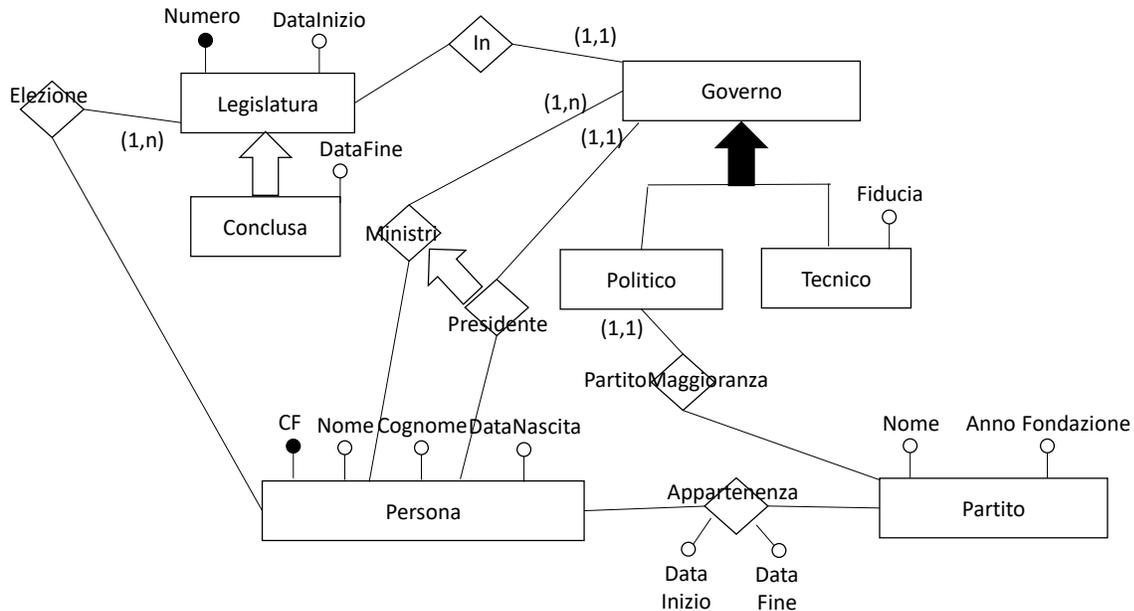


Appello di Basi di Dati – 17/10/2019

Prof. Tiziana Catarci

A.A. 2018-2019

DOMANDA 1 (10 punti)



Una persona non può appartenere a due partiti in intervalli che si sovrappongono

Per i governi politici il partito di maggioranza deve coincidere con il partito che esprime il maggior numero di ministri

Data inizio e data fine sulla relazione appartenenza e di una legislatura conclusa devono essere consistenti

DOMANDA 2

Si lascia allo studente la ristrutturazione dello schema consistente nell'eliminazione di ISA (su entità e relazione) e generalizzazione, e nell'aggiunta di due identificatori alle entità Governo e Partito.

Persona(CF, Nome, Cognome, DataNascita)

Legislatura(Numero, DataInizio)

INC: Legislatura[Numero] \subseteq Elezione[Legislatura]

Elezione(Legislatura, Persona)

FK: Eletto[Legislatura] \subseteq Legislatura[Numero]

FK: Eletto[Persona] \subseteq Persona[CF]

Conclusa(Legislatura, DataFine)

FK: Conclusa[Legislatura] \subseteq Legislatura[Numero]

Governo(ID, Legislatura, Presidente)

FK: Governo[Legislatura] \subseteq Legislatura[Numero]

FK: Governo[Presidente] \subseteq Persona[CF]

INC: Governo[ID] \subseteq Ministri[Governo]

Governo[ID,Presidente] \subseteq Ministri[Governo,Persona]

Ministri(Governo, Persona)

FK: Ministri[Governo] \subseteq Governo[ID]

FK: Ministri[Persona] \subseteq Persona[CF]

Partito(ID, Nome, AnnoFondazione)

Appartenenza(Partito, Persona, DataInizio, DataFine)

FK: Militanza[Partito] \subseteq Partito[ID]

FK: Militanza[Persona] \subseteq Persona[CF]

Tecnico(Governo, Fiducia)

FK: Tecnico[Governo] \subseteq Governo[ID]

Politico(Governo, Partito)

FK: Politico[Governo] \subseteq Governo[ID]

FK: Politico[Partito] \subseteq Partito[ID]

Politico[Governo] \cup Tecnico[Governo] = Governo[ID]

Politico[Governo] \cap Tecnico[Governo] = \emptyset

$\forall x, y, di1, df1: \langle x, y, di1, df1 \rangle$

\in Appartenenza[Persona, Partito, DataInizio, DataFine]:

$\exists z, di2, df2: \langle x, z, di2, df2 \rangle$

\in Appartenenza[Persona, Partito, DataInizio, DataFine] \wedge (df1 > di2)

$\forall x, y, di1, df1: \langle x, y, di1, df1 \rangle$

\in Appartenenza[Persona, Partito, DataInizio, DataFine]: df1 > di1

$$\forall x, p, di: \langle x, p \rangle \in \text{Politico}[\text{Governo}, \text{Partito}] \wedge \langle z, di \rangle \in \text{Governo}[\text{ID}, \text{DataInizio}]: p = \text{argmax}(|\{per, di2, df2: \langle x, per \rangle \in \text{Ministri}[\text{Governo}, \text{Persona}] \wedge \langle per, x, di2, df2 \rangle \in \text{Appartenenza}[\text{Persona}, \text{Partito}, \text{DataInizio}, \text{DataFine}] \wedge di \geq di2 \wedge di \leq df2 \}|)$$

DOMANDA 3

Dato il seguente schema relazionale:

Citta (Codice, NumAbitanti, Nome, Provincia, Regione)

Monumento (Codice, Citta, Artista, Nome, AnnoCostruzione)

Artista (Codice, Nome, AnnoNascita)

si esprimano in SQL le seguenti interrogazioni:

1) Restituire nome, provincia e regione della città con il maggior numero di monumenti (**3 Punti**)

```
CREATE VIEW MonumentiPerCitta AS
SELECT Citta, count(*) as Conteggio
FROM Monumento
GROUP BY Citta
```

```
SELECT c.Nome, c.Provincia, c.Regione
FROM Citta c, MonumentiPerCitta m
WHERE c.Codice = m.Citta AND m.Conteggio = (SELECT
MAX(Conteggio) FROM MonumentiPerCitta)
```

2) Restituire il nome degli artisti che non hanno mai costruito monumenti in città del Piemonte o della Lombardia con più di 30000 abitanti (**4 Punti**)

```
SELECT Nome FROM Artista WHERE Codice NOT IN (SELECT m.Artista
FROM Monumento m, Citta c WHERE m.Citta = c.Codice AND
(c.Regione = 'Piemonte' OR c.Regione = 'Lombardia') and
c.NumAbitanti > 30000)
```

3) Restituire per ogni artista, il numero di monumenti creati prima di avere 30 anni (**4 Punti**)

```
SELECT m.Artista, count(*) FROM Monumento m, Artista a
WHERE m.Artista = a.Codice AND m.AnnoCostruzione - m.AnnoNascita
< 30 GROUP BY m.Artista
```