Assignment-3

CSA0563
DBMS

K. Thanuja
192373012
27107/2004

1. ER diagram design for Traffic flow management system (TFMS)

Entities and Attributes (Task 1)

a) Road
Road Id
Road Name
Length
Speed limit

b) Intersection
IntersectionID
Intersection name
Latitude
longitude

Trafficsignal
signalID
signal status
Timer

d) Traffic Data
Traffic data ID
Timestamp
Speed
congestion level

3) Road - Traffic data
2) Many to many
Intersection (Fk)
Traffic (Fk)

Road - Traffic data
(one - to - many)
Road (Fk)
Traffic (Fk)

Relationships Task 2
1. Road - Intersection (many - to - many)
Road ID (Fk)
Intersection_ID (Fk)

Normalisation conserderation
1. first normal form (1NF) ensure that all attributes contain atomic values.
2. second Normal form (2NF) ensure that all attributes are fully - functionally
3. Third Normal form (3NF) ensure that all attributes are on depent only primary key.

Justification Task (4.2)
1. Scalability: The design supports the addition of new roads, intersections, traffic signals and data, records without structural changes.
2. Real - Time data Processing
The traffic data entity is designed to accomidate high frequency updates with time tampsed records the one to many relationships b/w Road and traffic

2. Question 1 TOP 3 departments with highest Average salary

SQL QUERY

```
SELECT
    DepartmentID,
    DepartmentName,
    AVG(salary) AS AvgSalary
from
    employees
left join
departments on employes.idepartment
ID = Depaotments.Department ID
Group by
Department ID, Depaotment Name
ORDERBY
    Avg salary4 DESC
LIMIT 3;
```

Question 2: Retrieving hierarchical category paths

SQL QUERY

```
with recursive : category hirarchy
as (select.
    category ID,
    category Name,
CAST (category Name as varchar
(Max)) AS Path
FROM
    Category
Where
    Parent category ID is null
UNION ALL
  SELECT
    C. category ID,
    C. category Name,
  CAST ( ch. Path + '>' + C. category Name
AS VARCHAR (MAX) AS Path
From
    category ID,
    category Name,
    Path
ORDERD BY,
    Path;
```

Question 3: Total distinct customers by month

SQL Query

```
with months AS C
SELECT DATE - formate (DATE - ADD
('CURDATE()', interval - 1 *
n(MONTH)), '%. Y -%. m') AS month year
FROM
(SELECT@ row := @row+1 A S n FROM (SELECT
1 UNION ALL SELECT 2 UNION 2SELECT4)
)
SELECT
  m. monthyear AS MONITHName,
count (DISTINCT O. customer ID) AS
customercant
From
  months m.
```

```
LEF JOIN
Orders O ON DATE - Format (O order
date \y. y - %m') = m. year
GROUP BY
    m. month year
ORDERD BY
    m. month - year;
```

Question 4: Finding closest location

SQL QUERY

```
SELECT
    LOCATION ID,
    location Name,
    latitude
    longitude
    (L3 71 * ACOSC
  COS(RADIANS(@latitude))* COS(RAI
ANS (latitude))*
  COS(RADIANS (longitude) - RADIANS(@
longitude)) + SIN(RADIANS(@ latitude))
*SIN(RADIANS (latitude)) AS Distance
FROM:
    locations
ORDER BY
    Distance
LIMIT 5;
```

Question 5: optimizing query for orders

Table

SQL QUERY

```
SELECT
    *
WHERE
Orderdate >= curdate() - Interval 7 Day
ORDER BY
    Orderdate DESC;
```

ERD Diagram (Task 3)