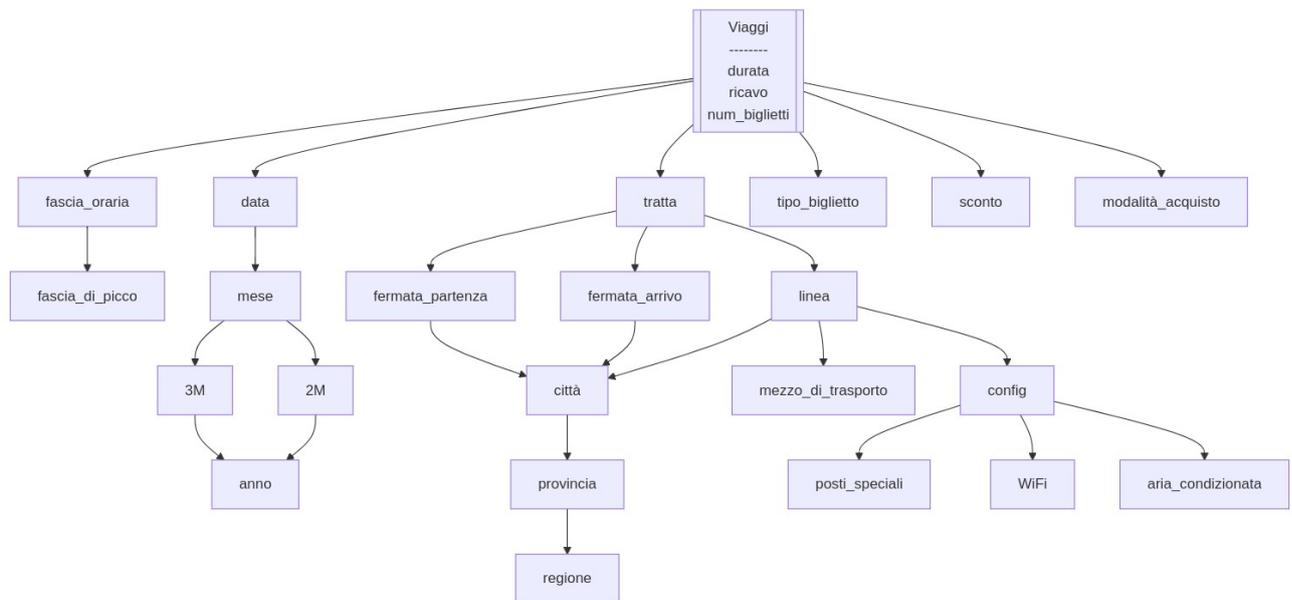


Homework 1

Progettazione Concettuale



Schema Logico

VIAGGI(DURATA, RICA VO, NUM_BIGLIETTI, **ID_FASCIA**, **ID_DATA**, **ID_TRATTA**, **ID_JUNK**)

FASCIA_ORARIA(**ID_FASCIA**, FASCIA_ORARIA, FASCIA_DI_PICCO)

DATA(**ID_DATA**, DATA, MESE, 2M, 3M, ANNO)

TRATTA(**ID_TRATTA**, TRATTA, FERMATA_PARTENZA, FERMATA_ARRIVO, CITTÀ, PROVINCIA, REGIONE, LINEA, MEZZO_DI_TRASPORTO, POSTI_SPECIALI, WIFI, ARIA_CONDIZIONATA)

JUNK_BIGLIETTO(**ID_JUNK**, TIPO_BIGLIETTO, MODALITÀ_ACQUISTO, SCONTO)

NB: le fermate di partenza e di arrivo appartengono alla stessa città perché stiamo modellando trasporti urbani.

Query

```
SELECT mezzo_di_trasporto, mese,
       SUM(num_biglietti) / COUNT(DISTINCT data),
       SUM(SUM(num_biglietti)) OVER (
           PARTITION BY mezzo_di_trasporto, anno
           ORDER BY mese
           ROWS UNBOUNDED PRECEDING)
       100 * SUM(num_biglietti) / SUM(SUM(num_biglietti)) OVER (PARTITION BY mese)
FROM VIAGGI V, TRATTA T, DATA D
WHERE V.ID_TRATTA=T.ID_TRATTA AND V.ID_DATA=D.ID_DATA
GROUP BY mezzo_di_trasporto, mese, anno
```

```
SELECT linea, città
       SUM(durata) / SUM(num_biglietti),
       SUM(SUM(ricavo)) OVER (PARTITION BY città),
       100 * SUM(ricavo) / SUM(SUM(ricavo)) OVER (PARTITION BY città),
       RANK() OVER (PARTITION BY città ORDER BY SUM(ricavo) DESC)
FROM VIAGGI V, TRATTA T, DATA D
WHERE V.ID_TRATTA=T.ID_TRATTA AND V.ID_DATA=D.ID_DATA
AND ANNO >= 2022
GROUP BY linea, città
```

Viste Materializzate

1)

SELECT: num_biglietti
GB: MEZZO_DI_TRASPORTO, MESE

2)

SELECT: num_biglietti
GB: MEZZO_DI_TRASPORTO, MESE, ANNO

3)

SELECT: num_biglietti, ricavo
GB: MEZZO_DI_TRASPORTO, MESE

4)

SELECT: num_biglietti, ricavo
FILTER: ANNO = 2024
GB: MEZZO_DI_TRASPORTO, MESE

5)

SELECT: num_biglietti, ricavo
GB: MEZZO_DI_TRASPORTO, MESE

```
1. CREATE MATERIALIZED VIEW MV1
2. BUILD IMMEDIATE
3. REFRESH FAST ON COMMIT
4. AS
5. SELECT mezzo_di_trasporto, mese, anno,
       SUM(num_biglietti) AS TotTickets,
       SUM(ricavo) AS TotRevenue,
6. FROM VIAGGI V, TRATTA T, DATA D
7. WHERE V.ID_TRATTA=T.ID_TRATTA AND V.ID_DATA=D.ID_DATA
8. GROUP BY mezzo_di_trasporto, mese, anno
```

```

1. CREATE MATERIALIZED VIEW LOG ON TRATTA
2. WITH ROWID, SEQUENCE (ID_TRATTA, mezzo_di_trasporto)
3. INCLUDING NEW VALUES;

1. CREATE MATERIALIZED VIEW LOG ON DATA
2. WITH ROWID, SEQUENCE (ID_DATA, mese, anno)
3. INCLUDING NEW VALUES;

1. CREATE MATERIALIZED VIEW LOG ON VIAGGI
2. WITH ROWID, SEQUENCE (ID_DATA, ID_TRATTA, num_biglietti, ricavo)
3. INCLUDING NEW VALUES;

```

Trigger

```

1. CREATE TABLE VM1 (
    mezzo_di_trasporto VARCHAR (50) CHECK (mezzo_di_trasporto IS NOT NULL),
    MESE, ANNO DATE CHECK (MESE, ANNO IS NOT NULL),
    num_biglietti INTEGER,
    Ricavo DOUBLE PRECISION)

```

```

1. INSERT INTO VM1 (mezzo_di_trasporto, MESE, ANNO, num_biglietti, ricavo)
    (SELECT mezzo_di_trasporto, MESE, ANNO, SUM(num_biglietti), SUM(ricavo)
    FROM TRATTA TR, DATA DT, VIAGGI V
    WHERE DT.ID_DATA = V.ID_DATA AND V.ID_TRATTA = TR.ID_TRATTA
    GROUP BY mezzo_di_trasporto, MESE, ANNO );

```

```

1. CREATE OR REPLACE TRIGGER Trigger1
2. AFTER INSERT ON VIAGGI
3. FOR EACH ROW

4. DECLARE
5. VarTranspMode VARCHAR (50);
6. VarMonth, VarYear DATE;
7. N INTEGER ;

8. BEGIN
9. SELECT mezzo_di_trasporto INTO VarTranspMode
10. FROM TRATTA
11. WHERE ID_TRATTA = :NEW.ID_TRATTA;

12. SELECT MESE, ANNO INTO VarMonth, VarYear
13. FROM DATA
14. WHERE ID_DATA = :NEW.ID_DATA;

15. SELECT COUNT (*) INTO N
16. FROM VM1
17. WHERE mezzo_di_trasporto = VarTranspMode AND MESE = VarMonth;

18. IF (N > 0) THEN
19. UPDATE VM1
20. SET num_biglietti = num_biglietti + :NEW.num_biglietti, ricavo = ricavo + :NEW.ricavo
23. WHERE mezzo_di_trasporto = VarTranspMode AND MESE = VarMonth
24. ELSE
25. INSERT INTO VM1 (mezzo_di_trasporto, MESE, ANNO, num_biglietti, ricavo)
26. VALUES (VarTranspMode, VarMonth, VarYear, :NEW.num_biglietti, :NEW.ricavo);
28. END IF;

29. END;

```