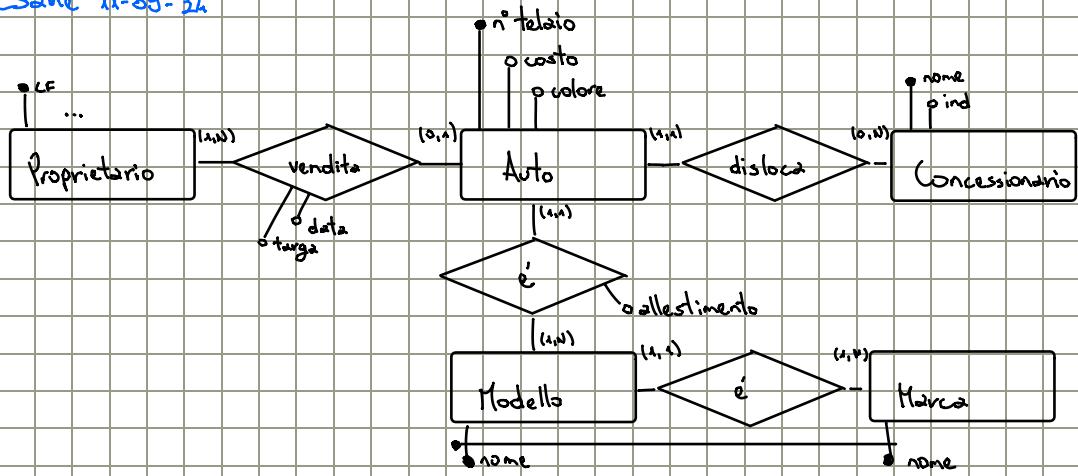


Esame 11-09-24



Auto (#telai, costo, colore, nomeC, nomeMod, nomeMarca)

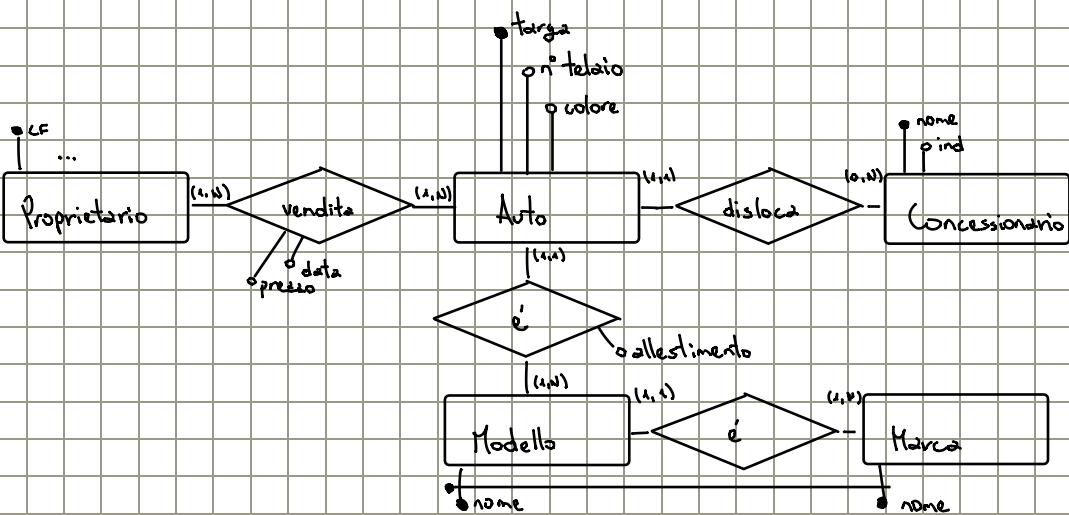
Concessionario (nome, indirizzo, cap, nomeResponsabile)

Modello (nomeMod, nomeMarca)

Marca (nomeMarca)

Proprietario (CF, nome, cognome)

Vendita (CF, #telai, targa, data)



Auto (#telai, targa, colore, nomeC, nomeMod, nomeMarca)

Concessionario (nome, indirizzo, cap, nomeResponsabile)

Modello (nomeMod, nomeMarca)

Marca (nomeMarca)

Proprietario (CF, nome, cognome)

Vendita (CF, #telai, costo, data)

2) SELECT COUNT(A.#telgio) AS NumeroVend, C.CP
 FROM Auto A
 JOIN Concessionario C ON A.nomeC = C.nomeC
 JOIN Vendita V ON A.#telgio = V.#telgio
 GROUP BY C.CP

b) SELECT A.#telgio, P.CF, V.dataVendita
 FROM Auto A
 JOIN Vendita V ON A.#telgio = V.#telgio
 JOIN Proprietario P ON V.CF = P.CF
 WHERE V.dataVendita = (SELECT MAX(Ve.dataVendita)
 FROM Vendita Ve
 WHERE A.#telgio = Ve.#telgio)

c) SELECT A.nomeMod, A.nomeMarca
 FROM Auto A
 JOIN Vendita V ON A.#telgio = V.#telgio
 GROUP BY A.nomeMod AND A.nomeMarca
 HAVING COUNT(V.#telgio) = (SELECT MAX(numVendite)
 FROM (SELECT COUNT(Ve.#telgio) AS numVendite
 FROM Vendita Ve
 JOIN Auto Au ON Au.#telgio = Ve.#telgio
 GROUP BY Au.nomeMod AND Au.nomeMarca))

chiavi: CP, CM, CO, cittM

CP → NomeP, Prezzo

CM, cittM → NomeM

CP, CM, cittM → Giacenza

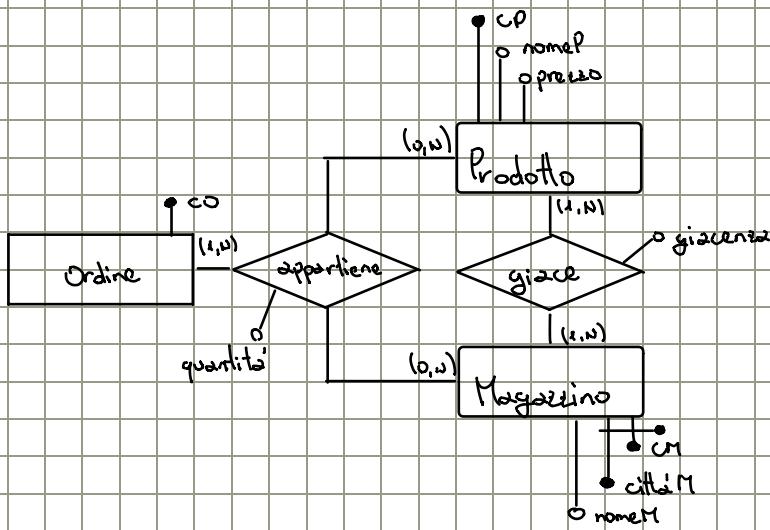
CO, CM, cittM, CP → QuantitàO

Prodotto(CP, NomeP, Prezzo)

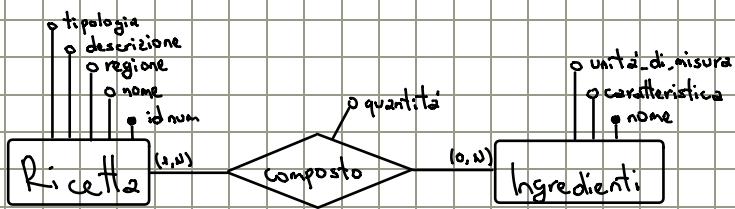
Magazzino(CM, cittàM, NomeM)

Giacenza(CP, CM, cittàM, Giacenza)

Ordine(O, CM, cittàM, CP, QuantitàO)



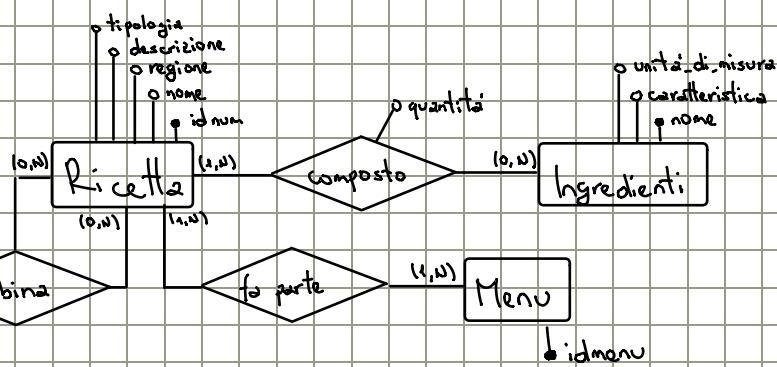
Esercizio 24 - 06 - 24



Ricetta (idnum, tipologia, descrizione, regione, nome)

Ingredienti (nome, caratteristica, unità di misura)

Composto (idnum, nome, caratteristica, quantità)



Ricetta (idnum, tipologia, descrizione, regione, nome)

Ingredienti (nome, caratteristica, unità di misura)

Composto (idnum, nome, caratteristica, quantità)

Menu (idmenu)

Abbinata (idnum, idnum)

Appartiene (idnum, idmenu)

a) SELECT C.idnum, C.nome, C.quantità, I.unita

FROM Composto C

JOIN Ingredienti I on C.nome = I.nome

b) SELECT DISTINCT A.idmenu

FROM Appartiene A

WHERE A.idmenu NOT EXISTS (SELECT A2.idmenu

FROM Appartiene A2

JOIN Composto C ON A2.idnum = C.idnum

JOIN Ingredienti I ON I.nome = C.nome
 WHERE I.caratteristica <> "vegano"

c) SELECT DISTINCT A.idmenu

FROM Appartiene A

WHERE A.idmenu NOT EXISTS (SELECT A2.idmenu

FROM Appartiene A2

JOIN Ricetta R ON A2.idnum = R.idnum

WHERE R.regione <> "Lazio")

$\prod_{R.idnum} (\exists_{R.tipologia = "secondi"} \wedge C.nome = "uova" (Ricetta \bowtie R.idnum = C.idnum \text{ Composto}))$

chiave: id_A, Cod-Marca, Cod-Categoria

id_A → Nome_A, Piano, Settore

Cod-Categoria → Categoria

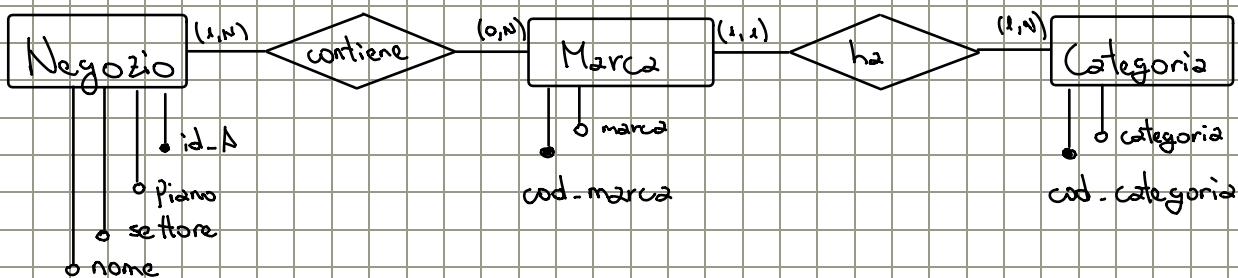
Cod-Marca → Marca, Cod-Categoria

Negozi(id_A, Nome_A, Piano, Settore)

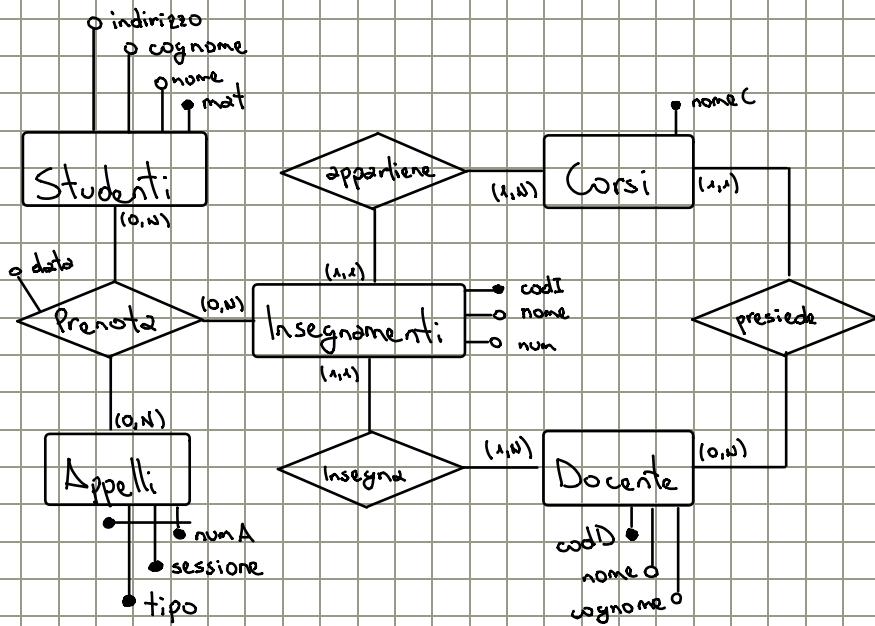
Categoria(Cod-Categoria, Categoria)

Marca(Cod-Marca, Marca, Cod-Categoria)

Contiene(id_A, Cod-marca)



Esame 25-09-24



Appelli (numA, Sessione, Tipo)

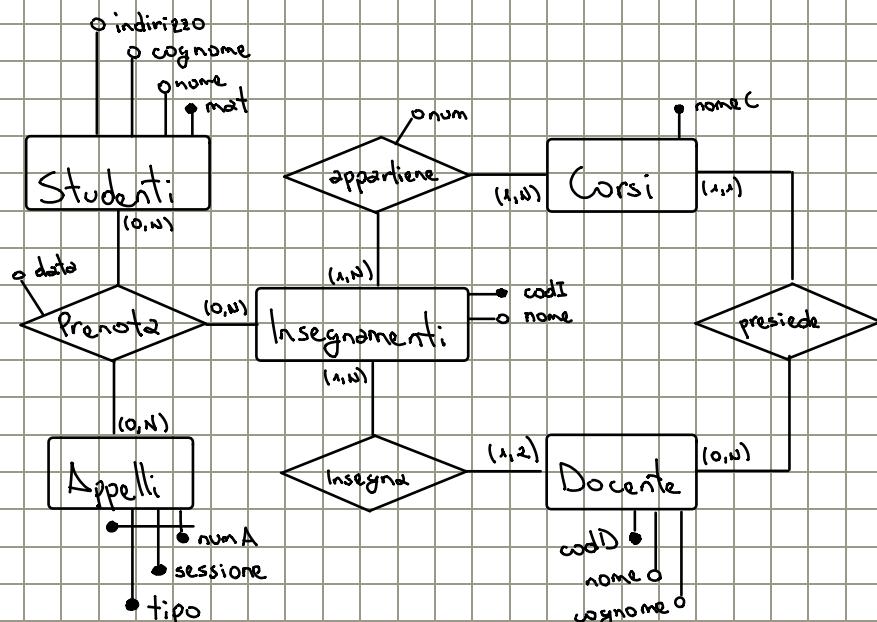
Studenti (Matricola, Nome, Cognome, Indirizzo)

Insegnamenti (codI, nome, num, codD, nomeC)

Corsi (nomeC, codD)

Docente (codD, nome, cognome)

Prenota (numA, Sessione, Tipo, Mat, codI, data)



Appelli (numA, Sessione, Tipo)

Studenti (Matricola, Nome, Cognome, Indirizzo)

Insegnamenti (codI, nome)

Corsi (nomeC, codD)

Docente (codD, nome, cognome)

Prenota | numA, Sessione, Tipo, Mat, codI, data|

Appartiene (nomeC, codI, num)

Insegna (codD, codI)

a) SELECT D.cogn, COUNT (P.mat)

FROM Prenota P

JOIN Insegna I ON P.codI = I.codI

JOIN Docenti D ON D.codD = I.codD

GROUP BY D.cogn

b) SELECT S.nome

FROM Studenti S

JOIN Prenota P ON S.mat = P.mat

WHERE P.numA, P.Sessione, P.Tipo, P.codI = (SELECT P2.numA, P2.Sessione, P2.Tipo, P2.codI
FROM Prenota P2

WHERE P2.mat = "00001")

c) SELECT P.mat, P.nomA, P.Tipo, P.Sessione

FROM Prenota P

JOIN Insegna I ON P.codI = I.codI

ORDER BY P.nomA, P.Tipo, P.Sessione

$\Pi_{I.codD \wedge A.nomC \wedge A.num} (\Delta_{A.codI = I.codI} \text{Insegna})$

chiavi: Cl, CdL, MacroArea, Esame

Cl \rightarrow Insegnamento, Crediti

CdL, Macroarea \rightarrow NomeCorsoL

CI, CdL, Macroarea → Iscritti

CE, Macroarea, CI → MediaVoti

CE, Macroarea → CI

Insegnamento (CE, Macroarea, Insegnamento, Crediti)

Corso (CdL, Macroarea, nome)

Iscritti (CE, CdL, Macroarea, num)

Medie (CE, Macroarea, media)

Codici (CE, Macroarea, CI)

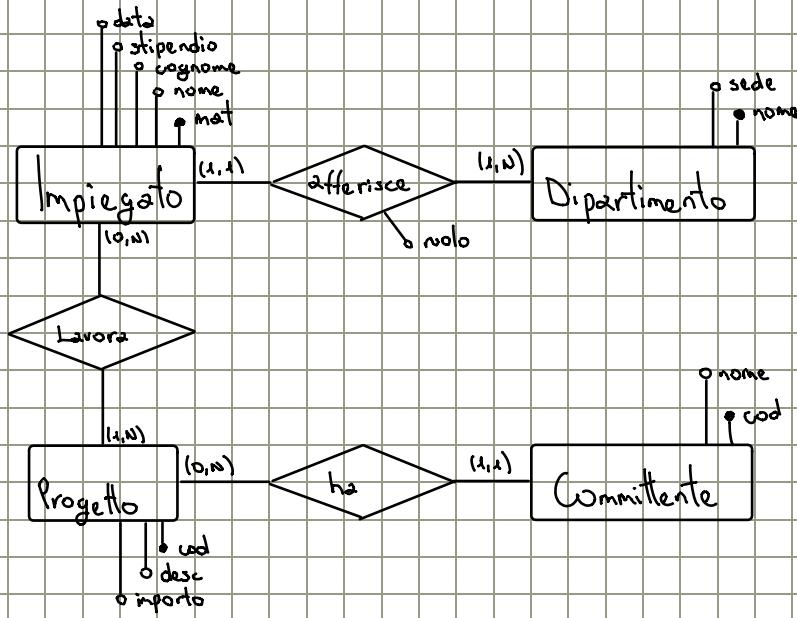
Esame Macroarea (CE, Macroarea, CI, Insegnamento, Crediti, media)

Corso (CdL, Macroarea, nome)

Iscrizione (CE, CdL, Macroarea, num)

Macroarea (Macroarea)

Esame 15-07-2024



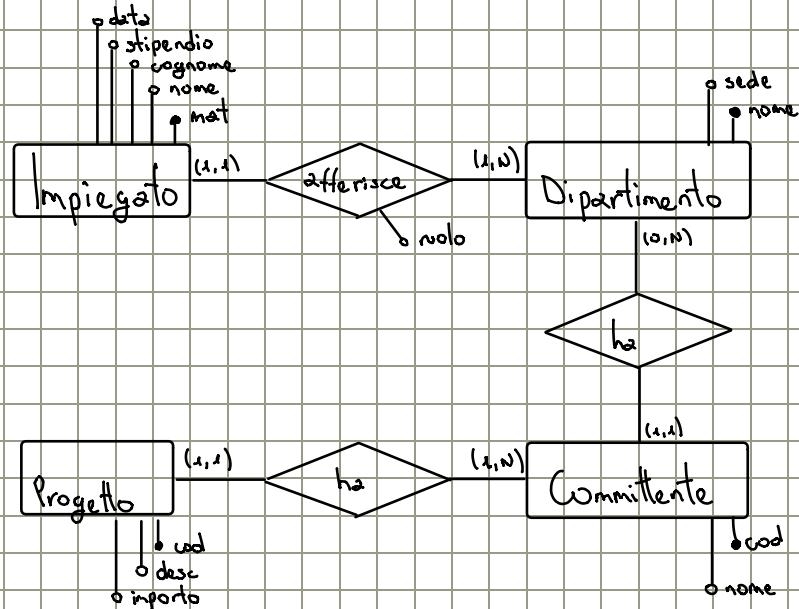
Committeente (codC, nome, codP)

Progetto (codP, desc, importo)

Dipartimento (nome, sede)

Impiegato (mat, nome, cogn, stipendio, data, nomeD, ruolo)

Lavora (mat, codP)



Committeente (codC, nome, nomeD)

Progetto (codP, desc, importo, codC)

Dipartimento (nomeD, sede)

Impiegato (mat, nome, cogn, stipendio, data, nomeD, ruolo)

a) `SELECT C.codC, SUM(P.importo) AS ImportoTotale
FROM Committente C
JOIN Progetto P ON C.codC = P.codC
GROUP BY C.codC`

b) `SELECT D.nomeD, COUNT(P.codP) AS Conta
FROM Dipartimento D
JOIN Committente C ON C.codD = D.codD
JOIN Progetto P ON C.codC = P.codC
WHERE D.nomeD, Conta = (SELECT D2.nomeD, MAX(COUNT(P2.codP))
FROM Dipartimento D2
JOIN Committente C2 ON C2.codD = D2.codD
JOIN Progetto P2 ON C2.codC = P2.codC
GROUP BY D2.nomeD)
GROUP BY D.nomeD`

c) `SELECT sum(P.importo), I.ruolo, D.nomeD
FROM Dipartimento D
JOIN Committente C ON C.codD = D.codD
JOIN Progetto P ON C.codC = P.codC
JOIN Impiegato I ON D.nomeD = I.nomeI
WHERE I.ruolo = 'direttore'
GROUP BY D.nomeD`

chiavi: cod, cc, CN, CM, CA

cod, NomeCittà → Titolo

cc, CN → NomeCittà

CN → Nazione

CA → Area

CM → Materia

cod, NomeCittà → CM → CA

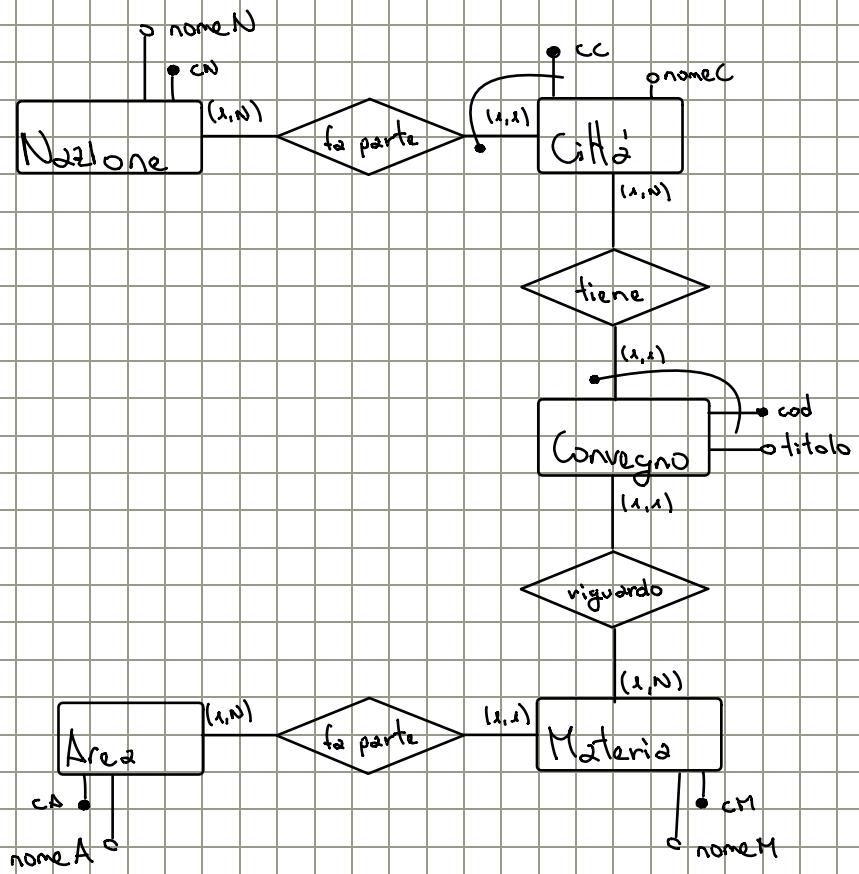
Materia (cM, nomeM, cA)

Area (cA, nomeA) ×

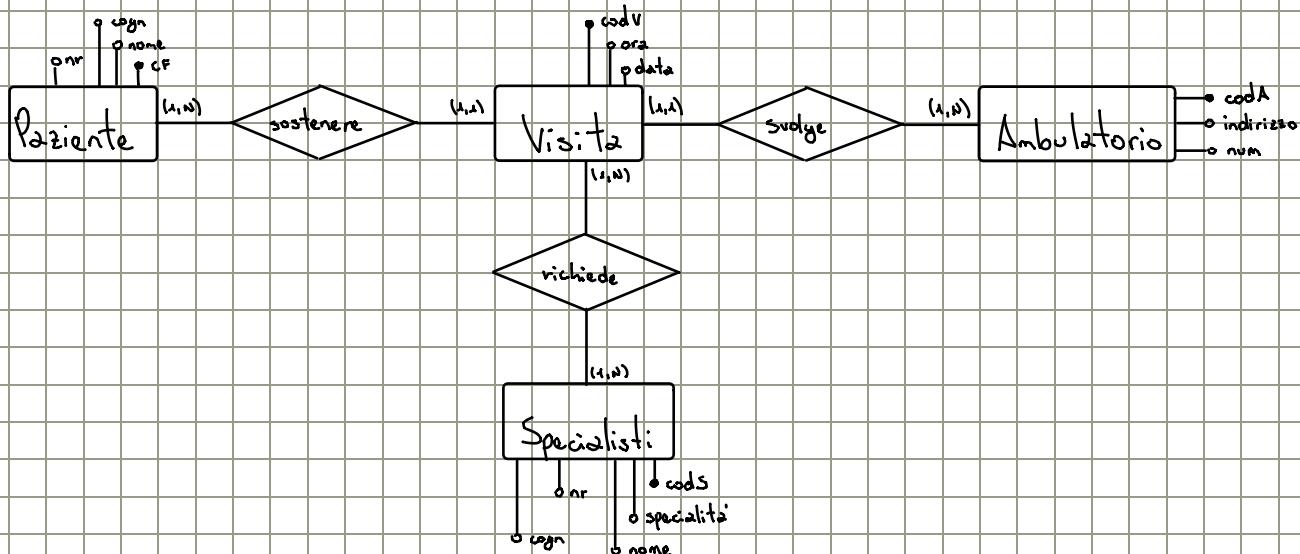
Nazione (cN, nomeN) ×

Città (cc, cn, nomeC) ×

Convegno (cod, cc, cn, titolo, cr)



Esame 29-01-25



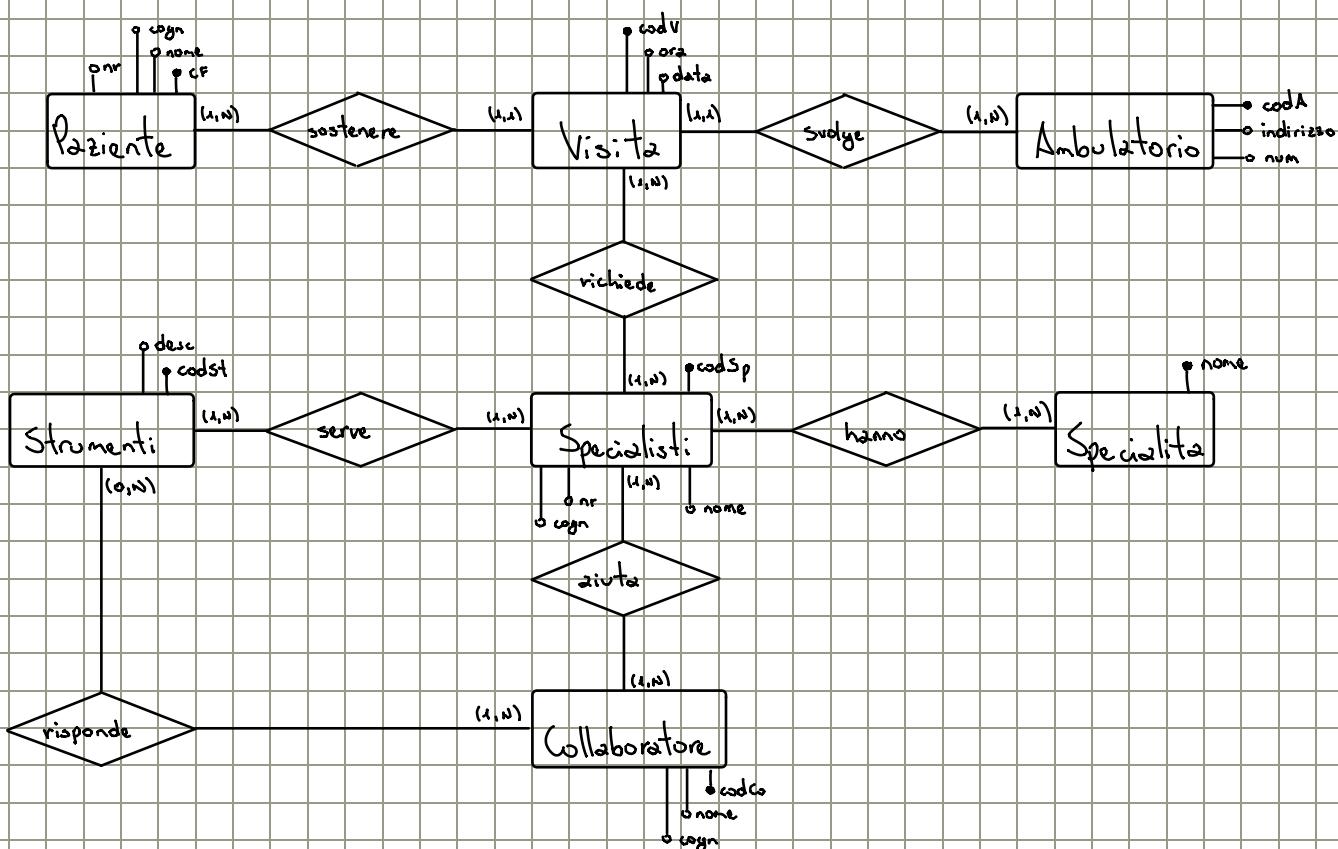
Ambulatorio (codA, ind, nr)

Paziente (cf, nome, cogn, nr)

Visita (codV, ora, data, cf [FK], codA [FK])

Specialisti (codS, nome, cogn, specialità, nr)

Richiede (codV [FK], codS [FK])



Ambulatorio (codA, ind, nr)

Richiede (codV [FK], codS [FK])

Paziente (cf, nome, cogn, nr)

Visita (codV, ora, data, cf [FK], codR [FK])

Specialista (codSp, nome, cogn, nr)

Specialità (nomeS)

Collaboratore (codCo, nome, cogn)

Strumenti (codSt, desc)

Aiuta (codSp [FK], codCo [FK])

Risponde (codSt [FK], codCo [FK])

Hanno (codSp [FK], nomeS [FK])

Serve (codSp [FK], codSt [FK])

a) SELECT V.data, R.codSp, V.ora, V.cf

FROM Visita V

JOIN Richiede R ON V.codV = R.codR

WHERE V.data = CURRENT_DATE

ORDER BY R.codSp, V.ora

b) WITH conteggio AS (

SELECT codSt, codSp, COUNT(*) AS cont

FROM Serve

GROUP BY codSp, codSt)

SELECT codSt, codSp, COUNT(*) AS cont

FROM Serve

WHERE codSt, codSp, cont = (SELECT codSt, codSp, MAX(cont)
FROM conteggio)

c) SELECT V.cf, V.codV, V.data

FROM Visita V

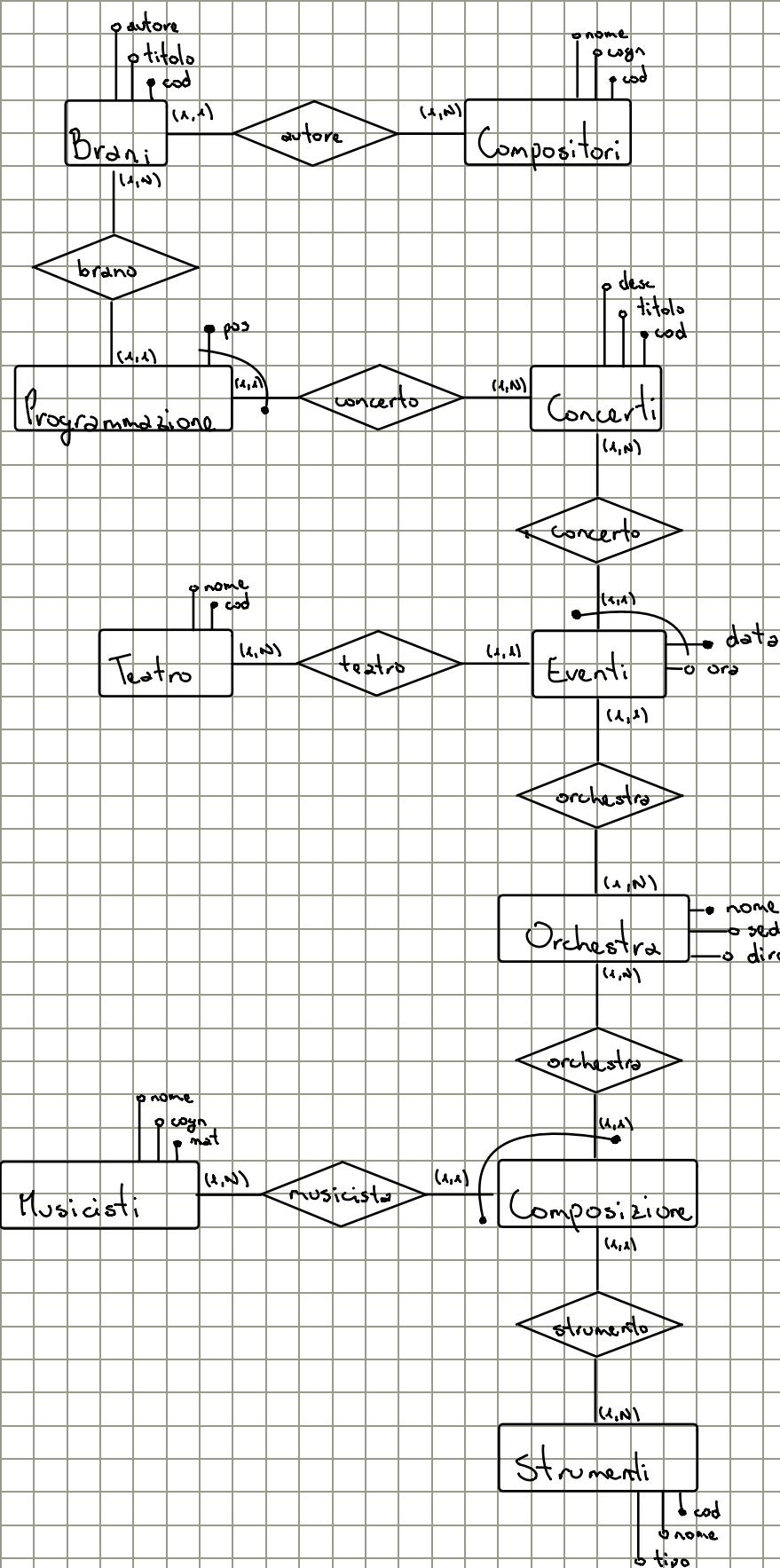
WHERE V.data >= CURRENT_DATE - INTERVAL '1 YEAR'

ORDER BY V.cf, V.data

dl Π
cf, conv, DATA | $\sigma_{\text{DATA} \geq \text{CURRENT_DATE} - \text{INTERVAL 'A YEAR'}} (Visits)$

1. lNF:

Esame ROMATRE 8-07-24

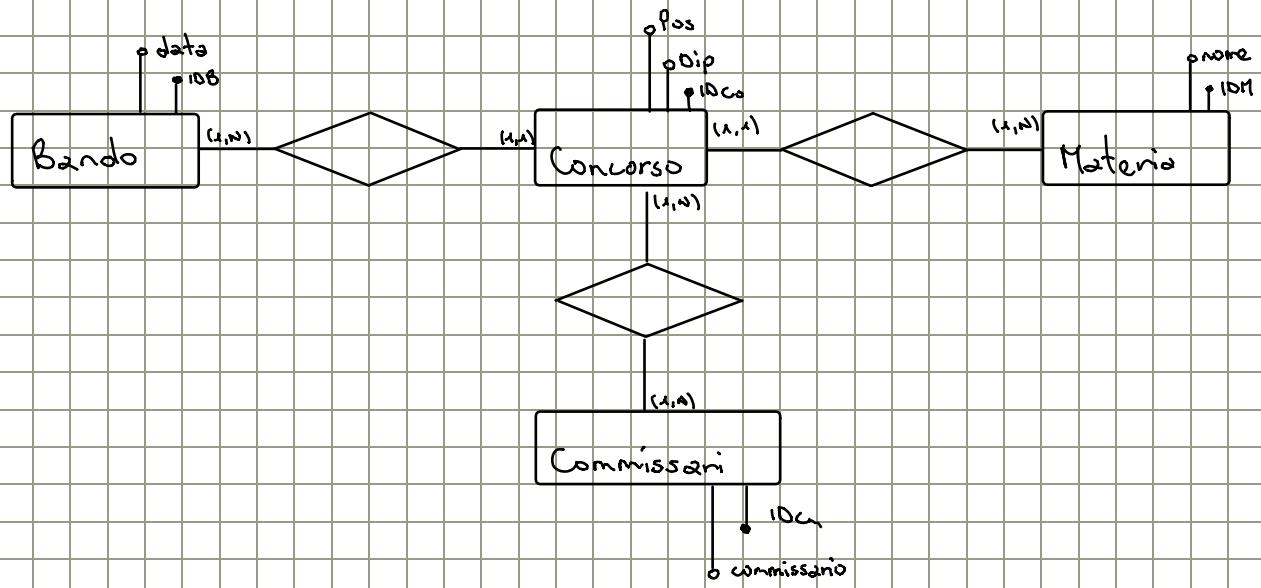


IDC_n → Commissario

IDM → Materia

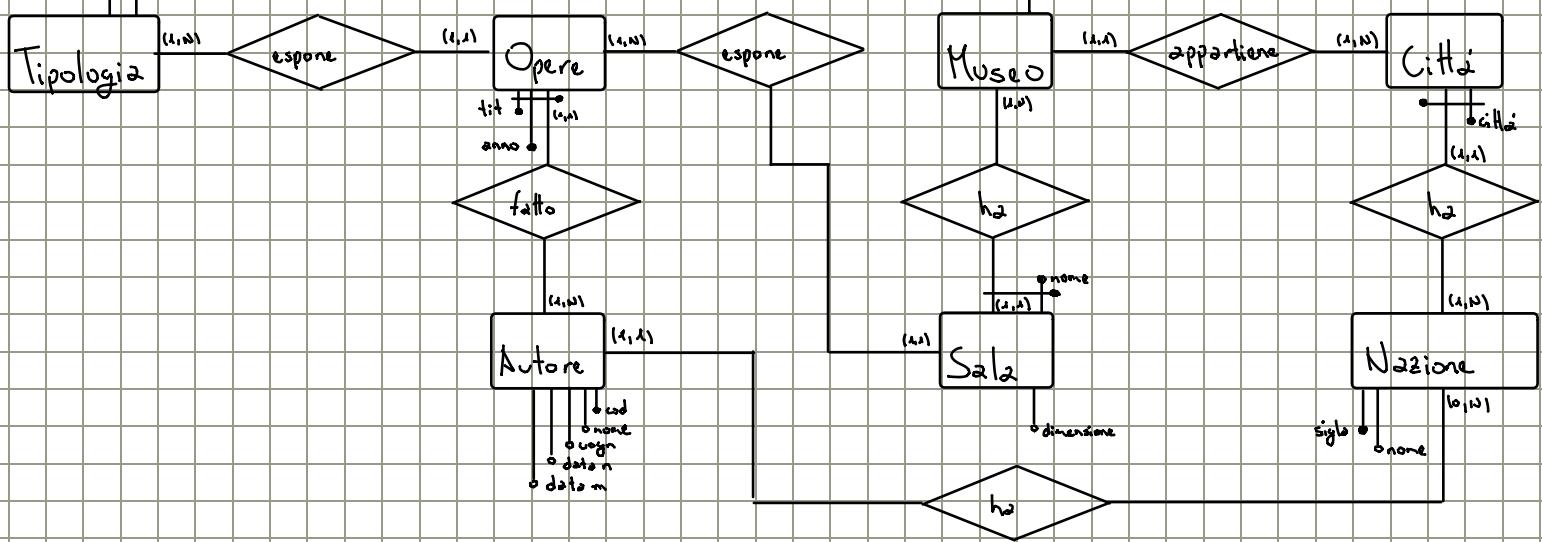
IDB → data

IDC → IDBando, Pos, Dip, IDMat

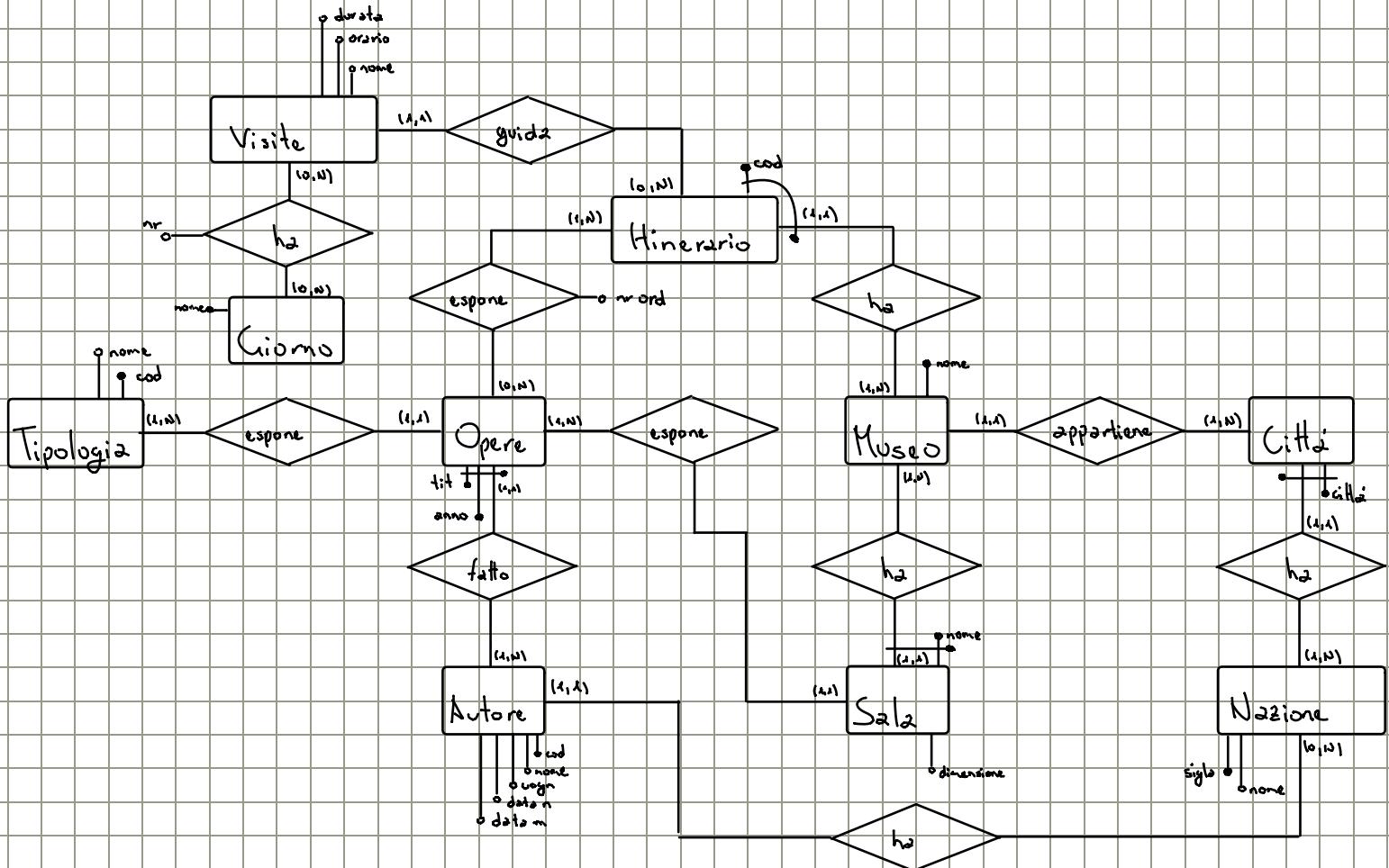


Esercizio 11-02-22 UNIROMA 3

1:



2:



SQL:

2) SELECT codice sum(importo) AS Importo
FROM RateEmesse
GROUP BY codice

(b) `SELECT COUNT(O.wdice), P.cf
FROM Ordini O
RIGHT JOIN Personae P ON O.cliente = P.cf
GROUP BY P.cf`

c)

`WITH RateP AS (
SELECT ordine, sum(importo) AS RateP
FROM Rate
WHERE datapagamento IS NOT NULL
GROUP BY ordine
)`

`WITH RateR AS (
SELECT ordine, sum(importo) AS RateR
FROM RateEmesse
GROUP BY ordine
)`

`SELECT ordine, RateR, RateP, RateR - RateP AS Debito
FROM RateEmesse E`

`JOIN RateP P ON P.ordine = E.ordine
JOIN RateR R ON R.ordine = E.ordine`

d) `WITH Debito AS (`

`SELECT ordine, RateR, RateP, RateR - RateP AS Debito
FROM RateEmesse E`

`JOIN RateP P ON P.ordine = E.ordine
JOIN RateR R ON R.ordine = E.ordine`

)

```
SELECT O.*, debito  
FROM ordine O  
JOIN debiti D ON D.ordine = O.ordine  
WHERE D.debito = (SELECT MAX(debito)  
                  FROM debiti)
```

Esame 19-07-2021 ROMATRE

a) SELECT P.sigla, P.nome, numero prime dosi + numero immunizzati
FROM Province P

JOIN Vaccinazioni V ON P.sigla = V.provincia

WHERE V.data = 25-06-2021

b) SELECT sum(numero prime dosi), data

FROM Vaccinazioni

GROUP BY data

ORDER BY data

c) WITH AbitantiT AS (

SELECT sigla, sum(Abitanti) AS Abitanti

FROM Province P

GROUP BY P.regione

)

WITH ImmunizzatiT AS (

SELECT sum(V.immunizzati) AS Immunizzati, provincia

FROM Vaccinazione V

JOIN Province P ON V.provincia = P.sigla

WHERE V.data = 25-06-2021

GROUP BY P.regione

)

SELECT I.Immunizzati, I.Immunizzati / A.Abitanti * 100 AS Percentuale

FROM ImmunizzatiT I

JOIN AbitantiT A ON I.provincia = A.sigla

d) WITH Statistica AS

```
SELECT I.Immunizzati, I.Immunizzati / A.abitanti * 100 AS Percentuale, Regione  
FROM Immunizzati I  
JOIN Abitanti A ON I.provincia = A.sigla  
)
```

SELECT Immunizzati, Percentuale, Regione

FROM Statistica

WHERE Percentuale >= (SELECT AVG(Percentuale)
FROM Statistica)

2) chiavi: CV

CV → CF, data, CT

CF → nome, codASL

codASL → ASL

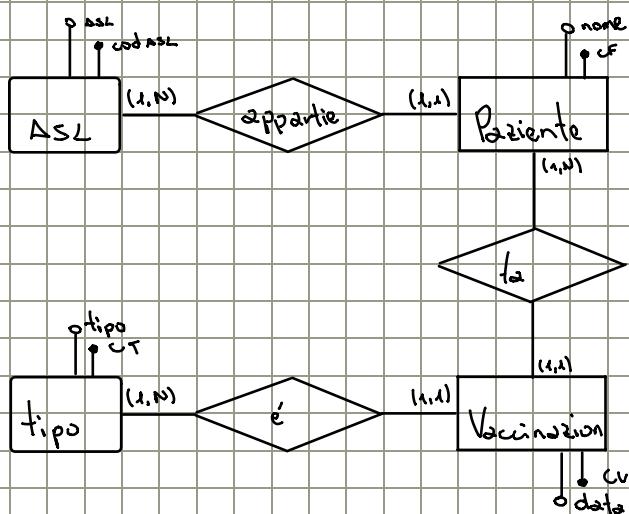
CT → tipo

tipi | CT, tipo |

ASL (codASL, ASL)

Pazienti | CF, nome, codASL [FK] |

Vaccinazione (CV, data, CT [FK])



Esame 2-2-2021

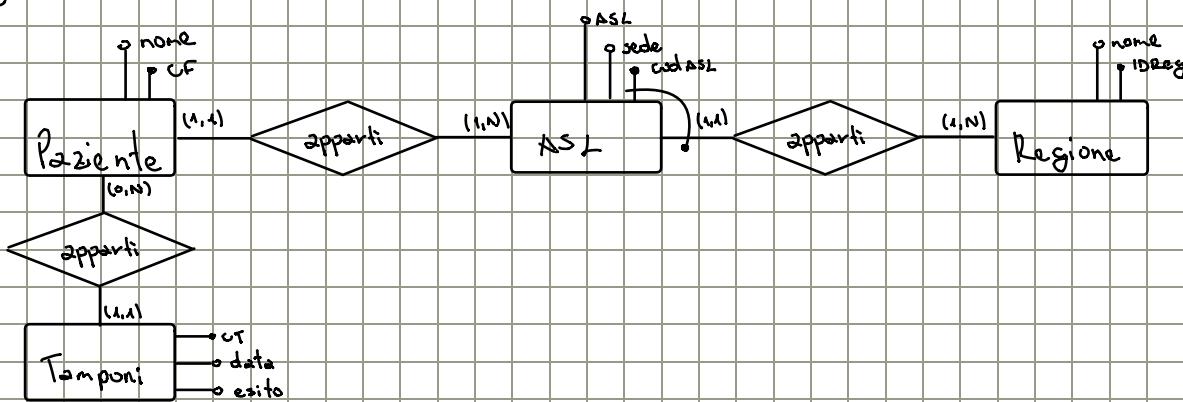
Chiavi: CT

CT \rightarrow CF, data, esito

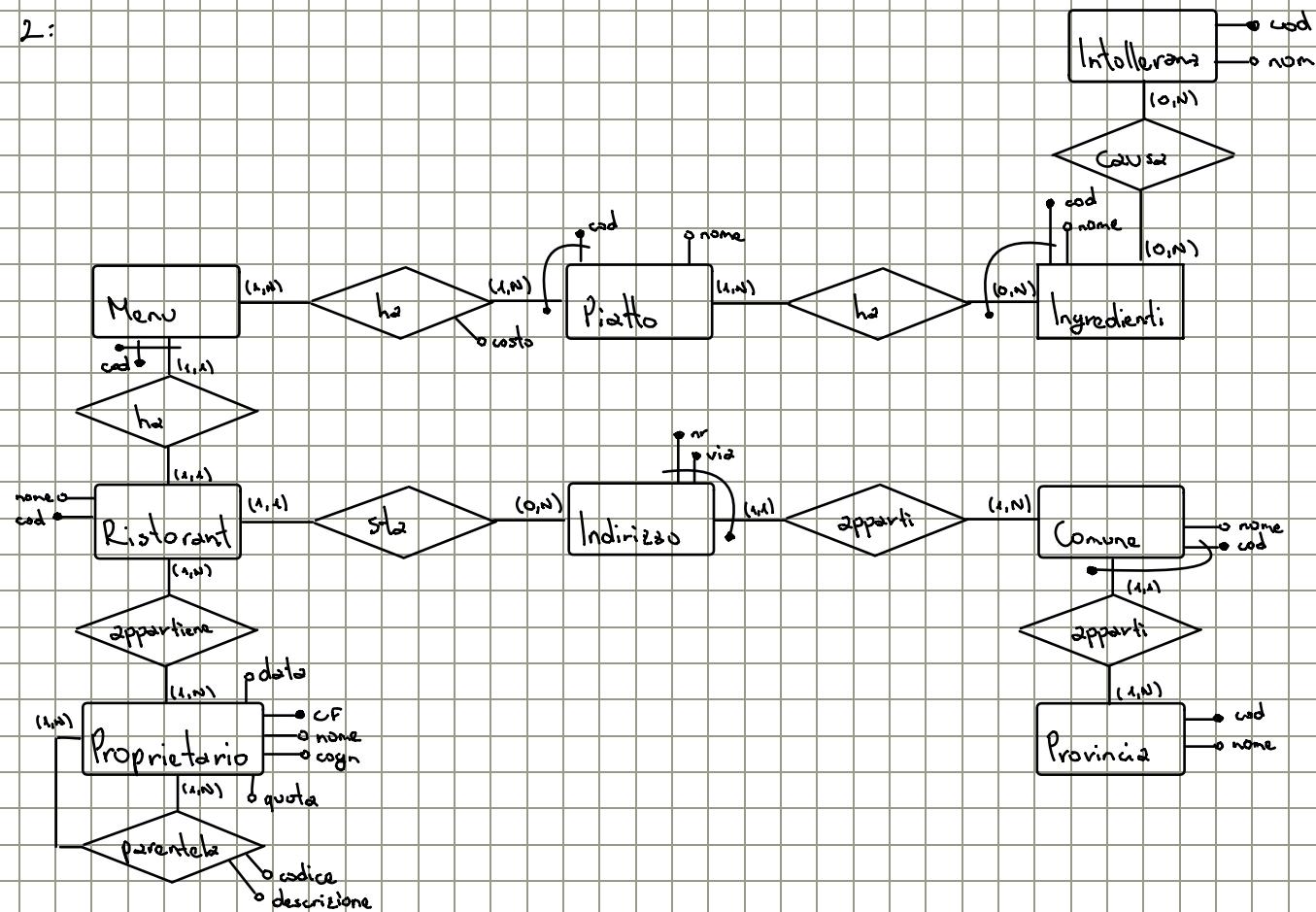
CF \rightarrow nome, codASL, IDReg

codASL, IDReg \rightarrow sedeASL, ASL

IDReg \rightarrow nome



2:



2) $\Pi_{sigla}(\text{Provincia}) - \Pi_{sigla}(\sigma_{data = 31-01-2024}(\text{Provincia} \bowtie \text{sigla_provincia} \text{ Tamponi}))$

SELECT P.*

FROM Provincia P

```
WHERE P.sigla NOT IN (SELECT Provincia  
FROM Tamponi T  
WHERE data = "31-01-2021")
```

c) SELECT P.sigla, P.nome, T.numT, T.numP
FROM Provincia P

JOIN Tamponi T ON sigla = provincia

WHERE data = 25-01-21

d) WITH somme AS (

```
SELECT sum(T.numT) AS numTr, sum(T.numP) AS numPr, regione
```

FROM Tamponi T

JOIN Provincia P ON sigla = provincia

GROUP BY regione

)

```
SELECT numTr, numPr, (numPr * 100) / numTr AS Percentuale, regione
```

FROM somme

e) WITH percentuali AS (

```
SELECT numTr, numPr, (numPr * 100) / numTr AS Percentuale, regione
```

FROM somme

)

WITH media AS (

```
SELECT Avg(Percentuale) AS Media
```

FROM percentuali

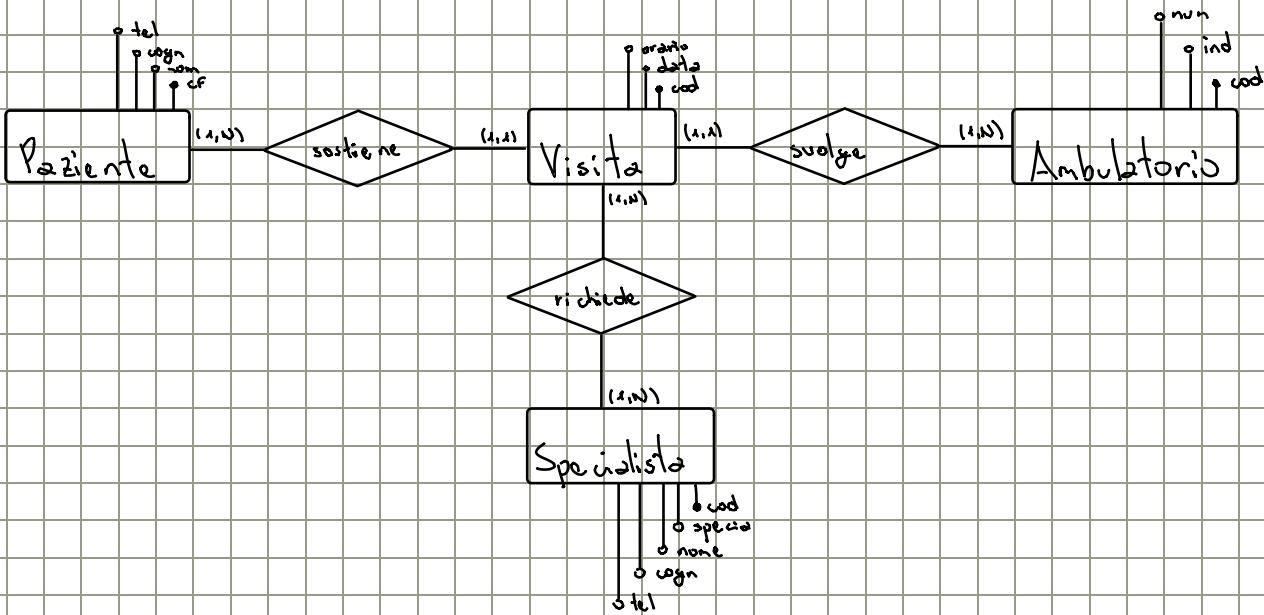
)

SELECT *

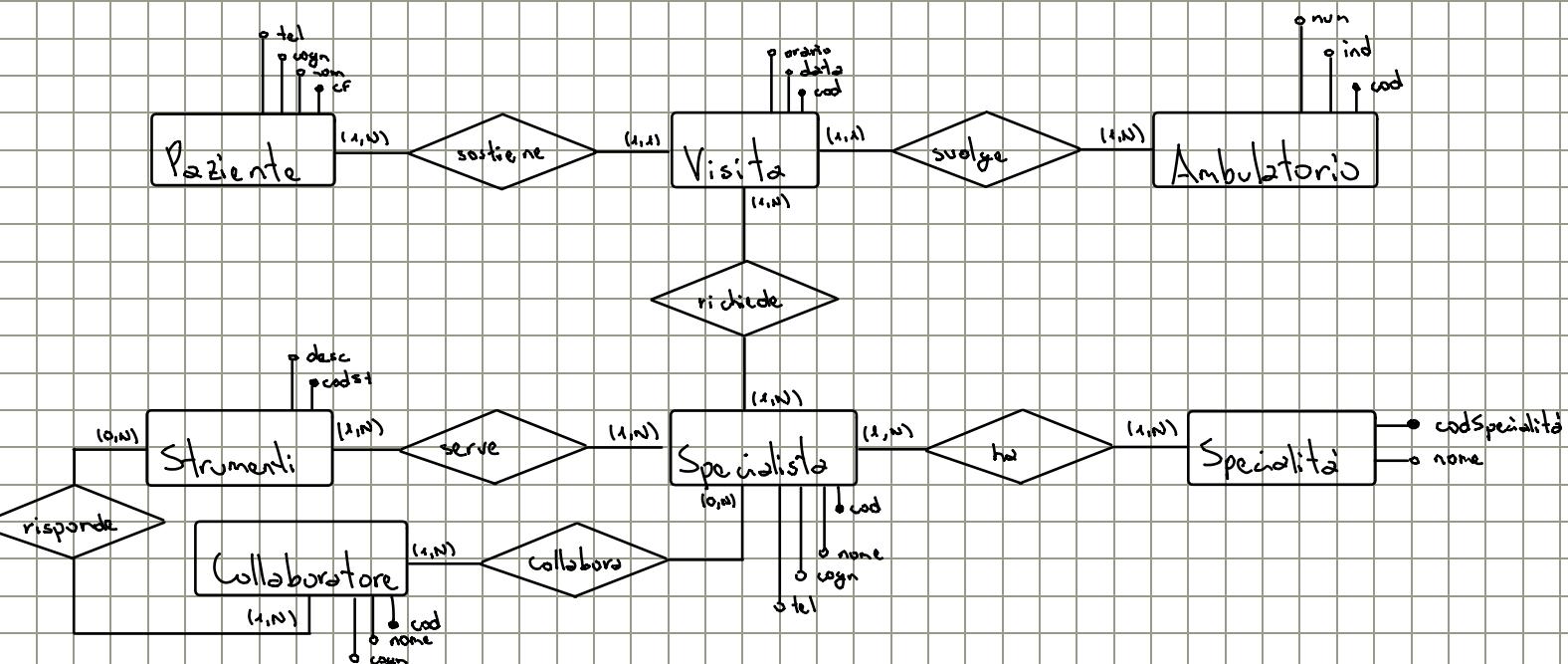
FROM Percentuali, media

WHERE percentuale > media

Esame 24-01-2025



Specialista-Visita (codS, codV) relazione N:N tra Visita Specialità



Paziente (CF, nome, cogn, tel)

Visita (codV, data, orario, CF[FK], codA[FK])

Ambulatorio (codA, nomeA, telA)

Specialista (codS, nome, cogn, tel)

Specialista-Visita (codS[FK], codV[FK]) relazione N:N tra Visita Specialità

Specialista (codSp, desc)

Strumenti (codSt, desc)

Collaboratore (codC, nome, cogn)

Specialista_Collaboratore (codSp [FK], codC [FK]) rel N:N tra Specialista_Collaboratore

Strumenti_Collaboratore (codSt [FK], codC [FK]) rel N:N tra Strumenti_Collaboratore

Specilista_Specialità (codSp [FK], codSpe [FK]) rel N:N tra Specilista_Specialità

Specilista_Strumenti (codSp [FK], codSt [FK])

Query

a) SELECT V.codV, V.data, V.orario, R.codSpecialista, P.cf

FROM Visita V

JOIN Richiede R ON V.codV = R.codV

JOIN Paziente P ON V.cf = P.cf

WHERE data = CURRENT_DATE

ORDER BY R.codSpecialista, V.orario

b) WITH usato AS (

SELECT codSp, codSt, COUNT(codSp) AS utilizzoSp

FROM Specilista_Strumento

GROUP BY codSt, codSp

|

SELECT S.codSp, S.codSt, U.utilizzoSp

FROM Specilista_Strumento S

JOIN Usato U ON S.codSp = U.codSp

WHERE utilizzoSp = (SELECT MAX(utilizzoSp)

FROM usato

c) $\text{SELECT CF, data, codV}$

FROM Visita

$\text{WHERE data} \geq \text{CURRENT_DAY} - \text{INTERVAL}'1 YEAR'$

GROUP BY CF

ORDER BY CF, data

d) $\Pi_{CF, data, codV} (\sigma_{\text{data} = \text{CURRENT_DATE} - \text{INTERVAL}'1 YEAR'} (\text{Visita}))$

4:

a) Trasformarlo in 1NF: non ci devono essere valori ripetuti e i dati devono essere atomici.

b) Trasforma in 2NF: deve essere in 1NF e bisogna eliminare tutte le dipendenze parziali

c) Trasforma in 3NF: deve essere in 2NF e bisogna eliminare tutte le dipendenze transitiva

d) Trasforma in BCNF: deve essere in 3NF e per ogni dipendenza funzionale $X \rightarrow Y$ X deve essere una superchiave

Libri (codL, titolo, genere, ISBN, anno pubblicazione, editore [FK])

Autore (codA, nome, cogn)

Sezione (nomeSez, posScat(fale))

Editore (editore, indirizzoEditore, cittàEditore)

Autore-Libro ()

Sezione-Libro ()

Editore-Libro ()

Utenti (codUtente, nome, cogn, indirizzo[FR], email)

Indirizzi (indirizzo, città, cnp)

Utenti-Indirizzo ()

Prestito (codP, codL[FK], codU[FK], data prestito, data scad, data rest)

Auto (#telaio, targa, colore, nomeC, nomeMod, nomeMarca)

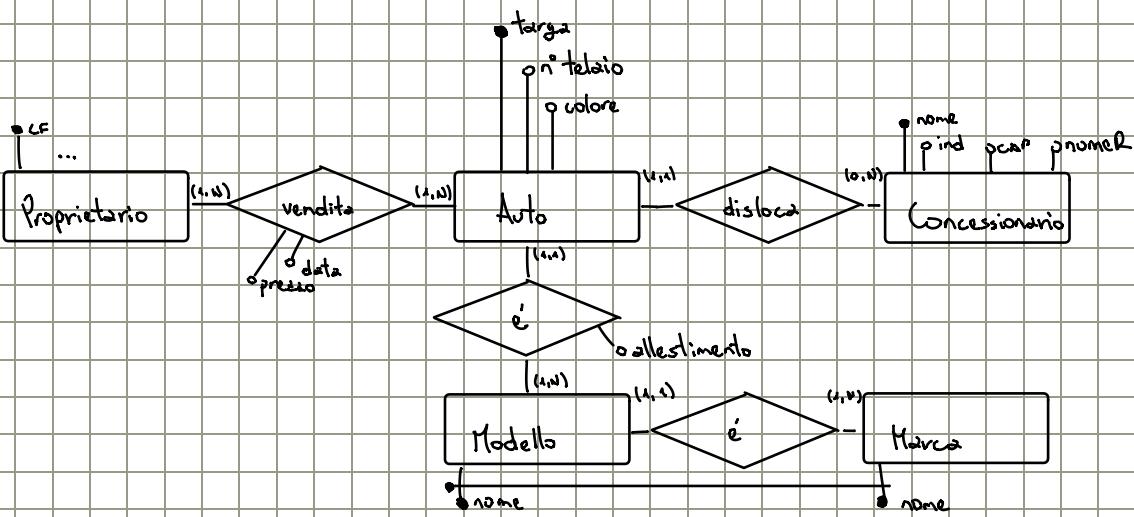
Concessionario (nome, indirizzo, cap, nomeResponsabile)

Modello (nomeMod, nomeMarca)

Marca (nomeMarca)

Proprietario (CF, nome, cognome)

Vendita (CF, #telaio, costo, data)



(a) SELECT count(targa) AS nvendite, cap

FROM auto A

JOIN concessionario C ON A.nomeC = C.nomeC

GROUP BY cap

(b)

SELECT cf, targa, data

FROM Vendita

WHERE cf, targa, data = (SELECT cf, targa, MAX(data)

FROM Vendita

GROUP BY targa)

(c) WITH venditeAuto AS

SELECT A.modello, A.marca, COUNT(V.targa) AS Vendite
FROM auto A

JOIN Vendite V ON A.telato = V.telato

GROUP BY A.modello, A.marca

)

SELECT A.modello, A.marca, MAX(Vendite)

FROM venditeAuto A

L: chiavi: cm, co, cp, cittM

cp → nomeP, prezzo

cm, cittM → nomeM

cp, cm, cittM → giacenza

co, cp, cm, cittM → QuantitàO

Prodotto (cp, nomeP, prezzo)

Magazzino (cm, cittM, nomeM)

Prodotto_Magazzino (cp, cm, cittM, giacenza)

Ordine (co)

Torna (co, cp, cm, cittM, quantità)

Chiave: cod

cod → titolo, cu, cn

cu, cn → città

cn → nazione

cod → cm

cm → materia, ca

ca → area

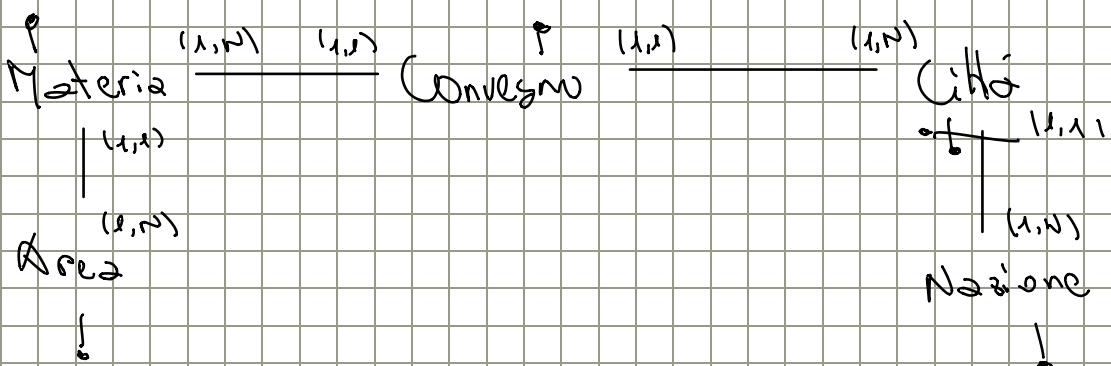
Convegno (cod, titolo, cu*, cm*, cn*)

Nazione (cn, nome)

Città (cu, cn*, nome)

Materia (cm, materia, ca*)

Area (ca, nome)



2) SELECT C.codC, C.nome, SUM(P.importo) AS ImportoTotale
FROM Committente C
JOIN Progetto P ON P.codC = C.codC
GROUP BY C.codC.

b) WITH Uscita AS (
SELECT I.nome, COUNT(P.progetto) AS conta
FROM Interagisci I
JOIN Progetto P ON I.codC = P.codC
GROUP BY I.nome
)

SELECT nome, MAX(conta) AS numero
FROM uscita

c) SELECT I.nome, SUM(P.importo) AS Importo, A.nomeD
FROM Impiegato I
JOIN Interagisci A ON I.nomeI = A.nomeI
JOIN Progetto P ON A.cod = P.cod
WHERE I.ruolo = "direttore"
GROUP BY A.nomeD