	1003	40
1004	Nikki	301	1004	17
1004	Nikki	303	1004	25

The part of the statement:

SELECT s.sup_num, sum(p.part_qty) ...GROUP BY s.sup_num
results in:

s.sup_num	SUM(p.part_qty)
1001	48
1002	87
1003	76
1004	42

Finally, the part of the statement: **HAVING SUM(p.part_qty) > 70**
results in:

s.sup_num	SUM(p.part_qty)
1002	87
1003	76

Thus, the result has 2 rows.

BSCCS2001: Practice Assignment with Solutions

Week 3

1. Consider the relational schema given in Figure 1.

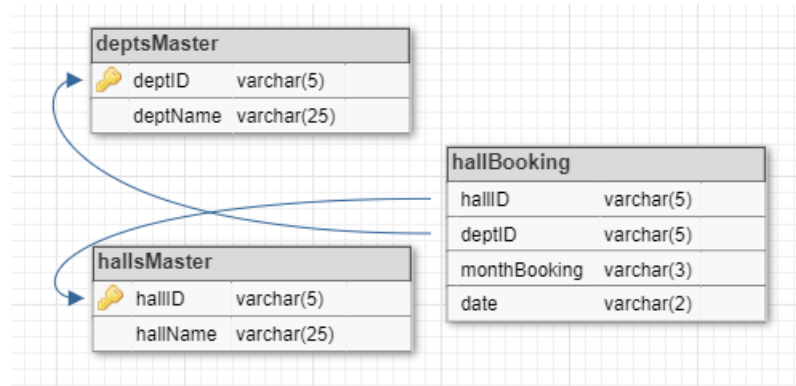


Figure 1: Hall Booking Relational Schema

Find the names of departments that have booked all the halls at least once in the month of January. [MCQ: 2 points]

- ☐ `SELECT deptName FROM deptsMaster
WHERE deptID IN
(SELECT DISTINCT deptID FROM hallBooking AS hb1
WHERE NOT EXISTS
(SELECT hm.hallID FROM hallsMaster AS hm
INTERSECT
SELECT hb2.hallID FROM hallBooking AS hb2
WHERE hb1.deptID = hb2.deptID
AND monthBooking = 'Jan'));`
- ✓ `SELECT deptName FROM deptsMaster
WHERE deptID IN
(SELECT DISTINCT deptID FROM hallBooking AS hb1
WHERE NOT EXISTS
(SELECT hm.hallID FROM hallsMaster AS hm
EXCEPT
SELECT hb2.hallID FROM hallBooking AS hb2
WHERE hb1.deptID = hb2.deptID
AND monthBooking = 'Jan'));`
- ☐ `SELECT deptName FROM deptsMaster
WHERE deptID IN
(SELECT DISTINCT deptID FROM hallBooking AS hb1
WHERE EXISTS`

```

        (SELECT hm.hallID FROM hallsMaster AS hm
        EXCEPT
        SELECT hb2.hallID FROM hallBooking AS hb2
        WHERE hb1.deptID = hb2.deptID
        AND monthBooking = 'Jan')));

```

○

```

SELECT deptName FROM deptsMaster
WHERE deptID IN
    (SELECT DISTINCT deptID FROM hallBooking AS hb1
    WHERE EXISTS
        (SELECT hm.hallID FROM hallsMaster AS hm
        INTERSECT
        SELECT hb2.hallID FROM hallBooking AS hb2
        WHERE hb1.deptID = hb2.deptID
        AND monthBooking = 'Jan')));

```

Solution:

```

SELECT hm.hallID FROM hallsMaster AS hm
        EXCEPT
        SELECT hb2.hallID FROM hallBooking AS hb2
        WHERE hb1.deptID = hb2.deptID
        AND monthBooking = 'Jan'

```

The above query fetches all hallIDs that have not been booked in January.

```

SELECT DISTINCT deptID FROM hallBooking AS hb1
        WHERE NOT EXISTS (hallIDs that have not been booked in January)

```

The above query will retrieve all department IDs that have not booked any halls that have not been booked in January.

```







SELECT deptName FROM deptsMaster
WHERE deptID IN (all department IDs that have not booked any halls
that have not been booked in January)

```

The above query fetches the names of departments that have not booked any halls that have not been booked in January.

Note that if we execute the nested queries directly on the sql prompt, they will give errors due to the aliases used.

2. Consider the relational schema given in Figure 2.

capital			
	countryID	varchar(25)	
	capitalID	varchar(25)	
	capitalName	varchar(25)	
 Add field			







country			
	countryID	varchar(25)	
	continent	varchar(25)	
	countryName	varchar(25)	
 Add field			

Figure 2: Country Capitals Relational Schema

What should be filled in the blank so that the following query will return the capitals of all countries that belong to Asia but not Europe? (Write the answer as a single word in all CAPS) [NAT: 2 points]

```
SELECT capitalName FROM capital
WHERE countryID IN (SELECT countryID FROM country
                    WHERE continent = 'Asia'
                    -----
                    SELECT countryID FROM country
                    WHERE continent = 'Europe');
```

Answer: EXCEPT

Solution: The first part of the inner query returns all countries that belong to Asia. If we need to find countries that belong to Asia but not Europe, then from the rows returned by the first part of the inner query, we have to remove those that contain countries that belong to Europe as well. Hence, to remove those rows, we use EXCEPT.

3. Based on the relations given in Figure 3 answer the question that follows.

employee			
empID	empName	deptID	desgID
E00001	Akash	D0002	G0001
E00002	Akshay	D0002	---(a)---
E00003	Subha	D0003	G0003
E00004	Lavanya	---(b)---	G0002
E00005	Diya	D0001	G0001

department	
deptID	deptName
D0001	Purchase
D0002	Sales
D0003	Accounts

designation		
desgID	desgName	Salary
G0001	Clerk	5000
G0002	Supervisor	7000
G0003	Manager	10000

Figure 3: Employee instance

What should be filled in blank (a) in the table **employee** in Figure 3, if the query given below returns the value: Akshay? [NAT: 2 points]

```
SELECT empName FROM employee
WHERE desgID LIKE '%2' AND deptID LIKE '%2';
```

Answer: G0002

Solution: Since the value returned is Akshay, it corresponds to second row of table **employee**. **desgID** in employee table is a foreign key that references department table, hence the **desgID** can only be any one value from {G0001, G0002, G0003}. The **WHERE** condition specifies **desgID** LIKE '%2'. It follows from all three reasons that the only possible value that can be filled in blank (a) is G0002.

4. Consider the following SQL statement:

[MSQ: 2 points]

```
CREATE TABLE boats(  
    boatID VARCHAR (8),  
    boatName VARCHAR (20),  
    boatColour VARCHAR (8),  
    yearOfPurchase INTEGER,  
    weight INTEGER,  
    PRIMARY KEY (boatID),  
    CHECK (boatColour IN ('Black', 'White', 'Red', 'Yellow')));
```

Which among the following will cause an integrity constraint violation in the `boats` table?

- ☐ INSERT INTO boats('B1', 'Liberty', 'Red', 2003, 500);
- ✓ ☒ INSERT INTO boats('B1', 'Liberty', 'Blue', 2003, 500);
- ✓ ☒ UPDATE boats SET boatColour = 'Green' WHERE boatID = 'B1';
- ☐ DELETE FROM boats;

Solution: In option 1, there is no constraint violation.

In option 2, since the permitted colors do not include blue, it will cause a violation.

In option 3, the permitted colors do not include green and hence it will cause a violation.

In option 4, there is no constraint violation.

5. Consider the relational schema given in Figure 4.

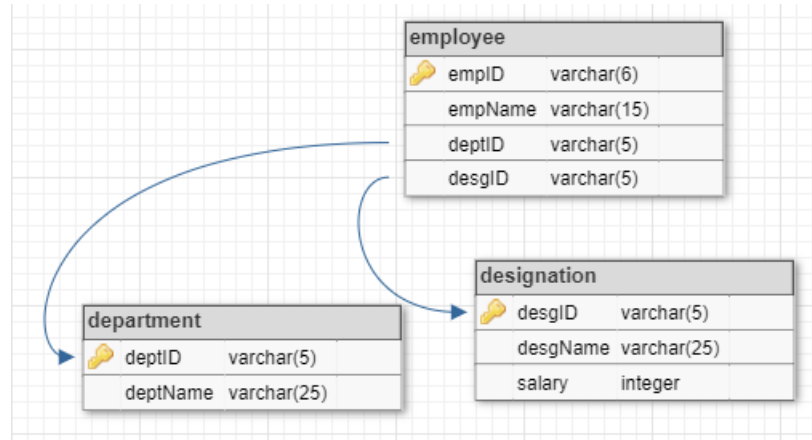


Figure 4: Employee Schema

If the relations **employee**, **designation** and **department** have 100, 6, 5 rows respectively, what is the difference between the maximum and the minimum number of rows returned by the following query? [NAT: 2 points]

```
SELECT * FROM employee LEFT OUTER JOIN designation
ON employee.desgID = designation.desgID;
```

Answer: 0

Solution: Left outer join (also known as Left join) returns all tuples returned by natural join along with those tuples in the left table (here, **employee**) that does not have matching entry in the right table. In the given question, however, **desgID** is the foreign key in Table **employee** that references Table **designation**. Therefore, there will not be any tuple in the left table that does not have a matching entry in the right table. Thus, the maximum number of rows returned by the left join in the given example is 100.

The case when **employee** table has no rows is the case when left outer join will have the minimum number of rows. In this case, however, the **employee** table has 100 rows. So, there will be at least 100 rows returned by the left join.

The answer is $100 - 100 = 0$.

6. Choose the appropriate query/queries to find the names of batsmen who scored the second-highest runs. [MSQ: 2 points]

- ✓ `SELECT name, MAX(runs) AS runs
FROM batsman WHERE runs < (SELECT MAX(runs) FROM batsman);`
- ✓ `SELECT name, MAX(runs) AS runs
FROM batsman WHERE runs IN
(SELECT runs FROM batsman MINUS (SELECT MAX(runs) FROM batsman));`
- ✓ `SELECT name, runs AS runs
FROM batsman WHERE runs = (SELECT runs FROM batsman
ORDER BY runs LIMIT 1,1);`
- ☐ `SELECT name, MAX(runs) AS runs FROM batsman
WHERE runs > (SELECT MIN(runs) FROM batsman);`

Solution:

- MAX, MIN functions are used to find out the record with maximum and minimum values respectively among a record set.
- The SQL MINUS operator is used to return all rows in the first SELECT statement that are not returned by the second SELECT statement.
- The LIMIT statement is used to limit the number of records returned based on a limit value.
- The ORDER BY keyword is used to sort the result-set in ascending or descending order.

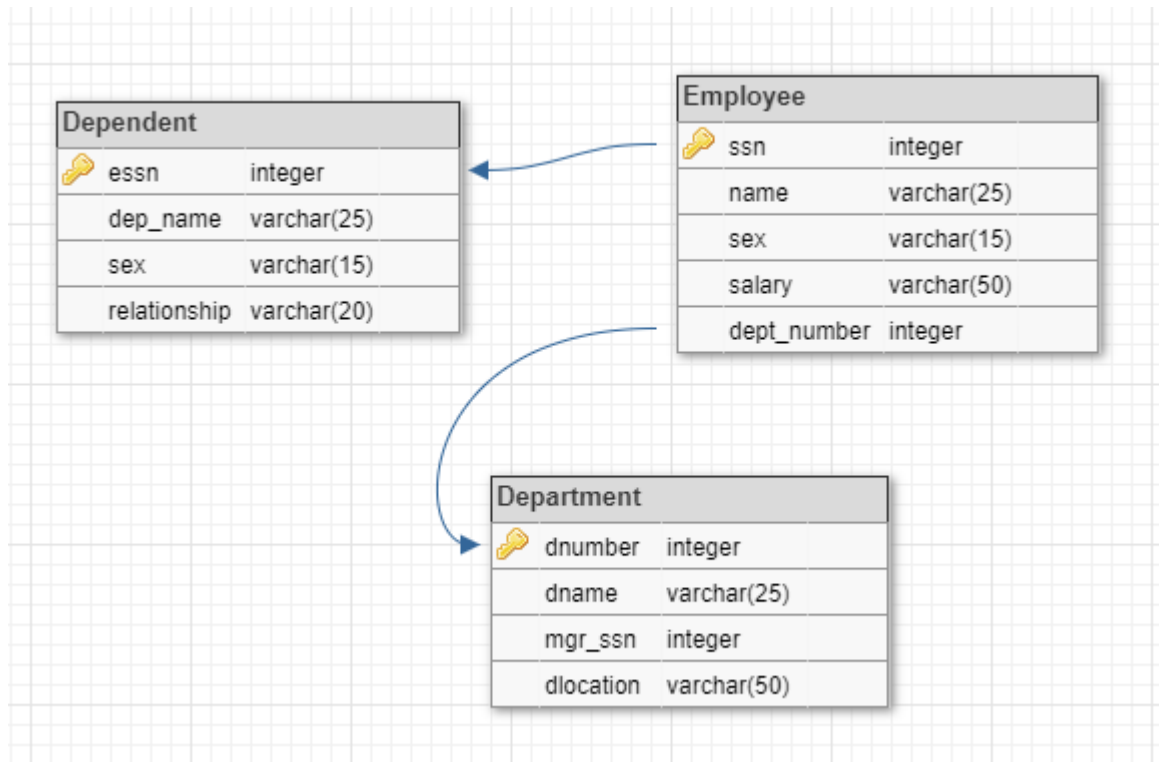
Option 1 - The inner query will fetch the maximum runs and then the outer query will return the runs value which is maximum among all and lesser than the value retrieved by the inner query. Hence, the second-highest value of runs is fetched.

Option 2 - The inner query will fetch all the runs values other than the maximum runs and from this set, the IN operator will retrieve the maximum value. Hence, the second-highest value of runs is fetched.

Option 3 - The inner query will fetch the second-highest value of runs using the Limit operator, then using the '=' the outer query will retrieve it.

Option 4 - The inner query will return the minimum runs and the outer query will fetch the maximum runs greater than the runs value of the inner query. Hence it is incorrect.

Consider the table **Employee**, table **Department** and table **Dependent**, and answer the questions 7 and 8.



7. Select the suitable query to retrieve the names of employees who have no dependents. [MCQ: 2 points]

- ☐ SELECT name FROM Employee
WHERE NOT EXISTS (SELECT * FROM Dependent AS D WHERE D.ssn = essn);
- ☐ SELECT name FROM Dependent
WHERE NOT EXISTS (SELECT * FROM Employee WHERE ssn = essn);
- ☒ SELECT name
FROM Employee
WHERE NOT EXISTS (SELECT * FROM Dependent WHERE ssn = essn);
- ☐ SELECT name FROM Employee
WHERE IN (SELECT * FROM Employee WHERE ssn = essn);

Solution: The EXISTS/NOT EXISTS condition in SQL is used to check whether the result of a correlated nested query is empty (contains no tuples) or not. As per the question, to retrieve the names of employees who have no dependents, the outer query needs to fetch data from the table Employee and the inner query needs

to fetch data from the table Dependent. Hence, options 2 and 4 are incorrect.

In option 1- After aliasing Dependent as D, the condition must be $ssn = D.essn$. Hence, option 1 is incorrect.

In option 3 - Inner query will fetch all the dependents where attribute ssn of Employee is matched with essn from Dependent. Hence, only if there is no matched value, NOT EXISTS will be true and the names of the employees who have no dependents will be retrieved.

8. Select the suitable query to retrieve the names of employees who have some dependent(s) whose name ends with 'KUMAR'. [MCQ: 2 points]

- ☐ SELECT name FROM Dependent
WHERE ssn IN (SELECT essn FROM Employee
WHERE dep_name LIKE '%KUMAR');
- ☐ SELECT name FROM Employee
WHERE essn IN (SELECT ssn FROM Dependent
WHERE dep_name LIKE '%KUMAR%');
- ☒ SELECT name FROM Employee
WHERE ssn IN (SELECT essn FROM Dependent
WHERE dep_name LIKE '%KUMAR');
- ☐ SELECT name FROM Employee
WHERE ssn IN (SELECT essn FROM Dependent
WHERE dep_name LIKE 'KUMAR');

Solution: The LIKE operator is used in a WHERE clause to search for a specified pattern in a column. The percent sign (%) represents zero, one, or multiple characters and the underscore sign (_) represents one, single character.

So to retrieve all the names ending with KUMAR, it has to match '%KUMAR'. Hence, options 2 and 4 are incorrect.

In option 1, the inner query needs to fetch from table Dependent and outer query from table Employee. Hence, incorrect.

In option 3, the inner query will fetch the Dependent(s) whose name ends with KUMAR and using IN keyword, the outer query will retrieve the names of the corresponding employees. Hence, correct.

9. Consider the table **employee** and table **department** as shown in Figure 5, and answer the question that follows. [MCQ: 2 points]

employee			
emp_name	emp_id	age	dept_id
WADE	1	23	10
MADDEN	4	54	10
HARM	6	34	13
TALLY	3	41	16
RODEY	2	46	14
JONES	7	38	14
MULE	5	49	16

department		
dept_name	dept_id	dept_location
MATHS	10	Houston
ENGLISH	15	San Antonio
PHYSICS	14	Houston
COMPUTER	13	New York
CHEMISTRY	16	Chicago

Figure 5: employee & department

What will be the output of the following query?

```
SELECT emp_id, dept_name
FROM employee NATURAL JOIN department
ORDER BY age desc;
```

✓ Output:

emp_id	dept_name
4	MATHS
5	CHEMISTRY
2	PHYSICS
3	CHEMISTRY
7	PHYSICS
6	COMPUTER
1	MATHS

○ Output:

emp_id	dept_name
1	MATHS
4	MATHS
6	COMPUTER
3	CHEMISTRY
2	PHYSICS
7	PHYSICS
5	CHEMISTRY

○ Output:

emp_id	dept_name
4	MATHS
3	CHEMISTRY
2	PHYSICS
5	CHEMISTRY
7	PHYSICS
6	COMPUTER
1	MATHS

○ Output:

emp_id	dept_name
4	MATHS
3	CHEMISTRY
7	PHYSICS
5	CHEMISTRY
2	PHYSICS
6	COMPUTER
1	MATHS

Solution: As per the query, after NATURAL JOIN on employee table and department table, the resultant table will be -

emp_id	dept_name
1	MATHS
4	MATHS
6	COMPUTER
3	CHEMISTRY
2	PHYSICS
7	PHYSICS
5	CHEMISTRY

And as and when we put ORDER BY age in descending order, we will fetch the following resultant table -

emp_id	dept_name
4	MATHS
5	CHEMISTRY
2	PHYSICS
3	CHEMISTRY
7	PHYSICS
6	COMPUTER
1	MATHS

10. Consider a table **Employee**(*eid*, *dept*, *ename*, *salary*, *ebonus*). The table has no records initially.

[MCQ:2 points]

```
CREATE OR REPLACE FUNCTION bonus_fun() RETURNS TRIGGER AS $$
BEGIN
    IF NEW.edept = 'R/D' THEN
        NEW.ebonus = NEW.esalary * .75;
    END IF;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER bonus_trig
BEFORE INSERT ON Employee
EXECUTE PROCEDURE bonus_fun();

INSERT INTO Employee VALUES (4,'R/D','Diksha',30000);
```

```
INSERT INTO Employee VALUES (2,'Accounts','Raj',40000);  
SELECT ebonus FROM Employee;
```

If the given code is executed, then what will be the output?

- ☐ 22500
0
- ☐ 22500
NULL
- ☐ 22500
30000
- ☒ The code has errors.

Solution: The code is erroneous because the trigger definition does not explicitly mention its granularity (*for each row* or *for each statement*). This trigger checks each insertion and modifies the value of an attribute (*ebonus*) when the described condition satisfies, therefore it should work as a row level trigger.

11. Consider an instance of the table **Employee** given below.

[MCQ:2 points]

eid [PK] integer	edep character varying	ename character varying	esalary integer
1	Accounts	Rekha	35000
2	HR	Joseph	30000
3	HR	Arif	50000
4	Development	Debraj	45000
5	Accounts	Abhijit	90000
6	Marketing	Shahid	76000
7	Sales	Shabana	25000
8	Marketing	Meenakshi	42000
9	Sales	Digvijay	66000
10	Marketing	Shashi	54000

Figure 6: Table: Employee

If the given code is executed on this instance, then what will be the output/error?

[MCQ:2 points]

```
CREATE OR REPLACE FUNCTION salary_fun() RETURNS TRIGGER AS $$
DECLARE
    counter INT := 0;
BEGIN
    IF NEW.esalary > 75000 THEN
        counter = counter + 1;
        RAISE NOTICE 'Number of affected rows : %', counter;
        --//This statement prints => NOTICE: <whatever follows>
    END IF;
    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER salary_trig
AFTER UPDATE ON Employee
FOR EACH ROW
EXECUTE PROCEDURE salary_fun();

UPDATE Employee SET esalary = esalary * 1.5;
```

- ☐ NOTICE: Number of affected rows : 4
- ☐ NOTICE: Number of affected rows : 10
- ☐ NOTICE: Number of affected rows : 4
 NOTICE: Number of affected rows : 4
 NOTICE: Number of affected rows : 4
 NOTICE: Number of affected rows : 4

✓ None of the above

Solution: The trigger will be executed on every row affected by the UPDATE statement. Only 4 rows in the given instance will have their new salary more than 75000. The RAISE NOTICE statement is inside the IF clause, thus it will be executed 4 times. Observe the fact that the trigger fires for each updated row, and hence every time the variable *counter* is reinitialized with 0. It will be incremented to 1 with respect to that specific row, and thus we will get the output as:

NOTICE: Number of affected rows : 1
 NOTICE: Number of affected rows : 1
 NOTICE: Number of affected rows : 1
 NOTICE: Number of affected rows : 1

12. If we want to store/print the number of affected rows when an update or delete statement is executed, then which type of trigger should we use to count?

[MCQ:2 points]

- ✓ Statement level trigger
- ☐ Row level trigger
 - ☐ Both are equally efficient
 - ☐ Table level trigger

Solution: If we want to count the number of affected rows then we need not execute a trigger every time for each row. After all the modifications are over, a single execution of a trigger to count the affected rows should be done. Running a row level trigger will simply do the same job again and again for all the affected rows. Hence, Statement level triggers should be used here.

BSCCS2001: Practice Assignment with Solutions

Week 4

1. Consider the following relations : [MCQ: 2 points]

A = (P, Q, R)

B = (X, Y, Z)

Let relations $a(A)$ and $b(B)$ be given. Which of the following expressions in the tuple relational calculus is equivalent to $\Pi_{P,Z}(\sigma_{R=X}(a \times b))$?

- ☒ $\{t \mid \exists p \in a, \exists q \in b(t[P] = p[P] \wedge t[Z] = q[Z] \wedge p[R] = q[X])\}$
- ☐ $\{t \mid \exists p \in a, \exists q \in b(t[P] = p[P] \wedge t[Z] = q[Z] \vee p[R] = q[X])\}$
- ☐ $\{t \mid \exists p \in a, \exists q \in b(t[P] = p[P] \wedge t[Z] = q[Z] \wedge p[X] = q[R])\}$
- ☐ $\{t \mid \exists p \in a, \exists q \in b(t[Z] = p[Z] \wedge t[P] = q[P] \wedge p[R] = q[X])\}$

Solution:

$\Pi_{P,Z}(\sigma_{R=X}(a \times b))$ will return attributes P from relation $a(A)$ and Z from relation $b(B)$. So, options C and D are incorrect.

For SELECT operation, condition $R = X$ must be satisfied, So option B is incorrect. Thus, option 1 is correct.

Consider the following relational schema and answer questions 2 and 3.

[MCQ:2 points]

- *Owner*(*aadhar_number*, *o_name*)
 - *Vehicle*(*v_number*, *v_model*)
 - *Registration*(*aadhar_number*, *v_number*, *purchase_year*)
2. Which of the following relational algebra expressions is equivalent to the statement given below?

- Find the Aadhaar numbers of owners who purchased the vehicle model V20 after year 2020.
- ☐ $\sigma_{aadhar_number}(\Pi_{v_model="V20" \vee purchase_year > "2020"}(Registration \bowtie Vehicle))$
- ☐ $\sigma_{aadhar_number}(\sigma_{v_model="V20" \wedge purchase_year > "2020"}(Registration \bowtie Vehicle))$
- ☒ $\Pi_{aadhar_number}(\sigma_{v_model="V20" \wedge purchase_year > "2020"}(Registration \bowtie Vehicle))$
- ☐ $\Pi_{aadhar_number}(\sigma_{v_model="V20" \vee purchase_year > "2020"}(Registration \bowtie Vehicle))$

Solution: Selection Operator (σ), selects those rows or tuples from the relation that satisfies the selection condition.

Project operator is denoted by the symbol Π , and it is used to select desired columns (or attributes) from a table (or relation).

Option 1 and Option 2 are incorrect. Here, the SELECT operator is used, requiring a specific condition to select tuples from a relation.

Option 3: It will return the Aadhaar number of all the owners who purchased the vehicle model V20 **and** after year 2020.

So, Option 3 is correct.

Option 4: It will return the Aadhaar number of all the owners who purchased the vehicle model V20 **or** after year 2020.

3. Which of the following queries is equivalent to the statement given below?

- Find the names of all owners who purchased vehicles with number 123 before the year 2019.

- ☐ $\{T \mid \exists O \in Owner, \exists R \in Registration(O.aadhar_number = R.aadhar_number \wedge R.v_number = 123 \vee R.purchase_year < 2019 \wedge T.o_name = O.o_name)\}$
- ☐ $\{T \mid \exists O \in Owner, \exists R \in Registration(O.aadhar_number = R.aadhar_number \wedge R.v_number = 123 \vee R.purchase_year < 2019 \vee T.o_name = O.o_name)\}$
- ☐ $\{T \mid \exists O \in Owner, \exists R \in Registration(O.aadhar_number = R.aadhar_number \vee R.v_number = 123 \wedge R.purchase_year < 2019 \vee T.o_name = O.o_name)\}$
- ☒ $\{T \mid \exists O \in Owner, \exists R \in Registration(O.aadhar_number = R.aadhar_number \wedge R.v_number = 123 \wedge R.purchase_year < 2019 \wedge T.o_name = O.o_name)\}$

Solution:

$\exists O \in Owner, \exists R \in Registration(O.aadhar_number = R.aadhar_number)$ will perform the NATURAL JOIN operation of Owner and Registration schema and $(R.v_number = 123 \wedge R.purchase_year < 2019 \wedge T.o_name = O.o_name)$ is the required condition for the names of all the owners who purchased the vehicle number 123 before year 2019.

4. Consider the E-R diagram for a cricket-training-camp database as given in Figure 1.

[MCQ: 2 points]

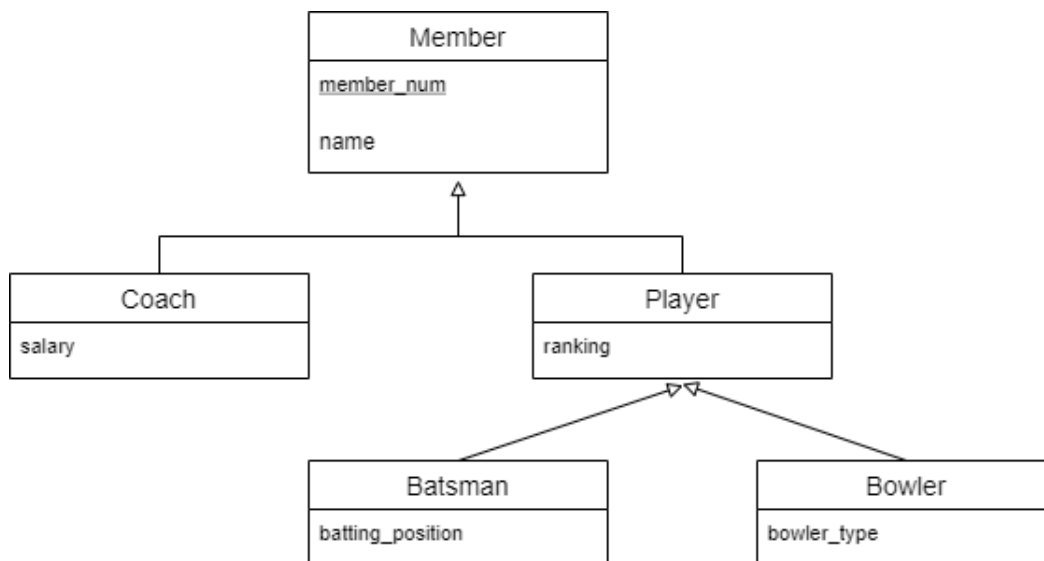


Figure 1: E-R diagram of cricket-training-camp database

Identify the option in which both the statements correctly describe the relations between the given entity sets.

- ☐ 1. Each member can be either a coach or a player or both in the cricket-training-camp.
2. Each player can be a batsman or a bowler or both.
- ☐ 1. Each member can be a coach or a player or both.
2. Each player can be either a batsman or a bowler. However, a player cannot be both, a batsman and a bowler at the same time.
- ☐ 1. Each member can be either a coach or a player. But, a member cannot be a coach and a player at the same time.
2. Each player can be either a batsman or a bowler, but cannot be both.
- ✓ 1. Each member can be either a coach or a player or just a member of the cricket-training-camp. But, a member cannot be a coach and a player at the same time.
2. Each player can be a batsman or a bowler or both.

Solution:

- **Coach** and **Player** are disjoint specializations of **Member**.

- **Batsman** and **Bowler** are overlapping specializations of **Member**.
- Both kind of specializations given in Figure 1 are partial specializations.

Hence,

- Each member must be a coach or a player or just a member. But, a member cannot be a coach and a player at the same time.
- Each player can be a batsman or a bowler or both.

5. Consider the E-R diagram given in Figure 2.

[MCQ: 2 points:Solve with instructor]

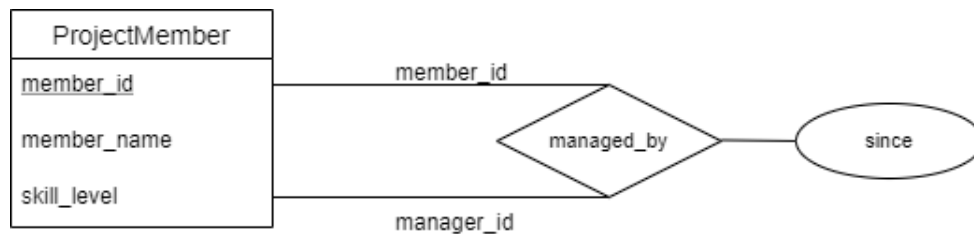


Figure 2: E-R diagram

The table for entity set **ProjectMember** is created using the command below:

```
CREATE TABLE ProjectMember(  
  member_id INT NOT NULL,  
  member_name VARCHAR(20) NOT NULL,  
  skill_level VARCHAR(20) NOT NULL,  
  PRIMARY KEY (member_id)  
);
```

Select the appropriate command to create the table for relationship set **managed_by**.

- ☐ CREATE TABLE managed_by(
 member_id INT NOT NULL,
 since INT NOT NULL,
 PRIMARY KEY (member_id),
 FOREIGN KEY (member_id) REFERENCES ProjectMember(member_id)
);
- ☒ CREATE TABLE managed_by(
 member_id INT,
 manager_id INT,
 since INT NOT NULL,
 PRIMARY KEY (member_id, manager_id),
 FOREIGN KEY (member_id) REFERENCES ProjectMember(member_id),
 FOREIGN KEY (manager_id) REFERENCES ProjectMember(member_id)
);
- ☐ CREATE TABLE managed_by(
 manager_id INT NOT NULL,
 since INT NOT NULL,
 PRIMARY KEY (manager_id),

```
FOREIGN KEY (manager_id) REFERENCES ProjectMember(member_id)
);
```

- CREATE TABLE managed_by(
member_id INT NOT NULL,
manager_id INT NOT NULL,
since INT NOT NULL,
PRIMARY KEY (member_id, manager_id),
FOREIGN KEY (manager_id) REFERENCES ProjectMember(member_id)
);

Solution: The table **managed_by** must have $\{member_id, manager_id\}$ as primary key, both reference to **ProjectMember**(*member_id*), and the descriptive attribute *since* also becomes an attribute in the table.

Thus, it must be created by the command:

```
CREATE TABLE managed_by(  
member_id INT,  
manager_id INT,  
since INT NOT NULL,  
PRIMARY KEY (member_id, manager_id),  
FOREIGN KEY (member_id) REFERENCES ProjectMember(member_id),  
FOREIGN KEY (manager_id) REFERENCES ProjectMember(member_id)  
);
```

Please note that since $\{member_id, manager_id\}$ is the primary key, the prime attributes *member_id* and *manager_id* by default not NULL.

6. Consider the entity set given in Figure 3.

[MCQ: 2 points]

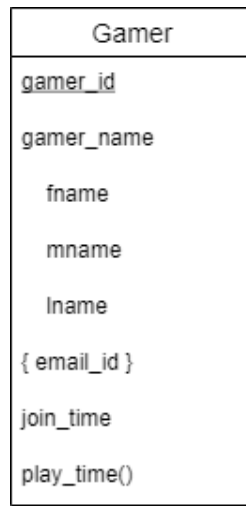


Figure 3: Entity set **Gamer**

Which of the following relational schemas appropriately represents the E-R diagram?

- ☐ **Gamer**(gamer_id, *gamer_name*, *fname*, *mname*, *lname*, *email_id*, *join_time*)
- ☐ **Gamer**(gamer_id, *fname*, *mname*, *lname*, *email_id*, *join_time*, *play_time*)
- ☐ **Gamer_email**(gamer_id, email_id)
- ☒ **Gamer**(gamer_id, *fname*, *mname*, *lname*, *join_time*)
- ☒ **Gamer_email**(gamer_id, email_id)
- ☐ **Gamer**(gamer_id, *email_id*, *join_time*)
- ☐ **Gamer_name**(gamer_id, *gamer_name*, *fname*, *mname*, *lname*)

Solution:

- The identifying attribute *gamer_id* becomes primary key for the schema.
- The composite attribute *gamer_name* will be replaced by its parts *fname*, *mname* and *lname* in the schema.
- The simple attribute *join_time* becomes another attribute.
- The derived attribute *play_time* does not need to be added in the schema.
- For the multivalued attribute *email_id* a separate relation has to be created which will be:
Gamer_email(gamer_id, email_id).

7. Consider the relations given below: [MSQ: 2 points:Solvewithinstructor]

- **doctor**(doc_id, doc_name, specialization)
- **patient**(patient_num, patient_name)
- **operationRoster**(doc_id, patient_num, operation_cost)

Identify the appropriate statement(s) to find the names of all doctors having specialization in orthopedics and who have charged more than \$1000 as surgery charges.

- ✓ $\Pi_{doc_name}(\sigma_{specialization="orthopedic" \wedge operation_cost > 1000}(doctor \bowtie operationRoster))$
- $\{s \mid \exists s \in doctor, \exists r \in operationRoster(s.doc_id = r.doc_id \wedge s.specialization = "orthopedic" \wedge r.operation_cost > 1000)\}$
- ✓ $\{t \mid \exists s \in doctor, \exists r \in operationRoster(s.doc_id = r.doc_id \wedge s.specialization = "orthopedic" \wedge r.operation_cost > 1000 \wedge t.doc_name = s.doc_name)\}$
- ✓ $\{< D_N > \mid \exists D_I \exists R_P \exists R_C (< D_I, D_N, "orthopedic" > \in doctor \wedge < D_I, R_P, R_C > \in operationRoster) \wedge R_C > 1000\}$

Solution: As per the specifications given in the question, a natural join needs to be applied between **doctor** and **operationRoster** as:

$doctor \bowtie operationRoster$.

Then, a select operation can be applied as:

$\sigma_{specialization="orthopedic" \wedge operation_cost > 1000}(doctor \bowtie operationRoster)$.

Finally, apply project the *doc_name* as:

$\Pi_{doc_name}(\sigma_{specialization="orthopedic" \wedge operation_cost > 1000}(doctor \bowtie operationRoster))$.

The equivalent **tuple relational calculus** is:

$\{t \mid \exists s \in doctor, \exists r \in operationRoster(s.specialization = "orthopedic" \wedge r.operation_cost > 1000 \wedge t.doc_name = s.doc_name)\}$.

The equivalent **domain relational calculus** is:

$\{< D_N > \mid \exists D_I \exists R_P \exists R_C (< D_I, D_N, "orthopedic" > \in doctor \wedge < D_I, R_P, R_C > \in operationRoster) \wedge R_C > 1000\}$.

Please note the tuple relation calculus:

$\{s \mid \exists s \in doctor \exists r \in operationRoster(s.doc_id = r.doc_id \wedge s.specialization = "orthopedic" \wedge r.operation_cost > 1000)\}$

projects all attributes rather than *doc_name* alone.

8. Consider the relations below: [MSQ: 2 points]

- **customer**(customer_id, customer_name, customer_city)
- **invoice**(invoice_number, customer_id, amount_payable)

Choose the correct relational algebra expressions that return the names of all customers having amount payable (*amount_payable*) more than \$1,000 and who are located in **Chennai**.

- $\Pi_{customer_name}(\sigma_{amount_payable > 1000 \wedge customer_city = "Chennai"}(customer \times invoice))$
- $\Pi_{amount_payable > 1000 \vee customer_city = "Chennai"}(customer \bowtie invoice)$
- ✓ $\Pi_{customer_name}(\sigma_{amount_payable > 1000 \wedge customer_city = "Chennai"}(customer \bowtie invoice))$
- ✓ $\Pi_{customer_name}(\sigma_{amount_payable > 1000 \wedge customer_city = "Chennai" \wedge customer.customer_id = invoice.customer_id}(customer \times invoice))$

Solution: First, natural join can be applied between the two relations **customer** and **invoice** such that tuples will be combined by equality in *customer_id*. Then, σ with predicate $amount_payable > 1000 \wedge customer_city = "Chennai"$ can be applied to find the tuples as per the specification given.

Alternatively, the same can also be achieved by a Cartesian product between **customer** and **invoice** along with σ with predicate equality of *customer_id* of both the relations and σ with predicate $amount_payable > 1000 \wedge customer_city = "Chennai"$.

Finally, we project (Π) the *customer_name*.

9. Consider the table given in Figure 4.

[MSQ: 2 points]

S1	
A	B
5	25
6	36
7	49
8	64
9	91

Figure 4: Relation S1

Choose the correct set of expressions that will return the tuple given below.

A
7

- ☐ $\sigma_A(\Pi_{B=49}(S1))$
- ☒ $\{t \mid \exists p \in S1(t[A] = p[A] \wedge p[B] = 49)\}$
- ☐ $\{t \mid \exists p \in S1(t[A] = p[A] \wedge p[B] = 7)\}$
- ☒ $\{<a> \mid \exists b(<a, b> \in S1 \wedge b = 49)\}$

Solution:

$\sigma_A(\Pi_{B=49}(S1))$ is logically incorrect, the correct TRC query is $\Pi_A(\sigma_{B=49}(S1))$, this will first perform the Select operation and return the row having B = 49 then it will project the corresponding value of attribute A.

$\{t \mid \exists p \in S1(t[A] = p[A] \wedge p[B] = 49)\}$ is equivalent to $\Pi_A(\sigma_{B=49}(S1))$

$\{<a> \mid \exists b(<a, b> \in S1 \wedge b = 49)\}$ is equivalent to $\Pi_A(\sigma_{B=49}(S1))$

10. Consider the E-R diagram in Figure 5.

[NAT: 2 points]

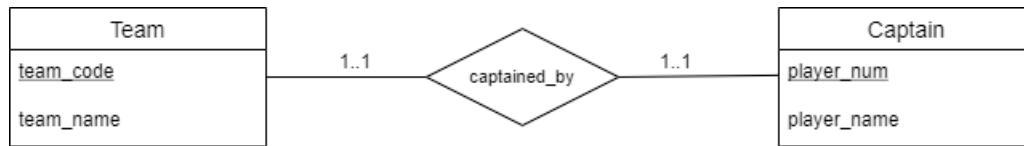


Figure 5: ERD

What is the minimum number of tables needed to represent this E-R diagram?

Solution: 1

The minimum and maximum cardinality is 1 (1..1).

- A minimum value of 1 indicates total participation.
- A maximum value of 1 indicates that the entity participates in at most one relationship.

Thus, it can be represented using a single table:

team_captain(team_code, team_name, player_num, player_name).

Consider the E-R diagram given in Figure 6 and answer the questions 11 and 12.

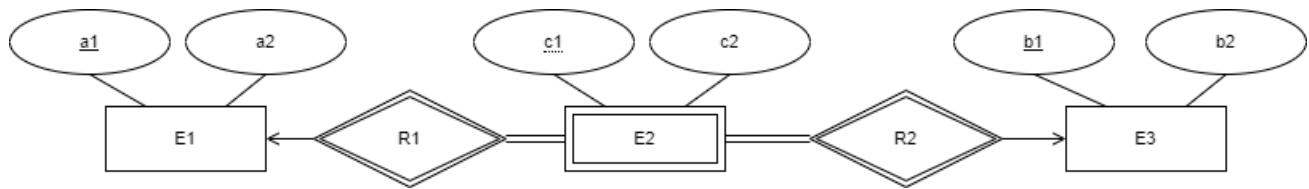


Figure 6: E-R diagram

11. The minimum number of tables required to represent the entity sets and relationship sets is [NAT: 2 points]

Answer: 3

Solution: 3

- **E1** is associated with **E2** via **R1** in a one-to-many relation.
- **E3** is associated with **E2** via **R2** in a one-to-many relation.

Many-to-one and one-to-many relationship sets that are total on the many-side can be represented by adding an extra attribute to the “many” side, containing the primary key of the “one” side. Thus, we can represent the entire ERD using 3 tables as follows:

- **E1**(a1, a2)
- **E2**(c1, c2, a1, b1)
- **E3**(b1, b2)

12. What will be the correct attribute set for the table corresponding to the entity set **E2**? [MCQ: 2 points:Solvewithinstructor]

- ☐ **E2**(c1, c2)
- ☐ **E2**(c1, a1, c2)
- ☐ **E2**(c1, a1, b1, c2)
- ☒ **E2**(c1, c2, a1, b1)

Solution:

- **R1** is a one-to-many relationship set from **E2** to **E1**.
- **R2** is a one-to-many relationship set from **E2** to **E3**.

Many-to-one and one-to-many relationship sets that are total on the many-side can be represented by adding an extra attribute to the “many” side, containing the primary key of the “one” side. Thus, we can represent the entire ERD using 3 tables as follows:

- **E1**(a1, a2)
- **E2**(c1, c2, a1, b1)
- **E3**(b1, b2)

13. Consider the E-R diagram with aggregation given in Figure 7.

[MCQ: 2 points]

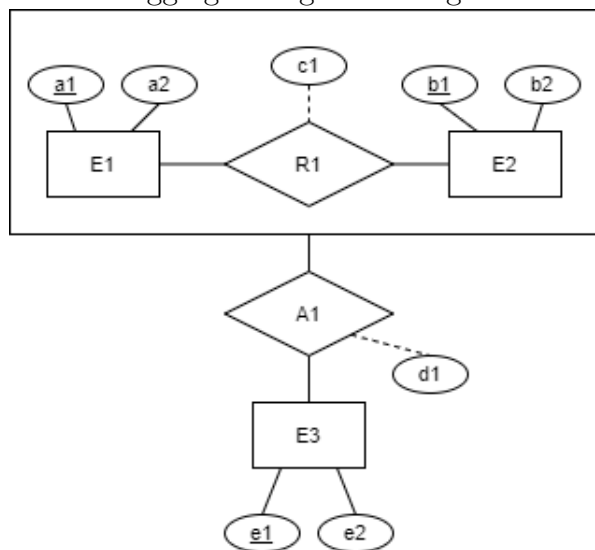


Figure 7: ERD

What will be the correct attribute set for the table corresponding to relationship-set **A1**?

- ☐ c1, e1, d1
- ☐ a1, b1, d1, c1, e1, e2
- ☐ a1, b1, d1, e1, e2
- ☒ a1, b1, e1, d1

Solution: The ER-diagram presents a scenario of aggregation. Thus, the relationship set **A1** must be mapped to a table having the following:

- Primary keys of **E1**, **E2** and **E3**.
- Any descriptive attributes of **A1**.

So the attribute set for **A1** is: $\{a1, b1, e1, d1\}$

BSCCS2001: Practice Solutions
Week 5

1. Consider the following

[MCQ: 2 points]

$X = \{A \rightarrow BC, B \rightarrow A, C \rightarrow A\}$

$Y = \{A \rightarrow B, B \rightarrow C, C \rightarrow A\}$

- ☐ X covers Y
- ☐ Y covers X
- ☐ X and Y are equivalent
- ☒ All the above

Solution: Let us check X covers Y , every functional dependency in Y logically implies in X .

FDs in Y : $A \rightarrow B, B \rightarrow C, C \rightarrow A$

check for $A \rightarrow B$,

$A^+ \rightarrow ABC \{A \rightarrow BC \text{ in } X\}$

Check for $B \rightarrow C$

$B^+ \rightarrow BAC, \{A \rightarrow BC, B \rightarrow A \text{ in } X\}$

Check for $C \rightarrow A$

$C^+ \rightarrow CAB \{C \rightarrow A, A \rightarrow BC \text{ in } X\}$

Therefore, X covers Y

Let us now check, If Y covers X

FDs in X : $A \rightarrow BC, B \rightarrow A, C \rightarrow A$

Check for $A \rightarrow BC$

$A^+ \rightarrow ABC \{A \rightarrow B, B \rightarrow C \text{ in } Y\}$

Check for $B \rightarrow A$

$B^+ \rightarrow BCA \{B \rightarrow C, C \rightarrow A \text{ in } Y\}$

Check for $C \rightarrow A$

$C^+ \rightarrow A \{C \rightarrow A \text{ in } X\}$

Therefore, Y covers X

Since, both X & Y covers each other, so they are equivalent $X \equiv Y$

2. Consider relation $\mathbf{R}(A,B,C,D,E,F)$ with the following functional dependencies:

$$\mathcal{F} = \{AB \rightarrow C, AC \rightarrow E, EF \rightarrow D, AB \rightarrow F\}$$

[MSQ: 2 points]

Which among the following is true about R ?

- ☐ $R1(D, E, F), R2(A, B, C, E, F)$ is a lossy decomposition of R .
- ☒ $R1(A, B, D), R2(C, E, F)$ is a lossy decomposition of R .
- ☐ $R1(A, D, E, C), R2(B, E, C)$ is a lossless decomposition of R .
- ☒ $R1(D, E, A, B), R2(C, F, B)$ is a lossy decomposition of R .

Solution:

This problem will be solved in the *Solve With the Instructor* session.

3. Consider relation **T20WC** defined as **W**(*Team, Ranking, Captain, Points, Players*) with the following functional dependencies:

$$\mathcal{F} = \{Team, Ranking \rightarrow Captain, Ranking \rightarrow Players, Captain \rightarrow Points\}$$

Then, which of the following is true ?

[MCQ:2points]

- ☐ $W1(Team, Ranking, Captain), W2(Captain, Points)$ is a lossless-join decomposition.
- ☒ $W1(Team, Ranking, Captain), W2(Points, Players)$ is a lossy-join decomposition
- ☐ $W1(Team, Ranking, Captain), W2(Ranking, Points, Players)$ is a lossless-join decomposition.
- ☐ None of the above

Solution:

Option 1 : $W1(Team, Ranking, Captain), W2(Captain, Points)$

$$W1 \cup W2 \neq W$$

Thus, it is lossy join decomposition.

Option 2 : $W1(Team, Ranking, Captain), W2(Points, Players)$

$$W1 \cup W2 = W$$

$$W1 \cap W2 = \phi$$

Thus, it is lossy join decomposition. So, option 2 is correct

Option 3 : $W1(Team, Ranking, Captain), W2(Ranking, Points, Players)$

$$W1 \cup W2 = W$$

$$W1 \cap W2 = Ranking$$

$$Ranking^+ \rightarrow Ranking, Players$$

Ranking is not superkey for any relation. Hence, we can't determine $W1$ and $W2$ from it. So, it is lossy decomposition.

Numerical Answer Type

4. Consider the relation **Book**(*Author, Publisher, Pages, Ratings, Type*) having the following functional dependencies : [NAT: 2 points]

$\mathcal{F} = \{ \textit{Author} \rightarrow \textit{Publisher}, \textit{Pages}$

$\textit{Publisher} \rightarrow \textit{Ratings}$

$\textit{Pages}, \textit{Ratings} \rightarrow \textit{Type}$

$\textit{Type} \rightarrow \textit{Author} \}$

What is the maximum number of candidate keys for **Book**?

Ans : 4

Solution: By estimating the closure of all combination of attributes, it can be observed that the closure of the following attributes produces all other attributes:

Author, Type, (Publisher, Pages), (Pages, Ratings).

Hence, these 4 are the candidate keys.

5. In a relation $\mathbf{R}(A, B, C, D, E)$, each attribute is a candidate key. Then, what is the maximum number of super keys possible for \mathbf{R} ?

[NAT: 2 points]

Ans: 31

Solution: Consider a relation $R(A_1, A_2, A_3, \dots, A_n)$, then maximum number of super keys are $2^n - 1$. (If Each attribute of a relation is candidate key)
Here, $n = 5$, so, the number of super keys for a given relation R is 31.

6. Which among the following is/are the use(s) of finding closure of attributes? [MSQ: 2 points]

- ☐ Find if an attribute or set of attributes is a superkey.
- ☐ Compute the canonical cover of a given set of functional dependencies
- ☐ Test if a specific functional dependency holds
- ☐ Compute the closure of a given set of functional dependencies
- ✓ All the above

Solution: The solution follows from the lectures.

7. Let $\mathbf{R}(A, B, C, D, E)$ be a relation with the following functional dependencies:

$$\mathcal{F} = \{A \rightarrow C, A \rightarrow B, C \rightarrow D, BC \rightarrow E\}$$

[MCQ: 2 points]

Then,

- ☐ $\mathcal{F}^+ = \{A \rightarrow C, A \rightarrow B, C \rightarrow D, BC \rightarrow E\}$
- ☐ $\mathcal{F}^+ = \{A \rightarrow C, A \rightarrow B, C \rightarrow D, A \rightarrow D, BC \rightarrow E\}$
- ☐ $\mathcal{F}^+ = \{A \rightarrow BC, C \rightarrow D, BC \rightarrow E\}$
- ☒ $\mathcal{F}^+ = \{C \rightarrow D, BC \rightarrow E, A \rightarrow BCDE\}$

Solution: This problem will be solved in the *Solve With the Instructor* session.

8. Let $\mathbf{R}(A, B, C, D, E)$ be a relation with the following functional dependencies:

[MCQ: 2 points]

$$\mathcal{F} = \{A \rightarrow B, C \rightarrow D, BD \rightarrow E\}$$

Then, which of the following functional dependencies can be derived from \mathcal{F} using Armstrong's Axioms?

☒ $AC \rightarrow E$

☐ $BE \rightarrow D$

☐ $B \rightarrow A$

☐ $C \rightarrow E$

Solution:

$$A \rightarrow B$$

$$AC \rightarrow BC \text{ \{Augmentation\}}$$

$$C \rightarrow D$$

$$BC \rightarrow BD \text{ \{Augmentation\}}$$

$$BD \rightarrow E$$

$$AC \rightarrow E \text{ \{Transitivity\}}$$

9. Choose the correct canonical cover of the set of functional dependencies \mathcal{F} that occur in a relation $\mathbf{R}(A, B, C, D)$, where

$$\mathcal{F} = \{A \rightarrow BC, AB \rightarrow C, A \rightarrow D, D \rightarrow C\}$$

[MCQ: 2 points]

- ☐ $A \rightarrow BC, AB \rightarrow C$
- ☐ $A \rightarrow BC, AB \rightarrow C, A \rightarrow D$
- ☒ $A \rightarrow B, A \rightarrow D, D \rightarrow C$
- ☐ $A \rightarrow B, B \rightarrow C, A \rightarrow D$

Solution: Given $A \rightarrow BC, AB \rightarrow C, A \rightarrow D, D \rightarrow C$.

$$A \rightarrow D, D \rightarrow C \Rightarrow A \rightarrow C$$

That is, in the FD $A \rightarrow BC$, $A \rightarrow C$ is redundant.

Hence we can remove $A \rightarrow C$ from $A \rightarrow BC$.

$$\mathcal{F} = \{A \rightarrow B, AB \rightarrow C, A \rightarrow D, D \rightarrow C\}$$

Since $A \rightarrow C$ is a stronger constraint than $AB \rightarrow C$ and since $A \rightarrow C$ can be derived from $A \rightarrow D, D \rightarrow C$, we can remove $AB \rightarrow C$ from \mathcal{F} .

Therefore

$$\mathcal{F} = \{A \rightarrow B, A \rightarrow D, D \rightarrow C\}$$

10. Choose the set of FDs equivalent to:

$$A \rightarrow BC, B \rightarrow CE, C \rightarrow ED$$

[MSQ: 2 points]

☒ $A \rightarrow BE, B \rightarrow CD, C \rightarrow ED$

☐ $A \rightarrow B, B \rightarrow D, C \rightarrow E$

☒ $A \rightarrow B, B \rightarrow C, C \rightarrow ED$

☐ $A \rightarrow B, B \rightarrow C, C \rightarrow D, D \rightarrow A$

Solution:

This problem will be solved in the *Solve With the Instructor* session.

11. Given the relation **hospital** and its decomposition into **hosp1** and **hosp2** as shown in Figure 1, choose the correct set of options. [MCQ: 2 points]

hospital		
hospitalNum	patientNum	doctorID
H0001	P0001	D0001
H0002	P0002	D0002
H0003	P0001	D0003

hosp1		hosp2	
hospitalNum	patientNum	patientNum	doctorID
H0001	P0001	P0001	D0001
H0002	P0002	P0002	D0002
H0003	P0001	P0001	D0003

Figure 1: Decomposition of **hospital** relation

- ☐ The given decomposition is lossless and the natural join of **hosp1** and **hosp2** has 5 rows.
- ☐ The given decomposition is lossless and the natural join of **hosp1** and **hosp2** has 3 rows.
- ☒ The given decomposition is lossy and the natural join of **hosp1** and **hosp2** has 5 rows.
- ☐ The given decomposition is lossy and the natural join of **hosp1** and **hosp2** has 3 rows.

Solution:

hospital		
hospitalNum	patientNum	doctorID
H0001	P0001	D0001
H0002	P0002	D0002
H0003	P0001	D0003
H0001	P0001	D0003
H0003	P0001	D0001

Figure 2: Natural join of **hosp1** and **hosp2**

12. Consider a relation $R(A, B, C, D, E)$ having the following functional dependencies:

$$\mathcal{F} = \{A \rightarrow BCD, D \rightarrow E, C \rightarrow D\}$$

Which among the following are lossy decompositions?

[MSQ: 2 points]

- ☐ $R_1(A, B, C), R_2(B, C, D), R_3(C, D, E)$
- ☐ $R_1(A, B, C), R_2(A, C, D), R_3(A, D, E)$
- ☒ $R_1(A, B, C), R_2(A, C), R_3(A, D)$
- ☐ $R_1(A, B, C, D), R_2(A, C, D, E)$

Solution:

This problem will be solved in the *Solve With the Instructor* session.

BSCCS2001: Practice with Solutions
Week 6

1. Consider the relational schema $\mathbf{R}(A, B, C, D, E)$, where the domains of A, B, C, D and E include only atomic values. Identify the possible set of functional dependencies that \mathbf{R} can have such that \mathbf{R} is in BCNF.

[MSQ: 2 points]

- ☒ FD: $\{AB \rightarrow CDE\}$
☐ FD: $\{AB \rightarrow CD, B \rightarrow E\}$
☐ FD: $\{AB \rightarrow CD, C \rightarrow D, D \rightarrow E\}$
☐ FD: $\{AB \rightarrow CDE, D \rightarrow A, E \rightarrow B\}$

Solution: Given that in \mathbf{R} each attribute is a single-valued attribute. Thus \mathbf{R} is already in 1NF.

Option-1: FD: $\{AB \rightarrow CDE\}$

The only candidate key (thus primary key) is: AB as $(AB)^+ = \{ABCDE\}$.

As all the non-prime attributes are fully functionally dependent on the candidate key, it is already in 2NF.

$\{AB \rightarrow CDE\}$, where AB is a superkey. Thus, **it is in 3NF and also in BCNF.**

Option-2: $\{AB \rightarrow CD, B \rightarrow E\}$

The only candidate key (thus primary key) is: AB as $(AB)^+ = \{ABCDE\}$.

$B \rightarrow E$ is a partial functional dependency. Thus, it is in 1NF but not in 2NF.

Option-3: FD: $\{AB \rightarrow CD, C \rightarrow D, D \rightarrow E\}$

The only candidate key (thus primary key) is: AB as $(AB)^+ = \{ABCDE\}$.

There is no partial functional dependency. Thus, it is already in 2NF.

$AB \rightarrow CD$, where AB is superkey.

But, for $C \rightarrow D, D \rightarrow E$

- the functional dependencies are not trivial.
- L.H.S of the functional dependencies are not superkeys.
- R.H.S of the functional dependencies are not prime attributes.

Thus, these two FDs violate 3NF rules. So, \mathbf{R} is in 2NF but not in 3NF based on this set of FDs.

Option-4: FD: $\{AB \rightarrow CDE, D \rightarrow A, E \rightarrow B\}$

The candidate keys are: AB and DE as $(AB)^+ = \{ABCDE\}$ and $(DE)^+ = \{ABCDE\}$. The prime attributes are A, B, D, E .

There is no partial functional dependency. Thus, it is already in 2NF.

$AB \rightarrow CDE$, where AB is superkey.

For $D \rightarrow A, E \rightarrow B$ R.H.S of the functional dependencies are prime attributes.

Thus, it is in 3NF. However, These two FDs do not satisfy BCNF (as L.H.S are not superkeys). So, \mathbf{R} is in 3NF but not in BCNF based on this set of FDs.

2. Consider the relational schema $\mathbf{R}(A, B, C, D, E, F)$, where the domains for A, B, C, D, E and F include atomic values only. If \mathbf{R} satisfies the functional dependencies $\{AB \rightarrow CDE, E \rightarrow F, BF \rightarrow A, C \rightarrow B\}$, then identify the correct statement(s).

[MSQ: 2 points]

- ☐ \mathbf{R} is in 1NF but not in 2NF
- ☒ \mathbf{R} is in 2NF and also in 3NF
- ☒ \mathbf{R} is in 3NF but not in BCNF
- ☐ \mathbf{R} is in 3NF also in BCNF

Solution:

Candidate keys are: AB, BF, AC, BE, CF and CE . So, prime attributes are: A, B, C, E and F . For the FDs: $E \rightarrow F$ and $C \rightarrow B$, B and F are prime attributes. Thus, there is no partial dependency, thus R is in 2NF.

$AB \rightarrow CDE$ and $BF \rightarrow A$, as AB and BF both are candidate keys, the FDs are in 3NF.

$C \rightarrow B$ and $E \rightarrow F$ also in 3NF, since B and F are prime attributes. Thus, R is in 3NF.

$C \rightarrow B$ and $E \rightarrow F$ violate BCNF conditions as C and E are not superkeys. Thus, R is not in BCNF.

3. Consider the relational schema $\mathbf{Z}(P, Q, R, S)$ and the following functional dependencies on \mathbf{Z} . [MCQ: 2 points]

- $P \rightarrow QRS$
- $Q \rightarrow R$
- $RS \rightarrow P$

Which of the following is/are correct?

- ☐ \mathbf{Z} is in 3NF and also in BCNF
- ☒ \mathbf{Z} is in 3NF but not in BCNF
- ☐ \mathbf{Z} is in 2NF but not in 3NF
- ☐ \mathbf{Z} is in BCNF but not in 3NF

Solution: FD = $\{P \rightarrow QRS, Q \rightarrow R, RS \rightarrow P\}$

$P^+ = PQRS$

$RS^+ = PQRS$

$QS^+ = PQRS$

So, candidate keys are P , QS & RS and prime attribute are P , Q , R & S .

Since the schema \mathbf{Z} has no partial dependencies or transitive dependencies, so it is in 3NF.

Check for BCNF

$P \rightarrow QRS$ (P is candidate key) ✓

$Q \rightarrow R$ (Q is not candidate key) ✗

$RS \rightarrow P$ (RS is candidate key) ✓

So, \mathbf{Z} is in 3NF but not in BCNF.

4. Let $\mathbf{R}(P, Q, R, S, T, U, V, W)$ be a relation (all attributes have atomic values only) with the following functional dependencies:

- $\{PQ \rightarrow RSTU\}$
- $\{P \rightarrow R\}$
- $\{Q \rightarrow S\}$
- $\{R \rightarrow UV\}$
- $\{V \rightarrow W\}$
- $\{W \rightarrow U\}$
- $\{V \rightarrow U\}$

Find the highest normal form in which the relation \mathbf{R} is in.

[MCQ: 2 points]

☒ 1NF

☐ 2NF

☐ 3NF

☐ BCNF

Solution: Since all attributes in \mathbf{R} have atomic values, it follows that \mathbf{R} is in 1NF.

In order to check if \mathbf{R} is in 2NF, we must find the candidate keys. Using the given FDs, we find that PQV is the only candidate key. Hence P , Q and V are the prime attributes and the rest are non-prime.

Now due to the presence of partial dependency, the relation \mathbf{R} is not in 2NF.

Note: Partial dependency occurs when a non-prime attribute is functionally dependent on a subset of a candidate key.

5. Consider the instance of relation **Course** given in Figure 1.

[MSQ: 2 points]

course_name	instructor	book	edition
DBMS	Geeta	DBMS-Beginner	3
DBMS	Arjun	DBMS-Beginner	3
DBMS	Geeta	DBMS-Expert	2
DBMS	Arjun	DBMS-Expert	2
Java	Rahul	Java-Beginner	5
Java	Rahul	Java-Intermediate	3
Java	Rahul	Java-Expert	4
Java	Armaan	Java-Beginner	5
Java	Armaan	Java-Intermediate	3
Java	Armaan	Java-Expert	4

Figure 1: An instance of relation **Course**

Which among the following multivalued dependencies can be inferred from the given information?

- ☒ $course_name \twoheadrightarrow instructor$
- ☐ $course_name \twoheadrightarrow book$
- ☐ $course_name \twoheadrightarrow edition$
- ☒ $course_name \twoheadrightarrow book, edition$

Solution:

Let us first number the tuples as t_1, t_2, \dots, t_{10} .

Test for $course_name \twoheadrightarrow instructor$:

In relation **Course**, there exist two tuples t_1 and t_2 such that

$t_1[course_name] = t_2[course_name]$.

We also have two tuples t_3 and t_4 in **Course** with the following properties:

- $t_1[course_name] = t_2[course_name] = t_3[course_name] = t_4[course_name]$,
- $t_3[instructor] = t_1[instructor]$ and $t_2[instructor] = t_4[instructor]$,
- $t_1[book, edition] = t_2[book, edition]$ and $t_3[book, edition] = t_4[book, edition]$.

Thus it satisfies MVD conditions.

In the relation **Course**, there are three tuples t_5, t_6 and t_7 such that

$t_5[course_name] = t_6[course_name] = t_7[course_name]$.

We also have three tuples t_8, t_9 and t_{10} in **Course** with the following properties:

- $t_5[course_name] = t_6[course_name] = t_7[course_name] = t_8[course_name] = t_9[course_name] = t_{10}[course_name]$,

- $t_5[instructor] = t_6[instructor] = t_7[instructor]$ and $t_8[instructor] = t_9[instructor] = t_{10}[instructor]$,
- $t_5[book, edition] = t_8[instructor, edition]$,
 $t_6[book, edition] = t_9[instructor, edition]$
and $t_7[book, edition] = t_{10}[book, edition]$.

Thus, MVD conditions are satisfied.

Test for $course_name \twoheadrightarrow book, edition$:

MVD Complementation rule: In a relation R , if $X \twoheadrightarrow Y$, then $X \twoheadrightarrow R - XY$.

Since we already have $course_name \twoheadrightarrow instructor$, it follows that $course_name \twoheadrightarrow book, edition$ also correct.

If we follow the same procedures as discussed above, we will be able to show that the MVDs:

$course_name \twoheadrightarrow book$

$course_name \twoheadrightarrow edition$

are not satisfied on relation **Course**.

6. Consider the relational schema:

Intern(*intern_code*, *intern_name*, *project*, *hobby*).

An intern can work in several projects and can have several hobbies. However, it maintains the FD: $intern_code \rightarrow intern_name$.

Identify the most appropriate 4NF decomposition for the given schema.

[MCQ: 2 points]

- ☐ **R1**(*intern_code*, *intern_name*, *project*, *hobby*), **R2**(*intern_code*, *project*, *hobby*)
- ☐ **R1**(*intern_code*, *intern_name*, *project*), **R2**(*intern_code*, *hobby*)
- ☐ **R1**(*intern_code*, *intern_name*, *hobby*), **R2**(*intern_code*, *project*)
- ☒ **R1**(*intern_code*, *intern_name*), **R2**(*intern_code*, *project*), **R3**(*intern_code*, *hobby*)

Solution:

From the given information in the question, *intern_code* cannot be a super key for the given relation. Thus, $intern_code \rightarrow intern_name$ violates BCNF conditions.

Thus, a possible BCNF decomposition would be:

R1(*intern_code*, *intern_name*), where *intern_code* is the candidate key, and **R2**(*intern_code*, *project*, *hobby*).

R2 violates 4NF conditions as it has the following MVDs:

$intern_code \twoheadrightarrow project$, and

$intern_code \twoheadrightarrow hobby$

So the 4NF decomposition is:

R2(*intern_code*, *project*), and

R3(*intern_code*, *hobby*).

Thus, the 4NF decomposition is:

R1(*intern_code*, *intern_name*),

R2(*intern_code*, *project*),

R3(*intern_code*, *hobby*).

7. Let $\mathbf{S}(Y, U, V)$ be a relation. Let $\mathbf{R}(P, W, X, Y, Z)$ be another relation with the following functional dependencies:

$$\mathcal{F} = \{X \rightarrow ZW, Y \rightarrow X, W \rightarrow P\}$$

\mathbf{R} contains 300 tuples and \mathbf{S} contains 250 tuples. What is the maximum number of tuples possible as output of $\mathbf{R} \bowtie \mathbf{S}$?

[MCQ: 2 point]

- ☐ 75000
- ☒ 250
- ☐ 300
- ☐ 50

Solution: From the given set of functional dependencies, Y is a candidate key of relation \mathbf{R} . So all 300 values of Y must be unique in \mathbf{R} .

There is no functional dependency given for S and to get the maximum number of tuples in output, there can be two possibilities for S .

- All 250 values of Y in \mathbf{S} are same and there is an entry in \mathbf{R} that matches with this value. In this case, we get 250 tuples in output.
- All 100 values of Y in \mathbf{S} are different and these values are present in \mathbf{R} also. In this case also, we get 250 tuples.

8. Let $\mathbf{A}(T, U, V, W)$ be a relational schema with the following functional dependencies:
 $\mathcal{F} = \{W \rightarrow UT, UV \rightarrow W, V \rightarrow T, W \rightarrow U\}$
It is given that \mathbf{A} is not in BCNF.
Suppose \mathbf{A} is decomposed into two relational schemas, $\mathbf{B}(TV)$ and $\mathbf{C}(UVW)$.
Which of the following statement(s) is/are correct?

[MSQ: 2 points]

- ☐ Decomposition of schema \mathbf{A} into \mathbf{B} and \mathbf{C} is dependency preserving
- ☒ Decomposition of schema \mathbf{A} into \mathbf{B} and \mathbf{C} is lossless
- ☐ Both \mathbf{B} and \mathbf{C} are in BCNF
- ☒ Relation \mathbf{B} is in BCNF

Solution:

- $\mathbf{B}(TV)$ preserves $\{V \rightarrow T\}$ and has V as the candidate key. So, relation \mathbf{B} is in BCNF.
- $\mathbf{C}(UVW)$ preserves $\{UV \rightarrow W, W \rightarrow U\}$ and has UV and VW as the candidate keys. So, relation \mathbf{C} is in 3NF but not in BCNF, as W is not a superkey.
- The decomposition of schema \mathbf{A} into two relational schemas, \mathbf{B} and \mathbf{C} , does not cover all the functional dependencies of the original relation \mathbf{A} . Hence, it is not dependency preserving.
- The decomposition has common attribute (i.e., V) which is superkey of relation $\mathbf{B}(TV)$, so decomposition of \mathbf{A} into \mathbf{B} and \mathbf{C} is lossless join decomposition.

9. Consider the relational schema:

prescription(*doctor_id*, *doctor_name*, *patient_id*, *patient_name*, *medicine_id*, *medicine_name*),
where the domains of all the attributes consist of atomic values. Consider the following
FDs for the relation *department*.

[MCQ: 2 points]

- $doctor_id \rightarrow doctor_name$,
- $patient_id \rightarrow patient_name$,
- $medicine_id \rightarrow medicine_name$,
- $doctor_id \twoheadrightarrow patient_id$,
- $doctor_id \twoheadrightarrow medicine_id$

From among the decompositions given, identify the one that is in 4NF.

- ☐ (*doctor_id*, *doctor_name*),
(*patient_id*, *patient_name*),
(*medicine_id*, *medicine_name*),
- ☐ (*doctor_id*, *doctor_name*),
(*patient_id*, *patient_name*),
(*medicine_id*, *medicine_name*),
(*doctor_id*, *patient_id*, *medicine_id*)
- ☐ (*doctor_id*, *doctor_name*, *patient_id*, *patient_name*),
(*doctor_id*, *doctor_name*, *medicine_id*, *medicine_name*)
- ☒ (*doctor_id*, *doctor_name*),
(*patient_id*, *patient_name*),
(*medicine_id*, *medicine_name*),
(*doctor_id*, *patient_id*),
(*doctor_id*, *medicine_id*)

Solution: For the given relation, the candidate key is $\{doctor_id, patient_id, medicine_id\}$ and it is in 1NF. However, it is not in 2NF as the FDs:

$doctor_id \rightarrow doctor_name$,

$patient_id \rightarrow patient_name$,

$medicine_id \rightarrow medicine_name$, are partial functional dependencies. Thus, a possible decomposition is:

R1(*doctor_id*, *doctor_name*), where *doctor_id* is the candidate key,

R2(*patient_id*, *patient_name*), where *patient_id* is the candidate key,

R3(*medicine_id*, *medicine_name*), where *medicine_id* is the candidate key,

R4(*doctor_id*, *patient_id*, *medicine_id*), where $\{doctor_id, patient_id, medicine_id\}$ is the candidate key,

R1, **R2**, **R3** and **R4** are already in 3NF and BCNF.

R1, **R2** and **R3** are already in 4NF. The MVDs,
 $doctor_id \twoheadrightarrow patient_id$, and
 $doctor_id \twoheadrightarrow medicine_id$ violate 4NF conditions. Thus, **R4** is decomposed as:
R41($doctor_id, patient_id$) and
R42($doctor_id, medicine_id$).
Thus, the 4NF decomposition is:
R1($doctor_id, doctor_name$),
R2($patient_id, patient_name$),
R3($medicine_id, medicine_name$)
R41($doctor_id, patient_id$) and
R42($doctor_id, medicine_id$).

10. Consider the relational schema **R** as:

R(*A, B, C, D, E, F, G, H*), where the domains of all the attributes consist of atomic values. Consider the following FDs for the relation *department*.

- $A \rightarrow D$,
- $D \rightarrow EF$,
- $BH \rightarrow CG$,
- $G \rightarrow H$,

From among the decompositions given, identify the one that is in BCNF.

[MCQ: 2 points]

- ☐ (*A, D, E, F*), (*G, H*), (*B, C, G, H*) and (*A, B, H*)
- ☐ (*D, E, F*), (*A, D*), (*G, H*) and (*B, C, G*)
- ☒ (*D, E, F*), (*A, D*), (*G, H*), (*B, C, G*) and (*A, B, H*)
- ☐ (*D, E, F*), (*A, D*), (*B, C, G, H*) and (*A, B, H*)

Solution: Due to atomic values, the relation **R** is in 1NF.

Candidate key is: *ABH* as $(ABH)^+ = R$.

Test for 2NF: FD: $A \rightarrow D$ and $BH \rightarrow CG$ violate 2NF conditions (these are partial functional dependencies). Thus, the decomposition of **R** is:

Since $(A)^+ = ADEF$, **R1**(*A, D, E, F*), where *A* is the candidate key,

since $(BH)^+ = BHCG$, **R2**(*B, H, C, G*), where *BH* is the candidate key, and

R3(*A, B, H*) consists of the original candidate key of *R*.

Now, **R1**, **R2** and **R3** are in 2NF.

Test for 3NF: In **R1**, FD $D \rightarrow EF$ violates 3NF conditions (as *D* is not a superkey). Thus, the decomposition is:

R11(*D, E, F*), where *D* is the candidate key and **R12**(*A, D*), where *A* is the candidate key.

The relations **R3** and **R2** are already in 3NF (since in **R2**, FD: $G \rightarrow H$ satisfies 3NF conditions as *H* is a prime attribute).

Test for BCNF: The relations **R11**, **R12** and **R3** are already in BCNF. However, in relation **R2**, FD: $G \rightarrow H$ violates BCNF conditions as *G* is not a super key). Thus, the decomposition is:

R22(*G, H*), where *G* is the candidate key and **R22**(*B, C, G*), where *BCG* is the candidate key.

The final relations after decomposition are:

R11(*D, E, F*), **R12**(*A, D*), **R21**(*G, H*), **R22**(*B, C, G*) and **R3**(*A, B, H*). Please note that although the decomposition is lossless, it is not dependency preserving.

11. Which of the following statements is/are true regarding temporal relations?

[MSQ: 2 points]

- ☐ A uni-temporal relation can have only valid time.
- ☐ A uni-temporal relation can have only transaction time.
- ✓ ☒ A uni-temporal relation can have either valid transaction time or transaction time.
- ✓ ☒ A bi-temporal relation can have both valid transaction time and transaction time.

Solution:

- An uni-temporal relations has one axis of time, either valid time or transaction time.
- A bi-temporal relation has both axis of time, valid time and transaction time. It includes valid start time, valid end time, transaction start time, transaction end time.

BSCCS2001: Practice with Solutions
Week 7

1. Select the correct statement(s) from the following options: [MSQ: 2 points]

- ✓ Common Gateway Interface (CGI) is a standard interface between web and database server.
- ☐ The main function of the server side scripting is to correspond within a web-page.
- ✓ URIs can be classified as locators(URLs), or as names (URNs), or both.
- ☐ HTTP can be used to format most of the web documents into hypertext documents.

Solution:

- The Common Gateway Interface (CGI) provides the middleware between WWW servers and external databases and information sources.
- The main function of the server side scripting is to carry out a task at the server's end and then send the result to the client side.
- URNs and URLs are the subsets of the URIs.
- HTML can format most of the web documents into hypertext documents.

2. Identify the three main components of Application Architecture Layer. [MSQ: 2 points]

- ☐ Controller Layer, Data Access Layer, Backend Layer
- ☐ Presentation Layer, Controller Layer, Model Access Layer
- ✓ Presentation Layer, Middle Layer, Backend
- ✓ Presentation Layer, Business Logic Layer, Data Access Layer

Solution: Application layer consists of 3 sub-layers namely:

- Frontend or Presentation Layer
- Middle Layer or Application / Business Logic Layer
- Backend or Data Access Layer

Presentation layer follows model-view-controller (MVC) architecture.

3. Identify A, B and C marked in Figure 1.

[MCQ: 2 points]

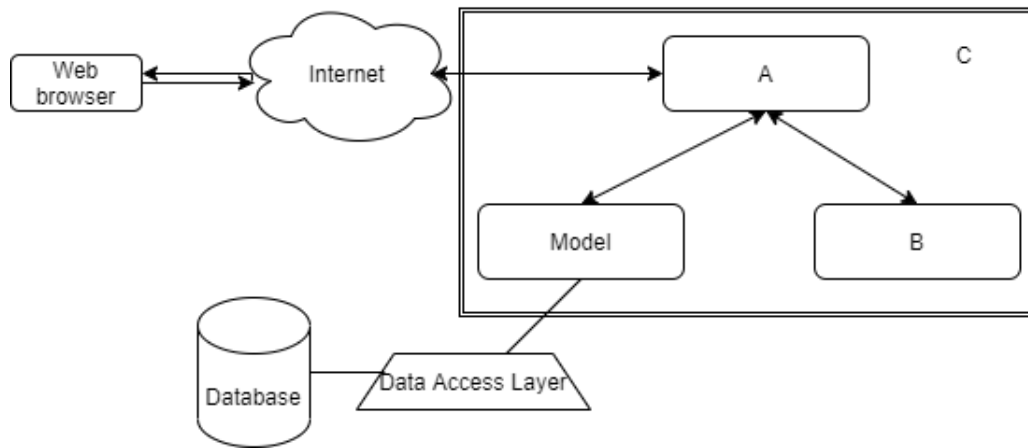


Figure 1: Web/Application Server

- ☐ A - View, B - Controller, C - Business Logic Layer,
- ☐ A - View, B - Controller, C - Presentation Layer
- ☐ A - Controller, B - View, C - Business Logic Layer
- ☒ A - Controller, B - View, C - Presentation Layer

Solution: Presentation Layer is constituted by MVC architecture where M is model, V is view, C is controller.

- The Controller component acts as an interface between Model and View components to process all the business logic and incoming requests, manipulate data using the Model component and interact with the Views to render the final output.
- The Model component corresponds to all the data-related logic that the user works with.
- The View component is used for all the UI logic of the application.

4. Which among the following is a Python library for parsing HTML? [MCQ: 2 points]

- ☐ Requests
- ✓ ☒ Beautiful Soup
- ☐ Paramiko
- ☐ Twisted Python

Solution: Beautiful Soup is an HTML parser that can handle all sorts of HTML. Requests is a powerful HTTP client library. Paramiko is used for implementing the SSH2 protocol. Twisted Python is a framework for asynchronous network programming.

5. Which of the following is/are the disadvantage(s) of using single factor authentication with password? [MSQ: 2 points]

- ✓ The password can be disclosed by guessing or sniffing of packets, if passwords are not encrypted.
- Using single factor authentication can be a time-consuming process, as it involves multiple steps like password plus one-time password sent by SMS.
- ✓ Passwords can be exposed if passwords are reused by a user across sites.
- ✓ Passwords can be captured by the specially designed spyware.

Solution: Please refer to slide: 35.13

6. Match the appropriate statements for

- a. 1-tier architecture
- b. 2-tier architecture
- c. 3-tier architecture
- d. n -tier architecture

Statement-1: It distributes different components of the 3-tiers between different servers and add interface tiers to enable interactions and load balancing.

Statement-2: It keeps all the components of an application on a single server or platform.

Statement-3: It separates its tiers as Presentation, Logical and Data Access.

Statement-4: It is based on client-server architecture, where all the interactions between client and server take place directly, without presence of any intermediate.

[MSQ: 2 points]

- ☐ a)-statement-1, b)-statement-3, c)-statement-2, and d)-statement-4.
- ☐ a)-statement-4, b)-statement-2, c)-statement-3, and d)-statement-1.
- ☐ a)-statement-4, b)-statement-3, c)-statement-2, and d)-statement-1.
- ☒ a)-statement-2, b)-statement-4, c)-statement-3, and d)-statement-1.

Solution: Please refer to slide: 31.17 - 31.20.
--

7. Which of the following tasks is/are performed by a web server?

[MSQ: 2 points]

- ☐ It has the core software component, named as rendering engine, that transforms HTML documents and other resources of a web page into an interactive visual representation on a user's device.
- ☒ It receives HTTP/HTTPS requests, and responds to the requests with the content of that requested resource or an error message.
- ☒ If the requested document is an executable program, it executes the program, and sends back the HTML document that is generated.
- ☐ It is used to access World Wide Web, and it can fetch content from the Web and display it on a user's device.

Solution:

- A web server is software and underlying hardware that accepts requests via HTTP or its secure variant, HTTPS.
- A web browser or crawler, requests for a specific resource using HTTP, and the server responds with the content of that resource or an error message.
- When a web server receives a request for a document which is an executable program, it executes the program, and sends back the HTML document that is generated.

8. Which of the following tasks is/are widely performed by a client-side script?

[MSQ: 2 points]

- ☒ It can check input validity of Web pages to avoid many round trips to server.
- ☐ It can make any system call, and can also access the file system of the host machine to perform read, write operations.
- ☒ It can provide rich user interface.
- ☐ It always executes at server-end and sends the result back to the client-end.

Solution: Client-side scripts are widely used to:

- forms basis of new generation of Web applications (called Web 2.0 applications) offering rich user interfaces
- check input for validity

However, in general, client-side scripts are

- firstly downloaded at the client-end and then interpreted and executed by the browser.
- not allowed to make any system calls directly, and disallowed with dangerous actions such as file writes.

9. Which of the following is not true about ODBC?

[MSQ: 2 points]

- ☐ ODBC is designed with an objective to support various Windows versions. Thus, it is not supported by the non-Windows operating systems.
- ✓ ODBC is a standard API for database connectivity, used by the application programs to communicate with database servers.
- ☐ ODBC is a Java-based technology developed by Sun Microsystems.
- ✓ An application written using ODBC can be easily ported to heterogeneous client and server platforms.

Solution: Please refer to Slide: 33.8

10. Consider the following tasks in a Java program to execute an SQL query at database server using JDBC.

1. Create a "Statement" object
2. Use the "Statement" object to execute the SQL statement
3. Create a "Connection" object
4. Fetch the query results in a "ResultSet" object

Identify the correct order in which the tasks must be performed. [MCQ: 2 points]

- ☐ 1 → 2 → 3 → 4
- ☐ 2 → 1 → 3 →
- ☒ 3 → 2 → 1 → 4
- ☐ 3 → 1 → 2 → 4

Solution: The appropriate order must be:

1. Create a "Connection" object
2. Create a "Statement" object
3. Use the "Statement" object to execute the SQL statement
4. Fetch the query results in a "ResultSet" object

11. Among the given options, which is/are a challenge in Web Application development?
[MSQ: 2 points]

- ☐ The number of rows that have been affected (modified, inserted, or deleted) by the last `execute()` procedure.
- ✓ Knowledge of framework and platforms
- ☐ Limited computing power
- ✓ Web security threats
- ☐ Limited memory

Solution: The challenges for Web application development are:

- User interface and user experience
- Scalability
- Performance
- Knowledge of framework and platforms
- Security

Limited computing power and limited memory are the challenges for Mobile application development.

BSCCS2001: Practice Solutions
Week 8

1. Which of the following does not belong to disk interface standards families?

[MCQ:2points]

- ☐ Serial ATA
- ☐ Small Computer System Interconnect
- ☒ Storage Area Networks
- ☐ Serial Attached SCSI

Solution: Disk interface standards families

- ATA (AT Attachment) range of standards
- SATA (Serial ATA)
- SCSI (Small Computer System Interconnect) range of standards
- SAS (Serial Attached SCSI)
- Several variants of each standard (different speeds and capabilities)

Please refer to slide No. 39.16

Answer questions 2 and 3 on the basis of the following data.

Consider you have a file named “ IITM_BSc ” in your hard disk. The file size is 1000 KB.

Seek time of your hard disk read head is 3ms, rotational speed is 30,000 RPM. The disk has 200 sectors/track and sector size is 512 bytes.

2. What is the transfer rate of your hard-disk (in KB/ms)?

[MCQ:2 points]

- ☒ 50 KB/ms
☐ 100 KB/ms
☐ 33.33 KB/ms
☐ 95 KB/ms

Solution:

Transfer rate is the rate at which data is read from the disk. It can be calculated as follows.

The given rotational speed of the disk = 30,000 RPM.

i.e. disk rotates 30,000 times in 60 sec.

So, time required for making 1 rotation = $60/30,000 = 2$ ms

*Total number of bytes present on one track = number of sectors/track * sector size*

Total number of bytes present on one track = $200 * 512 = 102400$ bytes

Transfer Rate = Bytes on one track / time for making one rotation

Transfer Rate = $102400/2 = 51200$ bytes/ms

Converting into KB/ms = $51200/1024 = 50$ KB/ms.

Hence, option 1 is correct.

3. Considering the fact that the file data is stored in all non-consecutive sectors, how much time will be required to read the whole file after the read request is made?

Note: Consider, Access time + Transfer time

[MCQ:2 points]

- ☐ 10.02 seconds
☒ 8.02 seconds
☐ 0.024 second

○ 0.026 second

Solution:

*Rotational latency = $(1/2) * \text{time for making one rotation}$.*

Therefore, Rotational latency = $(1/2) * 2 = 1 \text{ ms}$

Transfer time = File size / transfer rate

= $1000 * 1024 / 51200 = 20 \text{ ms}$

Since the file data is stored in random sectors (i.e., non-consecutive), hence each sector would require a new seek.

So each sector would have both seek latency and rotational latency.

Seek time + Rotational latency = $3 + 1 = 4 \text{ ms}$.

Number of sectors in which the file is stored, = $1000 * 1024 / 512 = 2000 \text{ sectors}$.

Time required for placing the head on sectors = $4 * 2000 = 8000 \text{ ms}$

Time required = Time required for placing the head on sectors (Access time) + transfer time

Therefore, time elapsed = $8000 + 20 = 8020 \text{ ms}$ or 8.02 seconds.

Hence, option 2 is correct.

4. Consider the following statements,

[MCQ:2 points]

1. DNA data storage is the process of encoding and decoding binary data to and from synthesized strands of DNA.
2. A DNA synthesizer machine builds synthetic DNA strands matching the sequence of digital code
3. Both DNA Digital Storage and Quantum Memory can store enormous data which is not possible in file based storage system.
4. Quantum Memory stores the information in binary states.

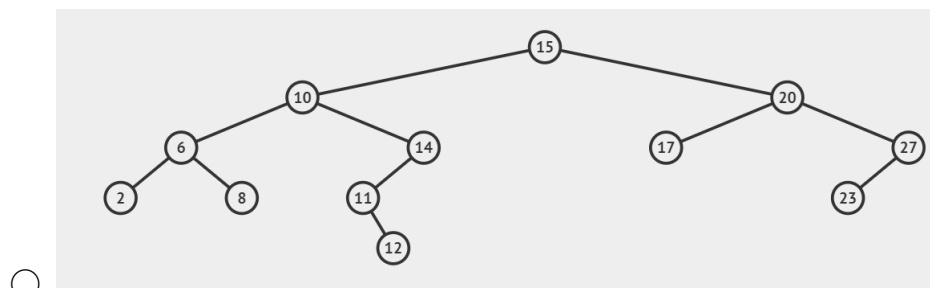
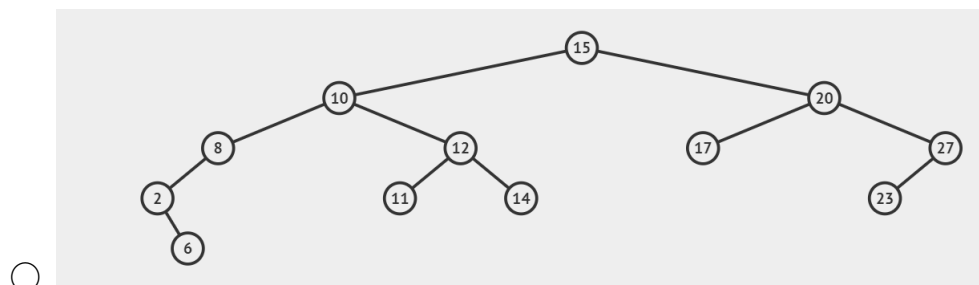
Choose the correct option below

- ☐ Statements 2,3 & 4 are correct
- ☐ Statements 1,3 & 4 are correct
- ☒ Statements 1,2 & 3 are correct
- ☐ All the statements are correct

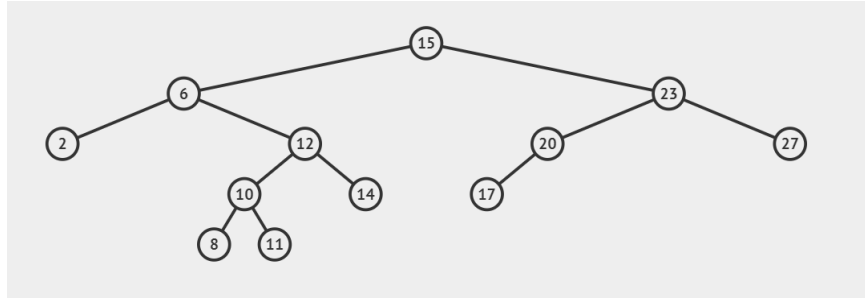
Solution: Please refer to slide no 39.31 and 39.32

5. Choose the correct Binary Search Tree (BST) for the following sequence:
15,10,20,6,12,17,23,2,8,11,14,27

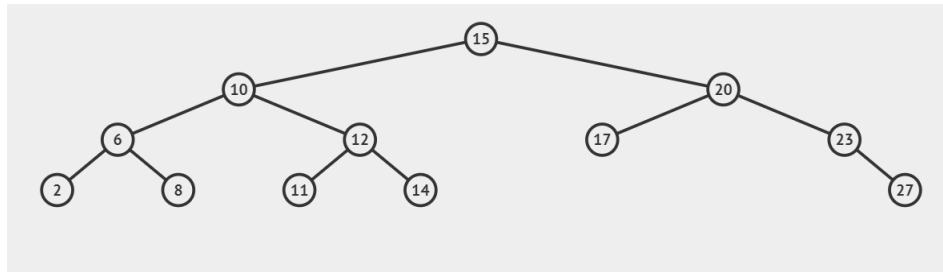
[MCQ:2 points]



☐



✓



Solution:

Option 1: It represents the BST for sequence
15,10,20,8,12,27,23,2,6,11,14,17

Option 2: It represents the BST for sequence
15,10,6,20,27,2,23,17,8,14,11,12

Option 3: It represents the BST for sequence
15,23,6,20,12,2,10,17,8,14,11,27

option 4: Is a correct answer.

6. Which of the following is an example of a volatile storage medium?

[MCQ:2 points]

- ☐ Flash memory
- ✓ ☒ Main memory
- ☐ Hard disk
- ☐ Magnetic tape

Solution:

All the options except main memory are examples of non-volatile storage. Main memory or RAM is a volatile storage medium. Hence, option 2 is correct.

7. Heap file organization is used for storing records of a relation **R**. The cardinality of **R** is 8192. A given selection operation (SELECT query) is such that it fetches only one specific record from **R**. What is the maximum number of search steps/compare operations required to run this SELECT statement?

[MCQ:2 points]

- ☐ 1
- ☒ 8192
- ☐ 13
- ☐ 8193

Solution:

In heap file organization records can be stored in any available free space and no ordering is done in this file organization. Hence a selection operation has to go through all the possible records to find the required record. In worst case the required record can be the last one and so 8192 search steps/comparisons will be required. Hence option 2 is correct.

8. Which of the following statements is/are correct about physical storage media?

[MSQ: 2points]

- ☐ Cache are the non-volatile and most costly form of storage.
- ☒ Flash memory are widely used in embedded devices such as digital cameras, phones, and USB keys.
- ☒ In magnetic-disk, data must be moved from the disk to main memory for access, and written back for storage.
- ☐ Reads and writes are faster in optical disk storage than magnetic disk.

Solution: Please refer to lecture no 8.4

9. Which of the following statements is/ are correct about a Buffer Manager?

[MSQ: 2 points]

- ✓ If the block is already in the buffer, the buffer manager returns the address of the block in the main memory.
- ☐ If the block is in the buffer, the buffer manager allocates space in the buffer for the block.
- ☐ In Buffer manager, the subsystem responsible for allocating buffer space in secondary memory.
- ✓ If the block is not in the buffer, the buffer manager reads the block from the disk to the buffer, and returns the address of the block in main memory to the requester.

Solution:

- If the block is already in the buffer, the buffer manager returns the address of the block in the main memory.
- If the block is not in the buffer, the buffer manager
 - Allocates space in the buffer for the block.
 - Reads the block from the disk to the buffer, and returns the address of the block in the main memory to the requester.
- In Buffer manager, the subsystem responsible for allocating buffer space in **main** memory.

10. Which of the following statements is/ are correct? [MSQ:2points]

- ✓ Disk controller is the interface between the computer system and the disk drive hardware.
- ☐ NOR flash storage is much cheaper than NAND flash storage.
- ✓ USB flash drives are removable and rewritable storage devices.
- ☐ All of the above

Solution:

- Disk controller is the interface between the computer system and the disk drive hardware.
- NAND flash storage is much cheaper than NOR flash storage.
- USB flash drives are removable and rewritable storage devices.

Please refer to Lecture no 8.4

11. Which of the following statements is/are correct? [MSQ:2 points]

- ☐ A Secure Digital (SD) card is a type of removable memory card which is used to read large quantities of data only.
- ✓ SSDs do not include any moving parts unlike HDD.
- ☐ The speed of SSD is lesser than that of HDD as it reads/writes data at lower input/output per second.
- ✓ Cloud storage supports file sharing dynamically as it can be shared anywhere with network access.

Solution:

- A Secure Digital (SD) card is a type of removable memory card used to **read and write** large quantities of data.
- SSDs do not include any moving parts unlike HDD.
- The speed of SSD is **much larger than** that of HDD as it reads/writes data at **higher** input-output per second.
- Cloud storage supports file sharing dynamically as it can be shared anywhere with network access.

Answer questions 12 and 13 based on the given data.

Consider a disk with 10 platters, 64 tracks/surface, 256 sectors/track, 512 bytes/sector. 4 bytes/sector is reserved for storing file system information (formatting data).

12. How much free space is available for use (in MB, upto two decimal places)?

[NAT:2 points]

Answer: 158.75

Solution:

Total number of surfaces = $2 * 10 = 20$

Total number of sectors = $20 * 64 * 256 = 327680$ sectors.

Total space reserved for file system data = $4 * \text{no. of sectors} = 1310720$ bytes.

Converting to MB = $1310720 / 2^{20} = 1.25$ MB

Total disk space = $20 * 64 * 256 * 512 = 167772160$ bytes.

Converting to MB = $167772160 / 2^{20} = 160$ MB

Disk space left for the user = $160 - 1.25 = 158.75$ MB.

Answer is 158.75.

In a 32 GB pen drive, the file's system information, called format overhead, is stored and hence not all of the 32 GB is available for use.

13. How many bits are required for addressing all the sectors ?

[NAT:2 points]

Answer: 19

Solution:

No. of sectors = 327680

No. of bits required to address all of them = $\lceil \log_2 327680 \rceil = 19$ bits.

Answer is 19.

14. Consider a string of pending block references in the given order: 3, 4, 1, 4, 2, 3, 1, 4, 2, 3. The system has a buffer with 3 slots. Assume that initially the buffer is empty. If LRU buffer replacement policy is used, then how many misses will occur while referencing all the requested blocks ?

[NAT:2 points]

Answer: 9

Solution:

Will be explained in practice live session.

This is an example of block reference order, where LRU worsens the working and increases the misses. As mentioned in the slides.

15. Choose the correct arrangement of the given growth functions in increasing order.

[MCQ:2points]

✓ $\log \log N < N \log N < N^2 < 2^N < N^N$

○ $\log \log N < N \log N < N^2 < N^N < 2^N$

○ $N \log N < \log \log N < N^2 < N^N < 2^N$

○ $N \log N < \log \log N < 2^N < N^2 < N^N$

Solution: Refer slide 36.18, 36.19 for the concept and similar example.

<p style="text-align: center;">BSCCS2001: Practice Solutions Week 9</p>

1. Consider a B -tree of order 17.

[MCQ: 2 points]

What are the minimum number of child pointers and keys that can be placed in a non-root node?

- ✓ min. number of child-node pointers = 9,
min. number of keys = 8.
- ☐ min. number of child-node pointers = 2,
min. number of keys = 1.
- ☐ min. number of child-node pointers = 9,
min. number of keys = 9.
- ☐ min. number of child-node pointers = 17,
min. number of keys = 16.

<p>Solution: A non-root node of a B-tree of order $p = 17$ has minimum $\lceil \frac{p}{2} \rceil = \lceil \frac{17}{2} \rceil = 9$ child-node pointers and $\lceil \frac{p-1}{2} \rceil = \lceil \frac{17-1}{2} \rceil = 8$ keys.</p>

2. Consider a non-empty B^+ -tree of order 17.

[MCQ: 2 points]

What are the maximum and minimum number of keys that can be placed in the root node?

- ☐ max. number of keys = 16,
min. number of keys = 8.
- ☐ max. number of keys = 17,
min. number of keys = 8.
- ☒ max. number of keys = 16,
min. number of keys = 1.
- ☐ max. number of keys = 16,
min. number of keys = 2.

Solution: A non-empty B^+ -tree must have a minimum of one key and a maximum of $p - 1$ keys, where p is the order of the B^+ -tree. Thus, in this case the maximum number of keys = 16, and the minimum number of keys = 1.

3. Suppose a data file has 1,00,000 records. Each disk block contains 10 such records (records are unspanned and of fixed-length) and the total space required to store the file is 100 MB (megabytes). We consider a primary (sparse index with an index entry for every block in file) index is constructed on the data file. The key field is 30 bytes and pointer field is 10 bytes for each entry of the index. How many blocks are required to store the index?

[NAT: 2 points]

Answer: 382

Solution: Since the data-file has 1,00,000 records and each disk block contains 10 such records, the total number of blocks required to store the entire data-file is $\frac{100000}{10} = 10000$.

The 10,000 blocks are stored in 100 MB (megabytes). Thus, each disk block size is

$$\frac{100 \times 2^{20}}{10000} = 10485.76 \text{ bytes.}$$

But we must consider the nearest smaller whole number as the usable block size because the records are unspanned.

Therefore, block size = 10485 bytes.

A dense index on the primary key requires one entry for each record in the given relation (thus, 1,00,000 entries) and the size of each entry is:

size-of-key-field + size-of-pointer-field = 30 + 10 = 40 bytes.

So, the space required to store the index is $40 \times 100000 = 4000000$ bytes.

Thus, the number of blocks required to store the index is

$$\left\lceil \frac{4000000}{10485} \right\rceil = 382$$

.

Consider a multilevel index with four levels as L_1, L_2, L_3 and L_4 . Let L_1 be the innermost and L_4 be the outermost levels. Let the index blocking factor or the maximum number of entries held by a block be 50. With the given information, answer the questions 4 and 5.

4. If the number of blocks in level L_1 is 1,00,000, then how many blocks are required at L_2, L_3 and L_4 ?

[MCQ: 2 points]

- ✓ Number of blocks at L_2 is 2000,
Number of blocks at L_3 is 40,
Number of blocks at L_4 is 1
- ☐ Number of blocks at L_2 is 2000,
Number of blocks at L_3 is 50,
Number of blocks at L_4 is 1
- ☐ Number of blocks at L_2 is 2000,
Number of blocks at L_3 is 200,
Number of blocks at L_4 is 4
- ☐ Number of blocks at L_2 is 20000,
Number of blocks at L_3 is 2000,
Number of blocks at L_4 is 10

Solution: Number of blocks in L_2 is $\lceil \frac{100000}{50} \rceil = 2000$.
 Number of blocks in L_3 is $\lceil \frac{2000}{50} \rceil = 40$.
 Number of blocks in L_4 is $\lceil \frac{40}{50} \rceil = 1$.

5. Given the multilevel access structure, the number of block accesses required to read a record is

[MCQ: 2 points]

- ✓ 5
- ☐ 4
- ☐ 3
- ☐ 17

Solution: For each level, there is a block access and access to the block having the target record. Thus, the number of block accesses required is 5.

6. Consider a B-tree based index with order $p = 19$. Assume each node of the B-tree is 73% full. If the height of the tree is 3, then what is the maximum number of keys that can be accommodated in the given B-tree?

[MCQ: 2 points]

✓ 38,415

☐ 8,663

☐ 12,765

☐ 68,605

Solution: On an average, each node will have 0.73×19 or approximately 14 tree pointers. Thus, on an average, each node will have 13 keys.

Level	Nos. of nodes	Nos. of Keys	Nos. of pointers
Level-0 (root)	1	$1 \times 13 = 13$	$1 \times 14 = 14$
Level-1	14	$14 \times 13 = 182$	$14 \times 14 = 196$
Level-2	196	$196 \times 13 = 2,548$	$196 \times 14 = 2,744$
Level-3	2,744	$2,744 \times 13 = 35,672$	

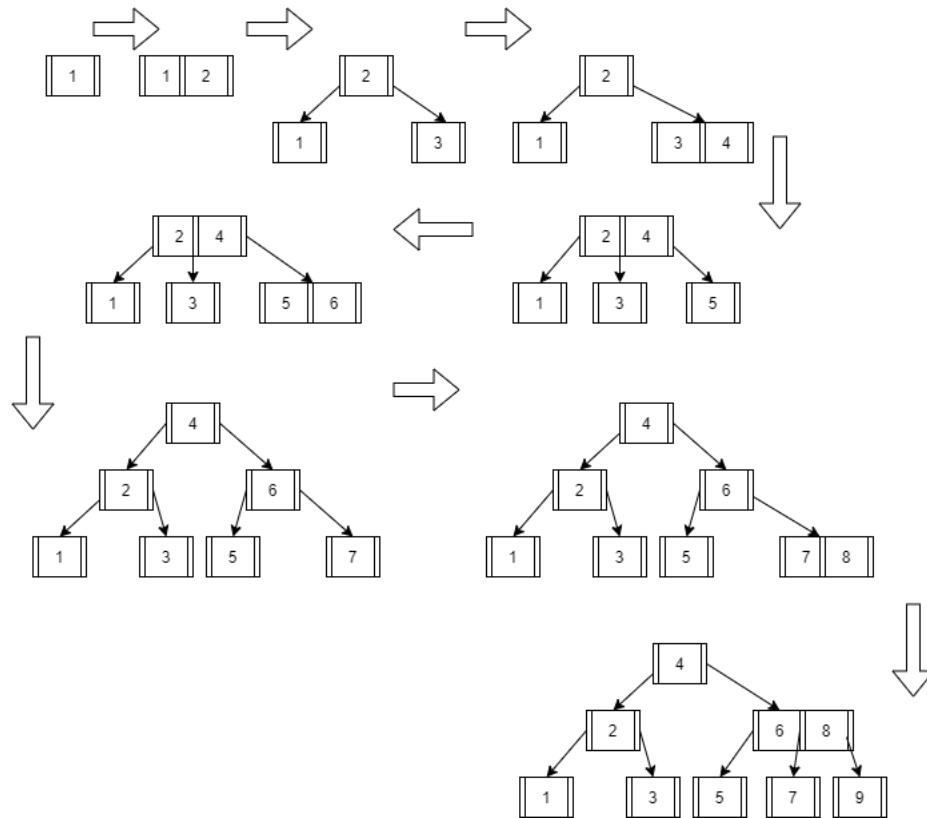
Thus, the number of keys that can be accommodated is $13 + 182 + 2,548 + 35,672 = 38,415$.

7. Consider a B-tree of order 3. If we have 9 elements to be inserted into the tree, then what will be the maximum number of splits that take place in the nodes?

[NAT: 2 points]

Answer: 5

Solution: Assume that the elements are $1 < 2 < 3 < 4 < 5 < 6 < 7 < 8 < 9$. The elements are in ascending order such that the B-tree becomes skewed, and the maximum number of splits take place. The steps for insertion are as shown in the figure, and it may be observed that the number of splits is 5.



8. Consider the instance of a 2-3-4 tree given in Figure 1 and answer the question that follows.

[MCQ: 2 points]

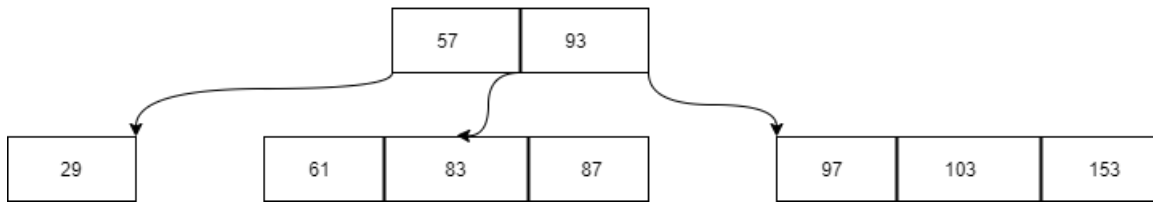


Figure 1: Instance of 2-3-4 tree

If we perform the following operations (in the given order) in the instance given in Figure 1, then choose the correct statement(s) about the resultant 2-3-4 tree.

- Insert value 38
 - Insert value 89
 - Insert value 100
- ☐ Insertion of value 38 will lead to a split.
- ✓ ☒ Insertion of value 100 will lead to a split.
- ✓ ☒ Insertion of value 89 will lead to a split.
- ☐ Insertion of value 100 will not lead to a split.

Solution: When value 38 is inserted, we can add it to the leaf node beside 29 and no split is required. When value 89 is inserted, the block containing values 61, 83, 87 will need to be split. When value 100 is inserted, the block containing values 97, 103, 153 will need to be split. Figure 2 shows the resultant 2-3-4 tree.

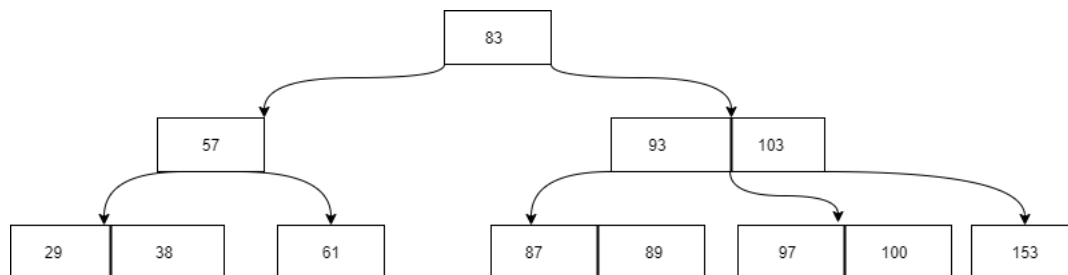


Figure 2: Resultant tree

9. Consider the table **Food**(*Fid*, *Fname*, *Foodtype*, *Rating*). There can only be two types of food - Dessert or Soup. The rating value is an integer in the range 1 to 5.

[MCQ: 2points]

Let the bitmap indexes for columns *Foodtype* and *Rating* be as given below:

Foodtype: 1010 for Dessert, 0101 for Soup

Rating: 0101 for 1, 0010 for 2, 0110 for 3, 0000 for 4, 1000 for 5

In order to find the *Fname* of an item that is a dessert with a rating of 2, which of the following operations will be performed?

- ☐ Logical AND operation on 0101 and 0010
- ☒ Logical AND operation on 1010 and 0010
- ☐ Logical OR operation on 1010 and 0010
- ☐ Logical XOR operation on 1010 and 0010

Solution: Dessert has a bitmap index of 1010 and Rating 2 has a bitmap index of 0010. Hence, in order to find *Fname* of an item that is a dessert and has a rating of 2, we must perform the Logical AND operation on 1010 and 0010.

10. Consider a B^+ tree in which the maximum number of keys allowed in a node is 7. Which among the following options are correct about the given tree?

[MSQ: 2 points]

- ☐ The minimum number of keys in any non-root node is 2.
- ✓ ☒ The B^+ tree has a maximum of 8 child pointers.
- ☐ The B^+ tree has a maximum of 7 child pointers.
- ✓ ☒ The minimum number of keys in any non-root node is 3.

Solution: Let the order of the given B^+ tree be p . Then, the maximum number of keys will be $(p-1)$. Hence the number of keys in the B^+ tree is 8. The minimum number of keys in any non-root node will be $\lceil \frac{p-1}{2} \rceil = 3$.

11. From among the given options, choose the incorrect statement(s) about static and dynamic hashing.

[MSQ: 2 points]

- ☐ Static hashing uses a fixed hash function to partition the set of all possible search-key values into subsets, and then maps each subset to a bucket.
- ✓ ☒ Static hashing is a method of hashing in which a variable number of buckets are allocated to a file to store the records.
- ☐ Dynamic hashing may use a second stage of mapping to determine the bucket associated with some search-key value.
- ☐ In dynamic hashing, memory is well utilized as it grows and shrinks with the data. Hence, there will not be any unused memory.

Solution: Static hashing is a method of hashing in which a fixed number of buckets are allocated to a file to store the records.
The other statements follow from definitions of static and dynamic hashing.

Consider the given table **USER** and answer the questions 12 and 13.

USER(*User_ID*, *User_name*, *User_city*, *User_country*, *User_skill*)

[MCQ: 2 points]

12. Choose the correct SQL statement from the options to create an index with the name 'idx_userid', for the attribute *User_ID*.

- ☐ CREATE INDEX AS idx_userid ON USER (User_ID);
- ☐ CREATE INDEX idx_userid AS USER (User_ID);
- ☐ CREATE INDEX AS idx_userid ON User_ID;
- ☒ CREATE INDEX idx_userid ON USER (User_ID);

Solution: The syntax to create index on a table is:

```
CREATE INDEX index_name ON table_name (indexed_attribute_name);
```

13. Choose the correct SQL statement from the options to remove the index with the name 'idx_userid' on *User_ID*.

- ☐ REMOVE INDEX ON idx_userid;
- ☐ DELETE INDEX ON User_ID;
- ☒ DROP INDEX idx_userid;
- ☐ DROP INDEX OF USER ON idx_userid;

Solution: The syntax to remove an existing index on a table is:

```
DROP INDEX index_name;
```

14. Which of the following statements is/are correct about the B^+ tree data structure that is used for creating an index of a relational database table?

[MSQ: 2 points]

- ✓ B^+ Trees are considered *balanced* because the lengths of the paths from the root to all leaf nodes are equal.
- ☐ Non-leaf nodes have pointers to data records.
- ☐ B^+ Trees are considered *balanced* because the number of records in any two leaf nodes differ by at most 1.
- ✓ Key values in each node are kept in sorted order.

Solution: The statements follow from the definition and structure of a B^+ tree.

BSCCS2001: Practice with Solutions
Week 10

1. Which of the following properties of transactions will be satisfied for a serial schedule always?

[MCQ 2points]

- ☐ Atomicity of all the transactions in the schedule
- ☒ Isolation of all the transactions in the schedule
- ☐ Consistency of all the transactions in the schedule
- ☐ Durability of all the transactions in the schedule

Solution: If a schedule is serial, then it implicitly follows the Isolation property of all its constituting transactions.

Isolation property makes it essential that each of the transaction should be able to operate in such a way that it does not get affected by the execution of another transaction in parallel. In a serial schedule, transactions are executed serially with no interleaving. Hence, Isolation property is trivially satisfied.

In a serial schedule, it is not a guarantee that all transactions in it will be completely executed. Hence, Atomicity is not guaranteed.

A transaction may keep the database inconsistent in a serial schedule. Hence, it is not a guarantee for Consistency

Serial schedules have nothing to do with Durability property.

2. The number of correct statements among the two given statements are

[NAT 2points]

1. All conflict serializable schedules are cascadeless recoverable
2. All strict schedules are conflict serializable

Ans: 0

Solution: Consider the schedule

S1: $r1(A), w1(A), r2(A), T2.commit, w1(B), T1.commit$

This schedule is conflict serializable but not recoverable. Hence, statement 1 is false

Consider the schedule

S2: $r1(X), w2(X), w2(Y), T2.commit, r1(Y)$

This schedule is strict but not conflict serializable.

So both statements are incorrect.

3. What is the number of schedules which are view equivalent to the given schedule **S**?

[NAT 2points]

T1	T2	T3
r(S) w(Q)	w(Q)	r(P) w(Q)

Figure 1: S

Ans: 11

Solution: Two schedules S1 and S2 are said to be view equivalent, if they satisfy all the following conditions:

1. Initial read of all data items must be done by same transactions in both schedules
2. Final write of all data items must be done by same transactions in both schedules
3. If in schedule S1, the transaction T1 is reading a data item updated by T2 and then, in schedule S2. T1 should read the value after the write operation of T2 on same data item

In the given schedule there are total 5 operations- 2 writes and 3 reads. This schedule does not have any instance where the update done by one transaction is read by another, and hence all possible arrangements of these 5 operations will follow the 3rd and 1st conditions implicitly.

But to make the schedules follow 2nd condition, we must ensure that w3(Q) is kept at its place.

Thus, except w3(Q) we have 4 operations in a grouping of (2, 1, 1) for (T1, T2, T3). Hence, the number of possible arrangements following the rules are

$$= \frac{4!}{2!1!1!} = 12$$

Since the given schedule is itself one arrangement, the number of other view equivalent schedules are 11

For the given schedule **S**, answer questions 4 and 5

S: $r5(Z), w1(Y), r2(Y), w3(Y), r4(Y), w2(P), r5(P), w4(X), r1(Q), r5(X), w5(Y)$

4. The schedule **S** is serializable to which of the following serial schedules?[MCQ 2points]

☒ $T3 \rightarrow T4 \rightarrow T1 \rightarrow T2 \rightarrow T5$

☐ $T1 \rightarrow T5 \rightarrow T3 \rightarrow T2 \rightarrow T4$

☐ $T5 \rightarrow T4 \rightarrow T3 \rightarrow T2 \rightarrow T1$

☐ $T3 \rightarrow T2 \rightarrow T5 \rightarrow T1 \rightarrow T4$

Solution: As the read-write relations should be maintained, hence

Transaction T2 must be after T1.

Transaction T4 must be after T3

Last write of attribute Y must be preserved. Therefore, T5 must be the last transaction in a serial schedule.

Thus, the answer is : $T3 \rightarrow T4 \rightarrow T1 \rightarrow T2 \rightarrow T5$

5. Given,

m = Number of serial schedules to which **S** is equivalent

n = Number of serial schedules to which **S** is conflict equivalent.

What is the value of m-n?

[NAT 2points]

Ans: 1

Solution:

$T3 \rightarrow T4 \rightarrow T1 \rightarrow T2 \rightarrow T5$: is a serial schedule equivalent to **S**

$T1 \rightarrow T2 \rightarrow T3 \rightarrow T4 \rightarrow T5$: is a serial schedule conflict equivalent to **S**.

6. Tom is working as a System Designer in a reputed firm. He has 3 transactions to be analyzed as follows. [NAT 2points]

T1: $w1(A), w1(C)$.

T2: $r2(D), w2(E)$.

T3: $w3(B)$.

His analysis involves checking each possible concurrent schedule that can be made using the transactions.

How many schedules does Tom have to check?

Ans: 30

Solution: If a schedule has 3 transactions with a, b, c number of operations in each respectively, then the number of concurrent schedules that can be formed using these transactions = $\frac{(a+b+c)!}{a!.b!.c!}$

Keep in mind that there should not be any reader-writer relation among the operations. Otherwise, all possible arrangements will not hold.

Putting a= 2, b=2, c=1 we get the answer = 30

7. Schedule **S** is as given: $w^2(P)$, $w^1(P)$, $w^3(P)$, $w^2(Q)$, $w^1(Q)$

Consider the statements:

- All Conflict serializable schedules are 2-P lockable
- The given schedule **S** is Conflict serializable
- The given schedule is 2-P lockable

The number of correct statements is

[MCQ 2points]

- ☐ 0
☒ 1
☐ 2
☐ 3

Solution: Let's try to apply 2 phase locking on the schedule:

T1	T2	T3
	X(P)	
	w(P)	
	X(Q)	
	U(P)	
X(P)		
w(P)		
U(P)		
		X(P)
		w(P)
		U(P)
	w(Q)	
	U(Q)	
X(Q)		
w(Q)		

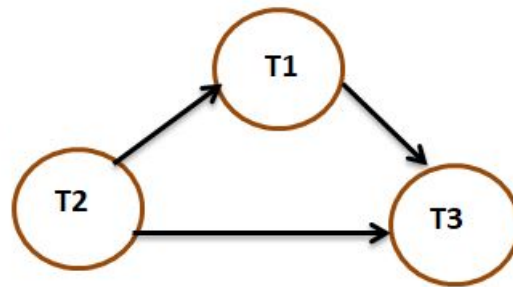
U(P) : Unlock all locks on P operand

$w(P)$: Write P

X(P) : Exclusive lock on operand P

We see that the red colored $X(Q)$ is not allowed according to the rule of 2 phase locking protocol, since unlocking phase of $T1$ has started already because $U(P)$ was done previously by $T1$. So this schedule is not 2 phase lockable.

The precedence graph of the given schedule is :



We observe that there exists no cycle in this graph and hence the schedule is Conflict serializable.

Hence, only the second statement is true.
The number of correct statements is 1.

8. Schedule **S** is as given: $w2(A), r1(A), r3(A), w1(A), w3(A), r2(B), w1(C)$
If the timestamps for transactions T1, T2, T3 are 20, 5, 10 respectively, then choose all the correct options.

[MSQ 2points]

- ☒ Timestamp ordering protocol does not allow the transaction.
- ☐ Timestamp ordering with Thomas Write rule allows the transaction.
- ☒ Neither Timestamp ordering nor Thomas Write rule allows the transaction
- ☐ None of the above

Solution: When T3 tries to perform $w3(A)$, it checks what is the read and write timestamp of A.

T1 has both read and written A previously in the schedule. Therefore, both read and write timestamp of operand A will be set to 20, as a result of which $w3(A)$ will not be performed since $T3.timestamp(10) < A.writetimestamp(20)$.

So T3 is rolled back by Timestamp ordering protocol.

Even Thomas write rule will not allow T3 to get executed because Thomas write rule ignores a late write only if the write timestamp of the operand is greater. In the given schedule, both read and write timestamps of operand A are greater than the timestamp value of T3.

So neither of the two protocols allow this schedule.

So correct choices are option 1 and 3.

9. Schedule **S** is as given: $w_2(A)$, $r_3(A)$, $w_1(A)$, $w_3(A)$, $r_2(B)$, $w_1(C)$

If the timestamps for transactions T1, T2, T3 are 20, 5, 10 respectively, then choose the correct options from the following.

[MSQ 2points]

- ☒ Timestamp ordering protocol does not allow the transaction
- ☒ Timestamp ordering with Thomas Write rule allows the transaction
- ☐ Neither Timestamp ordering nor Thomas Write rule allows the transaction
- ☐ None of the above

Solution: When T3 tries to perform $w_3(A)$, it checks what is the read and write timestamp of A.

As T1 has the greatest timestamp value and it has till now only wrote operand A and never did read it, so write timestamp of A will be 20 and read timestamp of A will be 10 at this point of time.

Timestamp ordering protocol will not allow the transaction since

$writetimestampof A(20) > timestampof T3$

but Thomas write rule will allow the transaction since

$readtimestampof A(10) = timestampof T3$.

Thomas write rule ignores the write timestamp.

So, option 1 and 2 are correct choices.

10. In the *wait-die* scheme for deadlock prevention, starvation can be avoided to a large extent by: [MCQ: 2 points]

- ☐ non pre-emptive rollback of younger transactions
- ☒ restarting rolled back transactions with original timestamp
- ☐ pre-emptive rollback of older transactions
- ☐ detecting deadlocks using precedence graphs

Solution: When both younger and older transactions are restarted, they both start with same timestamp and hence the older transactions lose the advantage of their seniority. There is a possibility that even in the next rollback step, the same transaction gets rolled back again. This can lead to starvation. When the timestamp of transactions are retained during rollback, the older transactions get precedence over younger ones, and hence the likelihood of starvation is reduced, though not completely avoided.

11. Choose the correct statement(s) about the lock-based protocols.

Statement I: A transaction will never be granted a lock on an item, even if the requested lock is compatible with locks already held on the item by other transactions.

Statement II: Any number of transactions can hold shared locks on an item, unless any transaction holds an exclusive on the item.

Statement III: If a lock cannot be granted, the requesting transaction is made to wait until all incompatible locks held by other transactions have been released.

[MCQ: 2 points]

☐ I & II

☒ II & III

☐ I & III

☐ I, II, & III

Solution: Statements II & III are correct by the definitions of shared and exclusive locks. In the case of Statements I, the correct statement would be - a transaction may be granted a lock on an item if the requested lock is compatible with locks already held on the item by other transactions.

12. Choose the correct output obtained on running the given SQL statements on Table **Employee**. [MCQ: 2 points]

EID	ENAME
E01	Arthur
E02	Raina
E03	Meena
E04	Arthur
E06	Joey

Table **Employee**

```
SQL> SAVEPOINT SP1;
SQL> UPDATE Employee SET ENAME='Jainie'
      WHERE EID='E06';
SQL> SAVEPOINT SP2;
SQL> DELETE FROM Employee WHERE EID='E02';
SQL> SAVEPOINT SP3;
SQL> UPDATE Employee SET ENAME='Raina'
      WHERE EID='E04';
SQL> ROLLBACK TO SP1;
```

☐

EID	ENAME
E01	Arthur
E02	Raina
E03	Meena
E04	Arthur
E06	Jainie

☐

EID	ENAME
E01	Arthur
E03	Meena
E04	Arthur
E06	Jainie

☐

EID	ENAME
E01	Arthur
E03	Meena
E04	Raina
E06	Jainie

☒

EID	ENAME
E01	Arthur
E02	Raina
E03	Meena
E04	Arthur
E06	Joey

Solution: Since savepoint SP1 is the first savepoint added before any of the DML statements in the list are executed, a rollback to SP1 will undo all modifications (since no commit statements have been executed).

13. Given below are four statements. Match each of them with the corresponding property in the set of ACID properties.

Statement 1: Any data written to the database must be valid according to all the defined rules like the check and key constraints and triggers.

Statement 2: Every completed transaction is saved into the secondary storage.

Statement 3: During money transfer, either the amount debited from the source account must be credited to the destination account or the money should not be debited from the source account at all.

Statement 4: If multiple transactions are being executed concurrently, then the final result should be the same immaterial of the sequence in which the transactions were executed.

Let A denote Atomicity, C denote Consistency, I denote Isolation and D denote Durability. From among the given options, find the correct match.

[MCQ: 2 points]

☐ 1 - A, 2 - C, 3 - I, 4 - D

☒ 1 - C, 2 - D, 3 - A, 4 - I

☐ 1 - D, 2 - I, 3 - C, 4 - A

☐ 1 - I, 2 - A, 3 - D, 4 - C

Solution: The statements follow from definition of each property in the set of ACID properties.

<p>BSCCS2001: Practice Assignment with Solutions</p> <p>Week 11</p>

{Write general instructions here}

1. Which of the following statements is/are true?

[MSQ:2 points]

- ☐ If a crash/rollback occurs before the operation completes, then the logical undo is performed and the physical undo information for the operation is ignored.
- ☐ If a crash/rollback occurs after the operation completes, then the physical undo information is used to undo the operation.
- ✓ If a crash/rollback occurs after the operation completes, then the operation-end log record is found.
- ✓ If a crash/rollback occurs before operation completes, then the operation-end log record is not found.

Solution:

- If crash/rollback occurs before the operation completes, then the physical undo information is used to undo the operation.
- If crash/rollback occurs after the operation completes, then the logical undo is performed and the physical undo information for the operation is ignored.
- If crash/rollback occurs after the operation completes, then the operation-end log record is found.
- If crash/rollback occurs before the operation completes, then the operation-end log record is not found.

2. Which of the following statements is/are true regarding checkpoints in the transaction?
[MSQ: 2 points]

- ☒ Any transactions committed before the last checkpoint should be ignored.
- ☐ Any transactions committed since the last checkpoint should be ignored.
- ☒ It scans backwards from the end of the log to find the most recent *< checkpointL >* record.
- ☐ Any transaction that was running at the time of failure needs to be redone.

Solution:

- It scans backwards from end of log to find the most recent *< checkpointL >* record.
- Any transactions that committed before the last checkpoint should be ignored.
- Any transactions that committed since the last checkpoint need to be redone.
- Any transaction that was running at the time of failure needs to be undone and restarted.

So, options 1 and 3 are correct.

3. Consider the following log records of transactions, where immediate database modification scheme is used. [MSQ: 2 points]

step	Details of log
1	$\langle T_0, start \rangle$
2	$\langle T_0, A, 1200, 900 \rangle$
3	$\langle T_0, B, 1000, 800 \rangle$
4	$\langle T_1, start \rangle$
5	$\langle T_1, D, 200, 50 \rangle$
6	$\langle T_0, commit \rangle$
7	$\langle T_2, start \rangle$
8	$\langle T_2, P, 700, 300 \rangle$
9	$\langle T_2, Q, 1150, 670 \rangle$
10	$\langle checkpointL \rangle$
11	$\langle T_1, E, 400, 320 \rangle$
12	$\langle T_1, commit \rangle$
13	$\langle T_2, R, 300, 100 \rangle$

Suppose the transactions failed after step 13, then which of the following is/are correct?

- ☐ Transaction T_1 needs no recovery
- ✓ Transaction T_2 needs to be undone
- ☐ Transaction T_0 and T_1 need no recovery
- ✓ Transaction T_1 needs to be redone

Solution:

Any transactions that committed before the last checkpoint should be ignored. So T_0 can be ignored.

Any transactions that committed since the last checkpoint need to be redone. So, T_1 need to be redone.

Any transaction that was running at the time of failure needs to be undone and restarted. So, T_2 need to be undone and restarted.

4. Consider the following statements:

[MCQ:2Points]

1. Hot backup refers to keeping a database up and running while the backup is being performed concurrently.
2. Hot backup is mainly used for Transaction Log Backup.
3. Transactional Logging is used in circumstances where a possibly inconsistent backup is taken.

Choose the correct option.

- ☐ Statements 1 & 2 are correct
- ☐ Statements 1 & 3 are correct
- ☒ All the statements are correct
- ☐ Only Statement 1 is correct

Solution: Refer to slide No. 51.25

5. Figure 1 shows the timeline of four transactions T1, T2, T3 and T4 respectively.

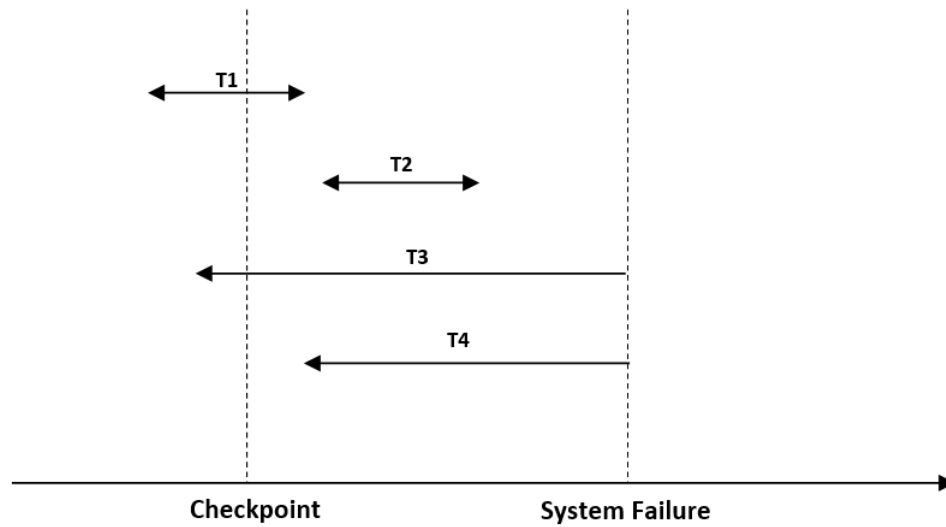


Figure 1: State of transactions

Which of the following action(s) will be taken by the recovery manager? [MSQ: 2 points]

- ☐ Transaction T1 is ignored.
- ☐ Transaction undo is performed for T2.
- ✓ Transaction redo is performed for T1 and T2.
- ✓ Transaction undo is performed for T3 and T4.

Solution:

1. Transaction T1 starts before the checkpoint and commits before system crash, so T1 needs to be redone.
2. Transaction T2 starts after the checkpoint and commits before system crash, so T2 needs to be redone.

6. Consider the following log involving three transactions T1, T2, T3: -

Beginning of log

<T1 start>
<T1, A, 10>
<T1 commit>
<checkpoint>
<T2 start>
< T2, B, 15>
<T2 commit>
<T3 start>
<T3 C, 45>

End of log at crash!

Which of the following will happen after the crash occurs? [MSQ: 2 points]

- ✓ T1 is committed before the checkpoint, it will be ignored.
- ☐ T2 is committed but after the checkpoint, so it should be ignored.
- ☐ T3 did not commit, so redo all its updates.
- ✓ T2 is committed but after the checkpoint, so redo all its updates.

Solution:

- Transaction T1 started before the checkpoint and committed before the checkpoint as well, so T1 will be ignored.
- Transaction T2 started after the last checkpoint and committed before the system crash, so the changes made to T2 needs to be redone.
- Since T3 did not commit before the system failure, so any changes made to T3 needs to be undone.

7. Consider the following statements:-

- Since B+ tree insertions and deletions release lock early, they can be restored by physical undo logging.
- Redoing the previous updates should be logged physically as they do not conflict with the early lock release.

Choose the correct option.

[MCQ: 2 points]

- ☐ Statement 1 is correct & Statement 2 is wrong.
- ☐ Both the statements are correct.
- ✓ ☒ Statement 1 is wrong & Statement 2 is correct.
- ☐ All the statements are wrong.

Solution: Please refer to slide 54.8 & 54.9.

8. What happens if a crash occurs after the operation in the logging process is completed?
[MSQ: 2 points]

- ☐ Physical undo operation is used to undo the operation.
- ✓ ☒ A logical undo operation is performed using the operation identifier.
- ✓ ☒ Physical undo operation is ignored.
- ☐ None of the above

Solution: Please refer to slide 54.12.

9. Consider the following statements.

[NAT: points]

1. It is mandatory to perform a full backup at least once.
2. Recovery from a full backup is the easiest.
3. There is very little to no dependency between consecutive full backups.

How many among the given statements are true regarding Full Backup?

Ans: 3

Solution: Statement 1 is true because we must have one full backup initially.
Statement 2 is true because, while recovering directly from a full backup one consolidated read would restore everything in the database.
It's the least complex recovery process.
Statement 3 is true. Since full backup backs up everything there will not be any dependency on multiple versions of backup.

10. Consider the given backup schedule, and answer the question.

Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
					<i>Full</i>	

Figure 2: .

You can have at most one differential backup in a week and all other backups shall be incremental (except the one full backup given).

On which day will you do the differential backup so as to minimise the number of backup sets required for recovery on any arbitrary day?

[MCQ: points]

- ☐ Friday
- ☐ Sunday
- ☒ Wednesday
- ☐ Number of sets required for recovery is independent of the day of differential backup.

Solution: If we keep the differential backup on Wednesday then irrespective of the day of failure we will not require more than 4 backup sets, which is the minimum considering all other scenarios.

Answer questions 11 and 12 on the basis of the given data.

A RAID-5 storage system with similar arrangement of parity blocks as described in slide 55.16 is used for storing the following data:

DISK-1	DISK-2	DISK-3	DISK-4	DISK-5	
0100	XXXX	0100	0001	0101	Block 1 row
0101	XXXX	0100	0100	0001	Block 2 row

Figure 3: RAID-5 data

11. According to the figure disk-2 has crashed. What data is present in the two blocks of disk-2?

Note: Assume block size is 4 bits

[2 points:MCQ]

- ☒ block 1: 0100, block 2: 0100
- ☐ block 1: 0101, block 2: 0100
- ☐ block 1: 0001, block 2: 0100
- ☐ block 1: 0001, block 2: 0001

Solution:

RAID-5 uses XOR recovery facility for lost data. So the data stored in a block of disk-2 can be obtained by performing bitwise XOR of all blocks of other disks in the same row.

So block 1 of disk-2 stores = 0100 **XOR** 0100 **XOR** 0001 **XOR** 0101 = 0100

block 2 of disk-2 stores = 0101 **XOR** 0100 **XOR** 0100 **XOR** 0001 = 0100

Hence option 1 is correct.

12. Assume that the binary values represent 8 bit ASCII code. What is the data word present inside this RAID-5 storage system?

[2 points:MCQ]

- ☐ IITM
- ☐ IITB
- ☒ DATA
- ☐ DBMS

Solution:

ASCII is 8 bit code so we will group every successive 8 bits from the RAID-5 storage, except the parity bits.

Disk-5 block-1, Disk-4 block-2 are parity blocks hence these will not be added into the data part.

So the data is :01000100 (68 ='D'), 01000001 (65 ='A'),01010100(84 ='T'), 01000001 (65 ='A').

So the word is 'DATA'. Option 3 is correct.

13. For the given set of statements below:

- RAID-1 can allow parallel read of two different disk blocks always
- RAID-4 cannot allow parallel read of different disk blocks always
- RAID-5 allows parallel processing of multiple write requests
- In a system where small write operations occur frequently RAID-5 is better than RAID-1

How many statement(s) is/are FALSE ?

[2 points:NAT]

Answer: 1

Solution:

statement 1: is true. Since data in RAID-1 is stored in two redundant copies, at any time it can allow two different reads of disk blocks.

statement 2: is true. Since RAID4 uses block interleaved parity, a single data instance may be distributed among all the disk's blocks. When a read of such a data is going on it will operate on all the disks at the same time and hence no disk would be free to service a different read request.

statement 3: is true, since RAID5 does not use a reserved disk for parity. Hence, all disks can separately maintain their own parity information. Two or more simultaneous write operation on different disks can work because there is no bottleneck for updating one parity disk for an ongoing write, like in case of RAID 3, 4.

statement 4: is false. A system where the frequency of small write is higher RAID 5 is not a good choice. As every write in RAID 5 requires read-modify-write cycle which is time consuming and therefore performing frequent writes is bad for RAID5. RAID 1 is a better choice in this scenario.

14. Consider the following statements regarding RAID 3.

[MCQ:2points]

1. RAID 3 consists of byte-level striping with dedicated parity.
2. RAID 3 uses multiple designated drives for parity.
3. RAID 3 cannot service multiple requests simultaneously.

Choose the correct option.

- ☐ Statement 1 & 2 are correct
- ☐ Only statement 1 is correct
- ☒ Only statements 1 & 3 are correct

☐ All the statements are correct

Solution:

- RAID 3 consists of byte-level striping with dedicated parity.
- RAID 3 has a single check disk with parity information.
- RAID-3 cannot service multiple requests simultaneously: This is so because any single block of data will be spread across all members of the set and will reside in the same physical location on each disk. Thus, every single I/O request has to be addressed by working on every disk in the array.