LA FABBRICA DI MARCHINGEGNI

Progetto di Basi di Dati

STUDENTI

Julie Chicca – Mat. 559364
j.chicca@studenti.unipi.it

Mara Magnanini – Mat. 566911
m.magnanini3@studenti.unipi.it

Serena Pestrin – Mat. 566913
s.pestrin@studenti.unipi.it

Anno accademico 2020/2021

1. Descrizione del dominio e Classi

La Fabbrica di Marchingegni è un'azienda che produce e vende **Marchingegni**. Il Sistema è un software utilizzato dalla fabbrica per gestire i dipendenti e gestire la costruzione e vendita dei Marchingegni.

Ciascuna **Tipologia di Marchingegno** ha un nome e un prezzo di vendita.

È responsabilità del **Dipendente** registrare la proprio entrata e uscita dalla Fabbrica, le **Presenze** vengono poi registrate nel Sistema e devono contenere ora di entrata, ora di uscita e data.

I **Team** di Dipendenti costruiscono i Marchingegni e al completamento il **CapoProgetto** assegna il numero di serie e specifica il team che l'ha costruito. Gli altri membri del Team devono confermare la loro reale partecipazione.

La Fabbrica vende Marchingegni soltanto a **Clienti registrati**, di cui il Sistema deve registrare le generalità.

Quando un Cliente effettua una **Prenotazione** deve indicare la quantità di Marchingegni e la data ultima di consegna.

L'AddettoVendite ha la responsabilità di gestire le vendite e di accettare le Prenotazioni.

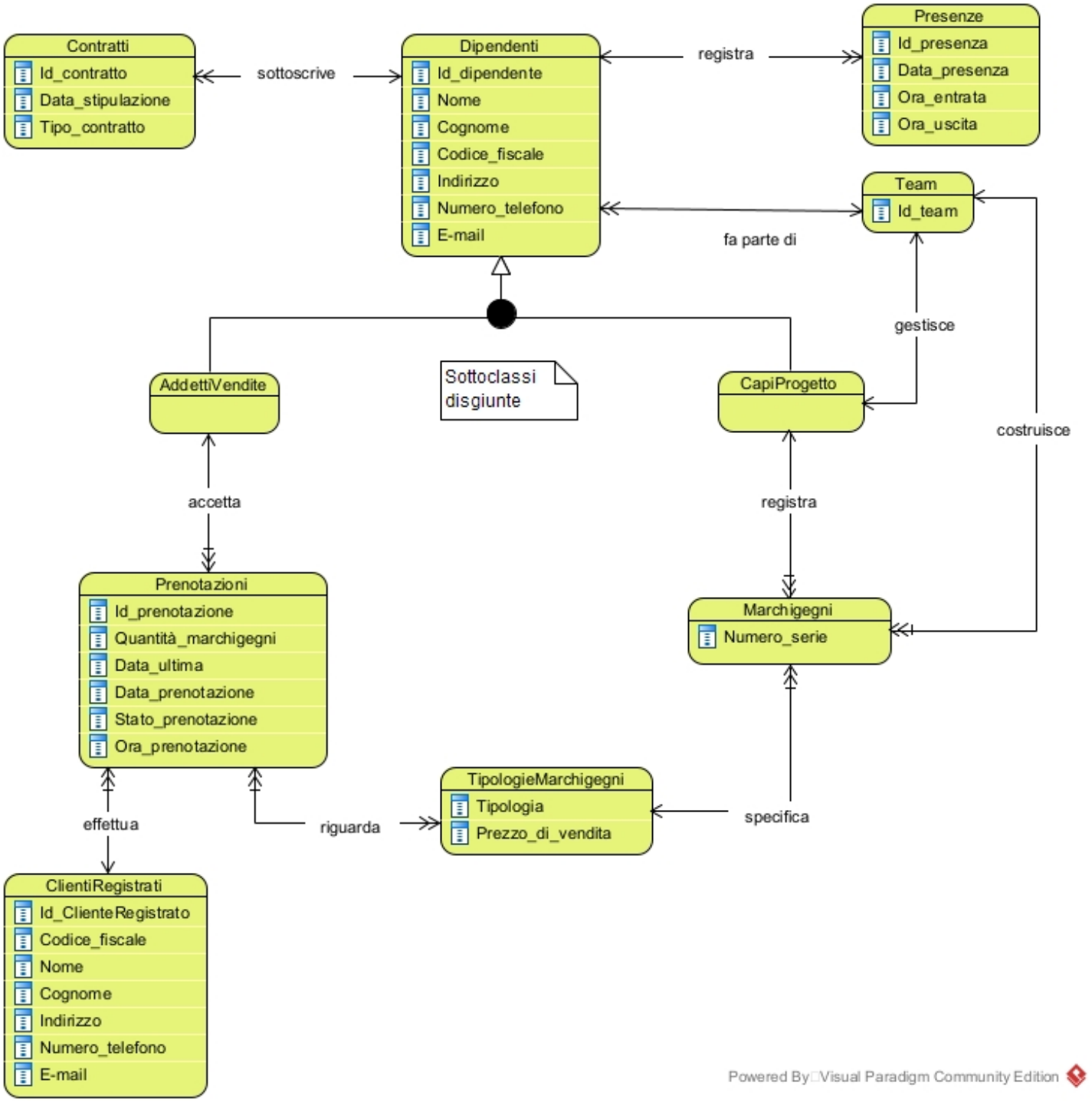
I **Contratti** di lavoro sono mensili, che possono cambiare di mese in mese, il Sistema registra per ogni dipendente il Contratto attuale e inoltre mantiene le informazioni di quelli stipulati in passato.

Termine	Descrizione	Collegamenti
Marchingegni	Prodotto della Fabbrica	Prenotazioni CapiProgetto Team TipologieMarchingegno
Dipendenti	Persona che lavora nella Fabbrica	Presenze Contratti Team CapiProgetto AddettiVendite
Team	Gruppo di dipendenti che costruiscono i marchingegni	Dipendenti CapiProgetto
CapiProgetto	Dipendente che supervisiona la costruzione di un progetto	Dipendenti Team Marchingegni
Presenze	Turno di lavoro di un Dipendente	Dipendenti
ClientiRegistrati	Cliente registrato sul sito web della Fabbrica	Prenotazioni
AddettiVendite	Dipendente che gestisce le vendite	Prenotazioni Dipendenti
Contratti	Contratto mensile di ogni Dipendente	Dipendenti
Prenotazioni	Effettuata dal Cliente registrato per comprare un Marchingegno	Clienti registrati AddettiVendite Marchingegni
TipologieMarchingegni	Modelli diversi di marchingegni	Marchingegni

Note:

- L'Ufficio del Personale usa il sistema per registrare i contratti di tutti i Dipendenti.

2. Schema Concettuale



Note:

Le date sono rappresentate nel formato aaaa/mm/gg.

Gli orari sono rappresentati nel formato hh:mm:ss.

Vincoli non catturati graficamente:

- Il cliente può recedere gratuitamente dalla prenotazione solo entro 48 ore.
- Se la prenotazione è accettata il Cliente anticipa il 40% del costo.
- Il contratto è rinnovato ogni metà del mese.
- Ogni membro del team deve confermare la sua reale partecipazione alla produzione di un Marchingegno.
- Il Capo Progetto, oltre a registrare il numero di serie al marchingegno, partecipa alla costruzione e specifica il team che l'ha prodotto.
- L'AddettoVendite gestisce la vendita e la registra sul Sistema.
- Alla fine di ogni mese il Sistema deve calcolare lo stipendio di ogni Dipendente.

Vincoli Intrarelazionali:

- **Contratti:**
  - Id_contratto è chiave primaria della classe.
  - Il Tipo_contratto indica il contratto predefinito (per esempio a ore lavorative).
- **Dipendenti:**
  - Id_dipendente è chiave primaria della classe.
- **Presenze:**
  - Id_presenza è chiave primaria della classe.
- **Team:**
  - Id_team è chiave primaria della classe.
- **Prenotazioni:**
  - Id_prenotazione è chiave primaria della classe.
  - Data_ultima indica il limite di tempo entro il quale deve avvenire la consegna.
- **Marchingegni:**
  - Numero_di_serie è chiave primaria della classe.
- **ClientiRegistrati:**
  - Id_ClienteRegistrato è chiave primaria della classe.
- **TipologieMarchingegni:**
  - Tipologia è chiave primaria della classe.

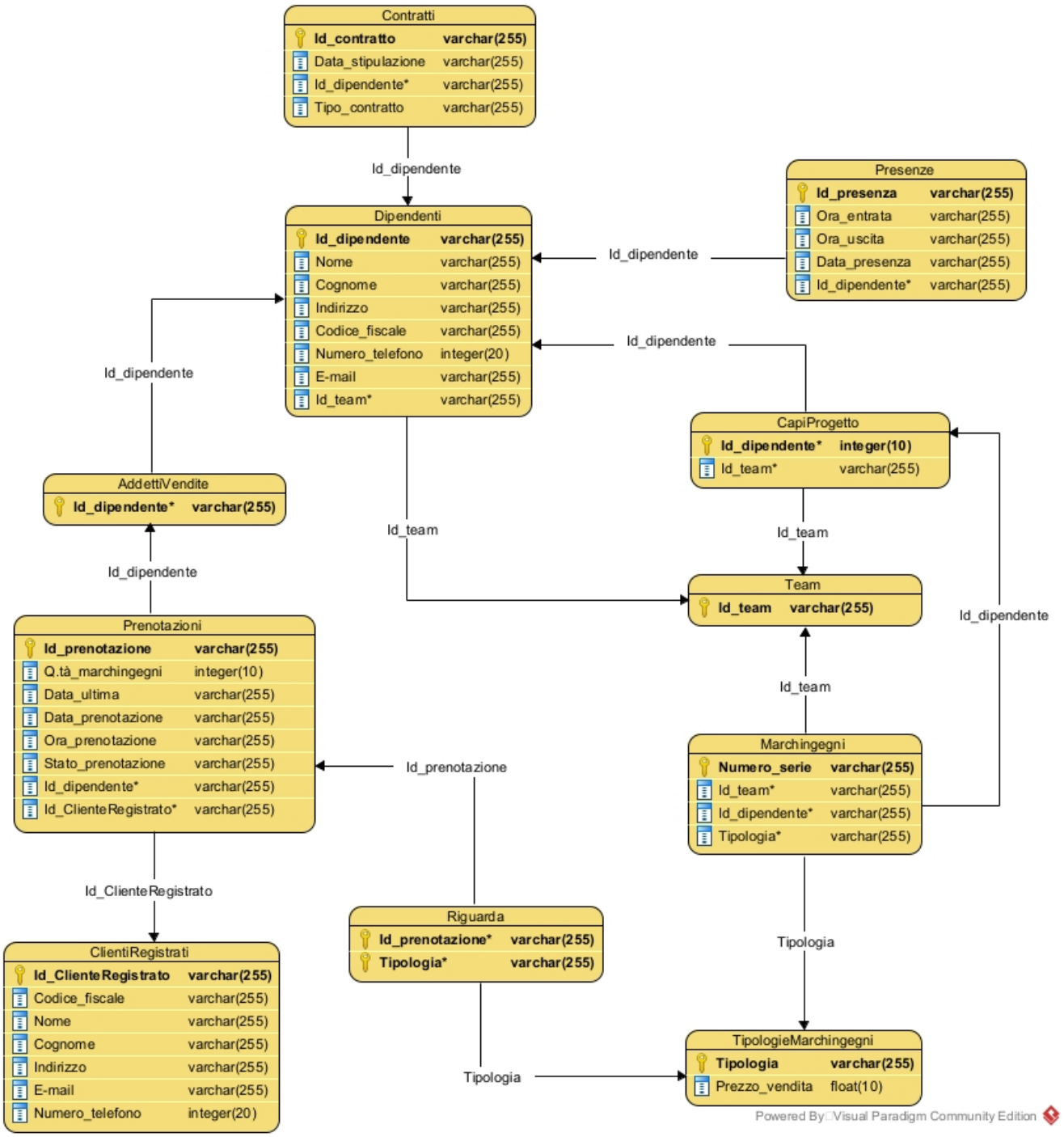
Vincoli Interrelazionali

- La prenotazione può essere effettuata solo da un cliente registrato.
- Ogni Dipendente è responsabile di registrare l'entrata e l'uscita dalla Fabbrica.
- Ogni metà mese il Dipendente deve sottoscrivere un nuovo contratto.
- Lo Stato_prenotazione della Prenotazione viene accettata/rifiutata o è in attesa dell'approvazione dell'AddettoVendite.

3. Schema Logico-Relazionale

Abbiamo deciso di interpretare le gerarchie nel modo seguente:

- La gerarchia Dipendenti-AddettoVendite-CapiProgetto è stata tradotta con un partizionamento verticale.



Note:

Id_dipendente nella classe Marchingegni rappresenta il capo progetto che ha assegnato il numero di serie.

Schema testuale in forma R(IdR, …, A*):

Contratti(Id_contratto, Data_stipulazione, Tipo_contratto, Id_dipendente*)

Dipendenti(Id_dipendente, Nome, Cognome, Codice_fiscale, Indirizzo, Numero_telefono, E-mail, Id_team*)

Presenze(Id_presenza, Data_presenza, Ora_entrata, Ora_uscita, Id_dipendente*)

AddettiVendite(Id_dipendente*)

Team(Id_team)

CapiProgetto(Id_dipendente*, Id_team*)

Marchingegni(Numero_serie, Id_team*, Id_dipendente*, Tipologia*)

TipologieMarchingegni(Tipologia, Prezzo_vendita)

Prenotazioni(Id_prenotazione, Q.tà_marchingegni, Data_ultima, Data_prenotazione, Ora_prenotazione, Stato_prenotazione, Id_dipendente*, Id_ClienteRegistrato*)

Riguarda(Id_prenotazione*, Tipologia*)

ClientiRegistrati(Id_ClienteRegistrato, Codice_fiscale, Nome, Cognome, Indirizzo, E-mail, Numero_telefono)

Dipendenze funzionali

Relazione	Dipendenze Funzionali
Contratti	Id_contratto -> Data_stipulazione, Tipo_contratto, Id_dipendente*
Dipendenti	Id_dipendente -> Nome, Cognome, Codice_fiscale, Indirizzo, Numero_telefono, E-mail, Id_team* Codice_fiscale -> Nome, Cognome
Presenze	Id_presenza -> Data_presenza, Ora_entrata, Ora_uscita, Id_dipendente*
CapiProgetto	Id_dipendente* -> Id_team*
Marchingegni	Numero_serie -> Id_team*, Tipologia*, Id_dipendente*
TipologieMarchingegni	Tipologia-> Prezzo_vendita
Prenotazione	Id_prenotazione -> Q.tà_marchingegni, Data_ultima, Data_prenotazione, Ora_prenotazione, Stato_prenotazione, Id_dipendente*, Id_ClienteRegistrato*
ClientiRegistrati	Id_clienteRegistrato -> Codice_fiscale, Nome, Cognome, Indirizzo, Numero_telefono, E-mail Codice_fiscale -> Nome, Cognome

Tutte le dipendenze funzionali soddisfano la forma normale di Boyce-Codd (BCNF).

4. Interrogazioni in SQL

a) Uso di proiezione, join e restrizione

```sql
/*Id_contratto di tutti i contratti stipulati nel mese di Luglio
e sottoscritti da dipendenti con cognome 'Rossi'.*/
SELECT Id_contratto
FROM Contratti C, Dipendenti D
WHERE C.Id_dipendente = D.Id_dipendente
AND Data_stipulazione LIKE '____/07/__'  AND D.Cognome = 'Rossi'
```

b) Uso di group by, con having, where e sort

```sql
/*Tipologia e numero di Marchingegni, se maggiori di 2,
approvati dal capoprogetto con Id_dipendente '034' e
ordinati per tipologia.*/
SELECT Tipologia, count(Numero_serie) as Numero_marchingegni_registrati
FROM Marchingegni
WHERE Id_dipendente = '034'
GROUP BY Tipologia
HAVING count(Numero_serie) > 2
ORDER BY Tipologia
```

c) Uso di join, group by con having e where

```sql
/*Data, codice fiscale e numero di prenotazioni
effettuate dai clienti che hanno fatto più di un ordine
al giorno nel mese di Luglio 2020.*/
SELECT Data_prenotazione, Codice_fiscale, count(Id_prenotazione) as Numero_prenotazioni
FROM Prenotazioni P, Clienti_registrati C
WHERE C.Id_cliente_registrato = P.Id_cliente_registrato
AND Data_prenotazione LIKE '2020/07/__'
GROUP BY Data_prenotazione, Codice_fiscale
HAVING count(Id_prenotazione) > 1
```

d) Uso di select annidata con quantificazione esistenziale

```
/*Gli Id di tutti i clienti che hanno
ordinato almeno un Orologio Sparachiodi.*/
SELECT Id_cliente_registrato
FROM Prenotazioni P
WHERE EXISTS (SELECT *
              FROM Riguarda R
              WHERE P.Id_prenotazione = R.Id_prenotazione
              AND Tipologia = 'Orologio Sparachiodi')
```

e) Uso di select annidata con quantificazione universale

```
/*Tutti i nomi dei dipendenti che non hanno
mai prodotto Tazzine Frullatrici.*/
SELECT Nome
FROM Dipendenti D
WHERE NOT EXISTS (SELECT Id_team
                  FROM Marchingegni M
                  WHERE M.Id_team = D.Id_team
                  AND Tipologia = 'Tazzina Frullatrice')
```

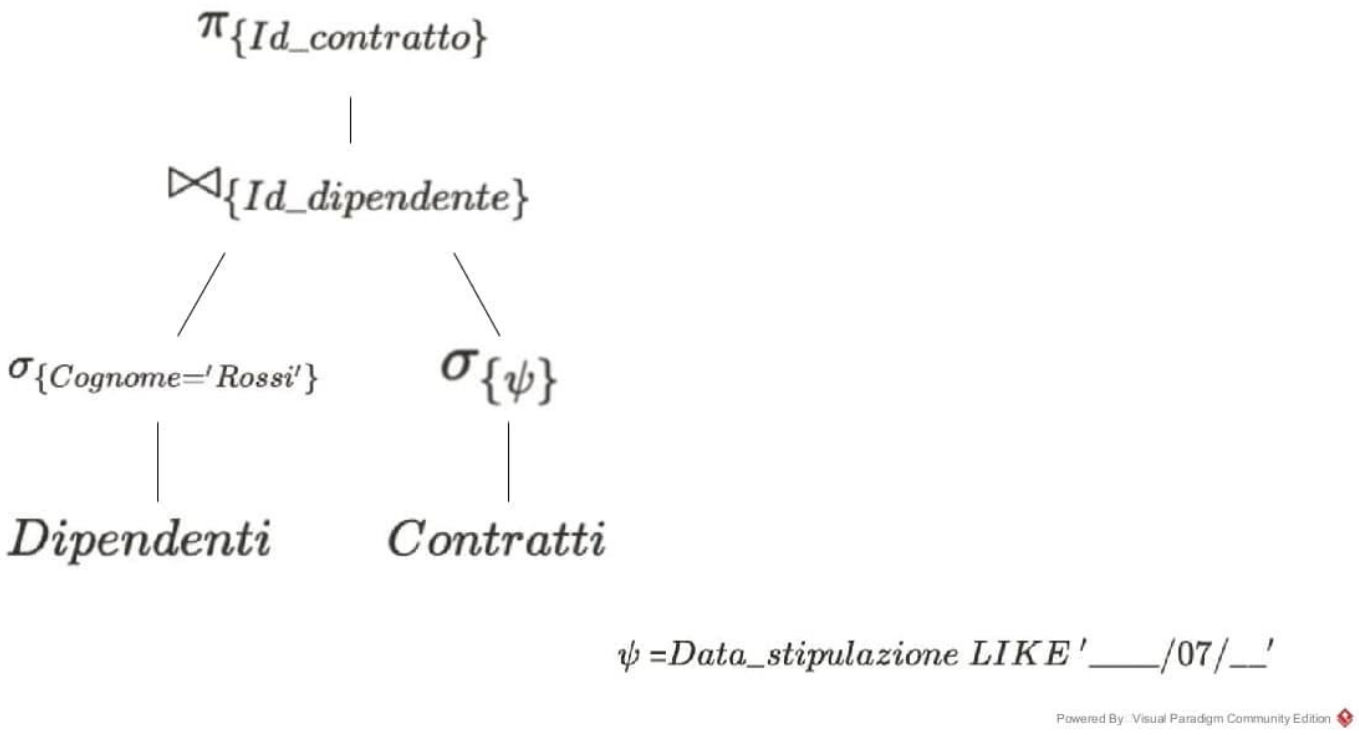
f) Uso di subquery di confronto quantificato usando una subquery

```
/*Nomi di tutti clienti che hanno
fatto una prenotazione di più di 10 marchingegni.*/
SELECT Nome
FROM Clienti_registrati C
WHERE 10 < ALL (SELECT Q.tà_marchingegni
                FROM Prenotazioni P
                WHERE P.Id_cliente_registrato = C.Id_cliente_registrato)
```

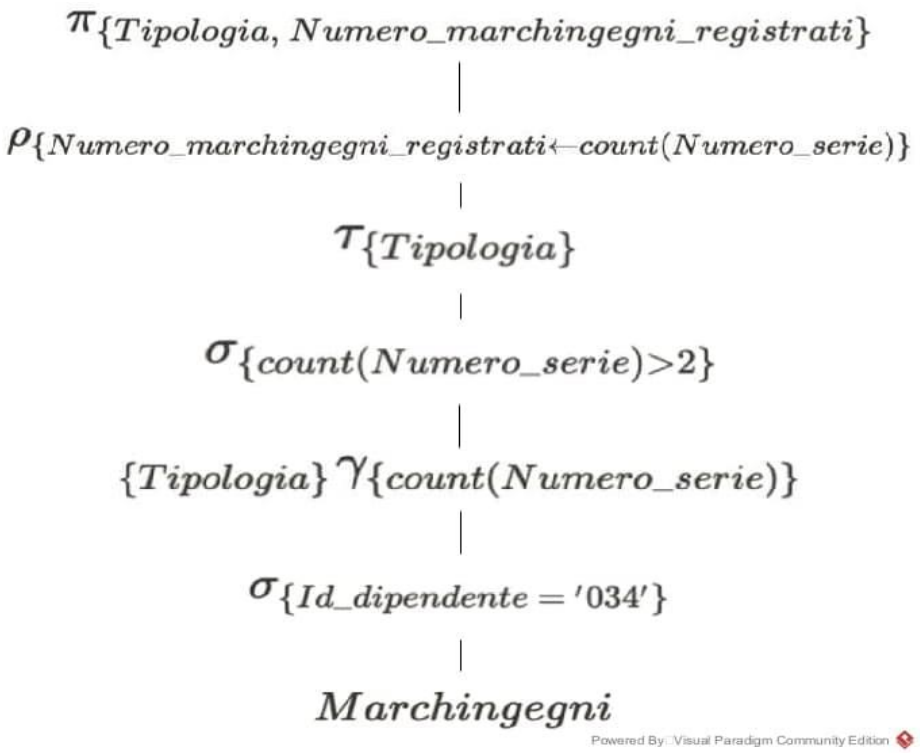
5. Piani di Accesso

I) Scrivere un piano di accesso logico delle query a), b), c);

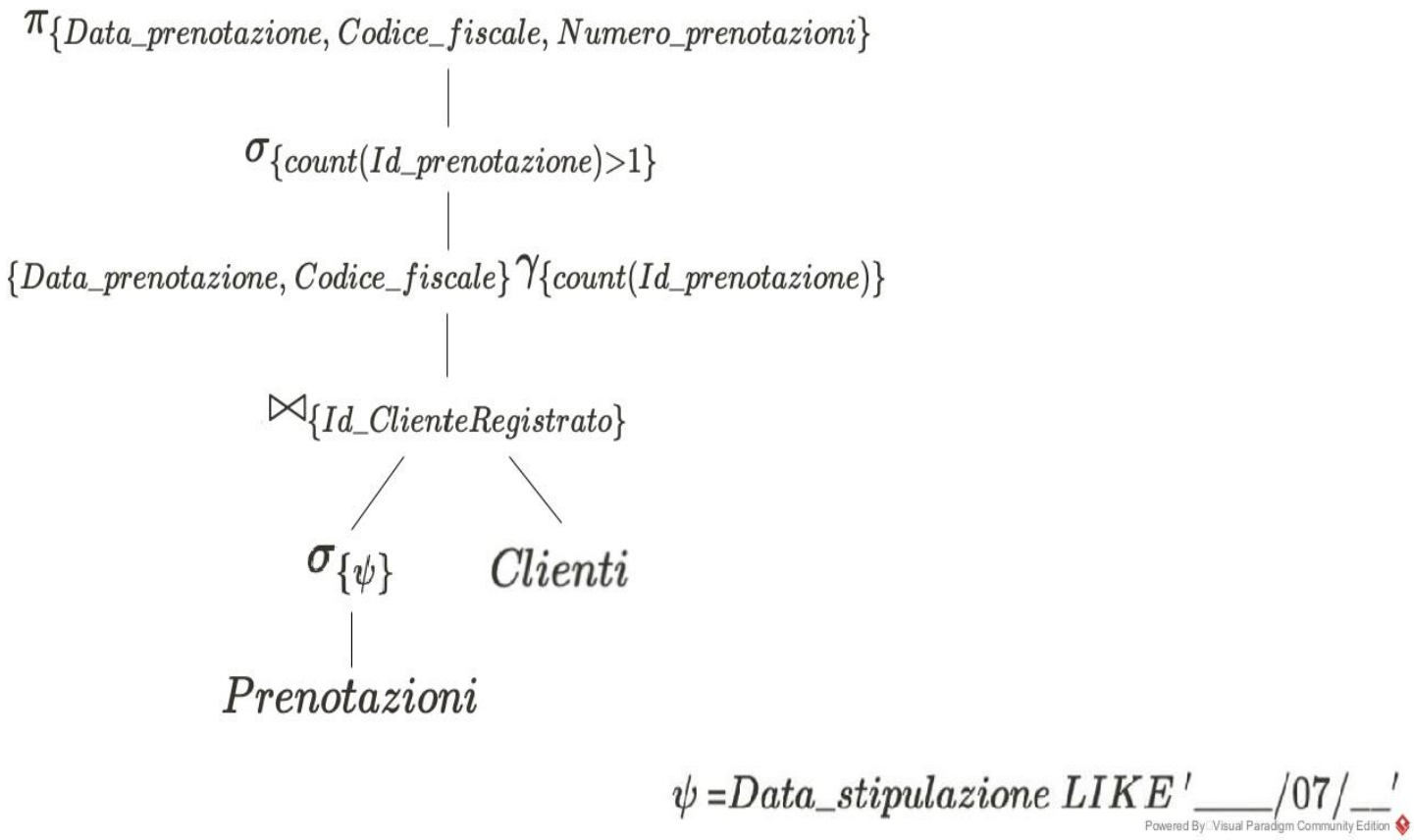
a)



b)

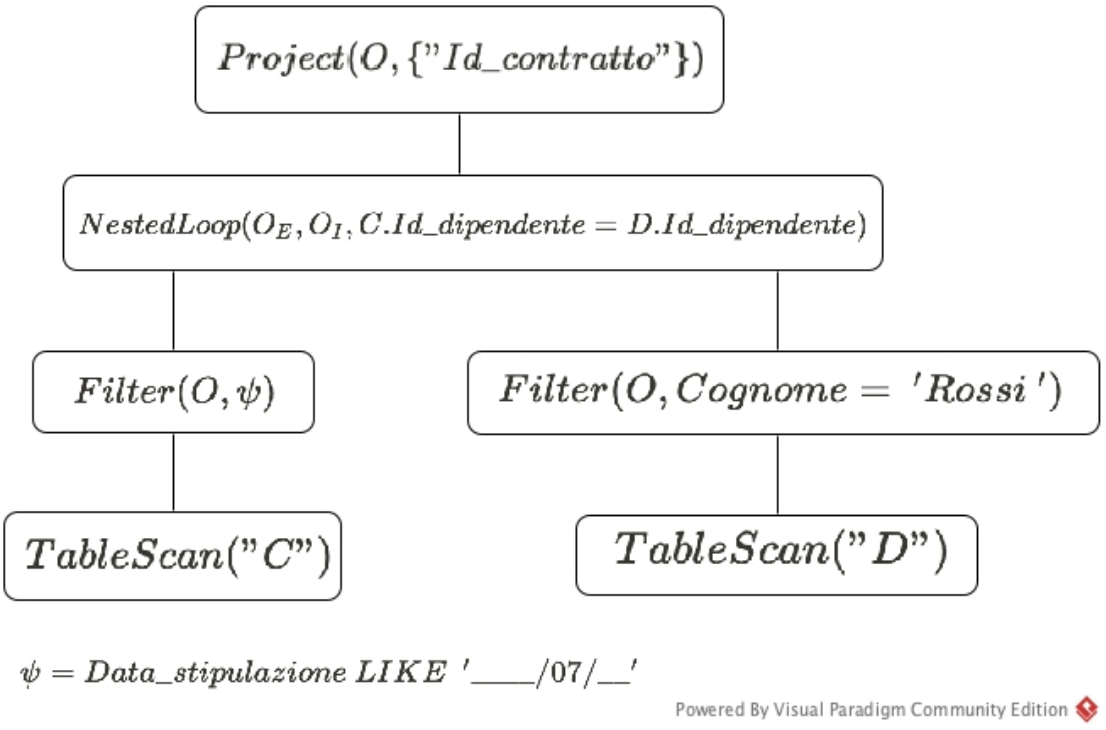


c)

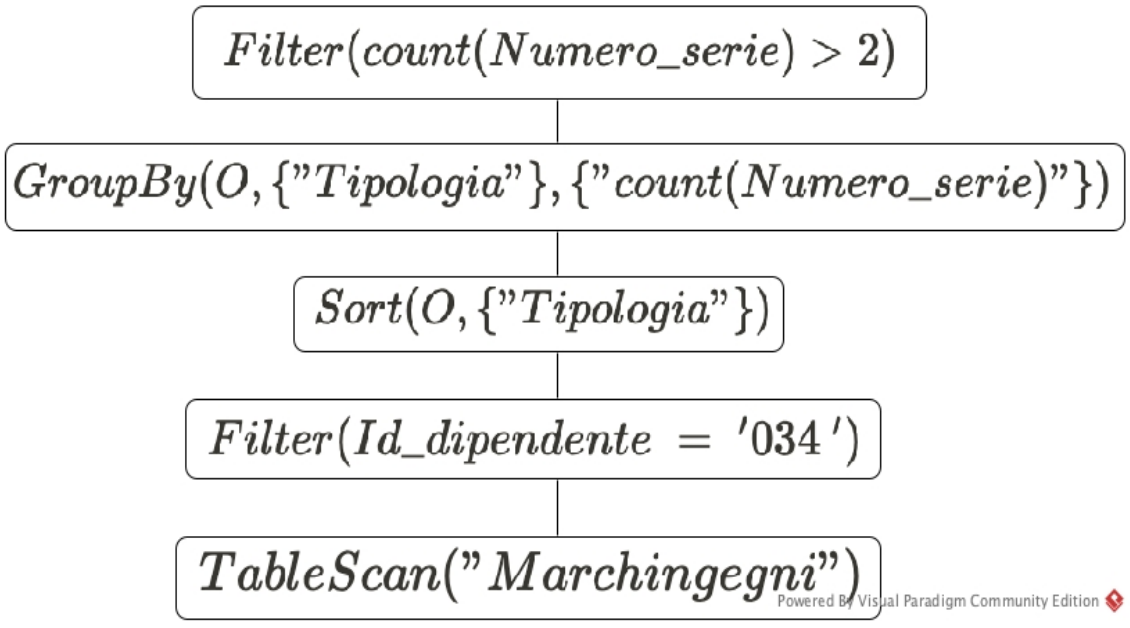

$\pi_{\{Data\_prenotazione, Codice\_fiscale, Numero\_prenotazioni\}}$
$\sigma_{\{count(Id\_prenotazione)>1\}}$
$\{Data\_prenotazione, Codice\_fiscale\}\gamma_{\{count(Id\_prenotazione)\}}$
$\bowtie_{\{Id\_ClienteRegistrato\}}$
$\sigma_{\{\psi\}}$
$Clienti$
$Prenotazioni$
$\psi = Data\_stipulazione\ LIKE\ '\_\_\_\_/07/\_\_'$
Powered By Visual Paradigm Community Edition

II) Scrivere un piano di accesso fisico efficiente per i tre piani di accesso logico al punto
I che non fanno uso di indici, e (opzionale) verificare se la sort prima della Group By può essere evitata;

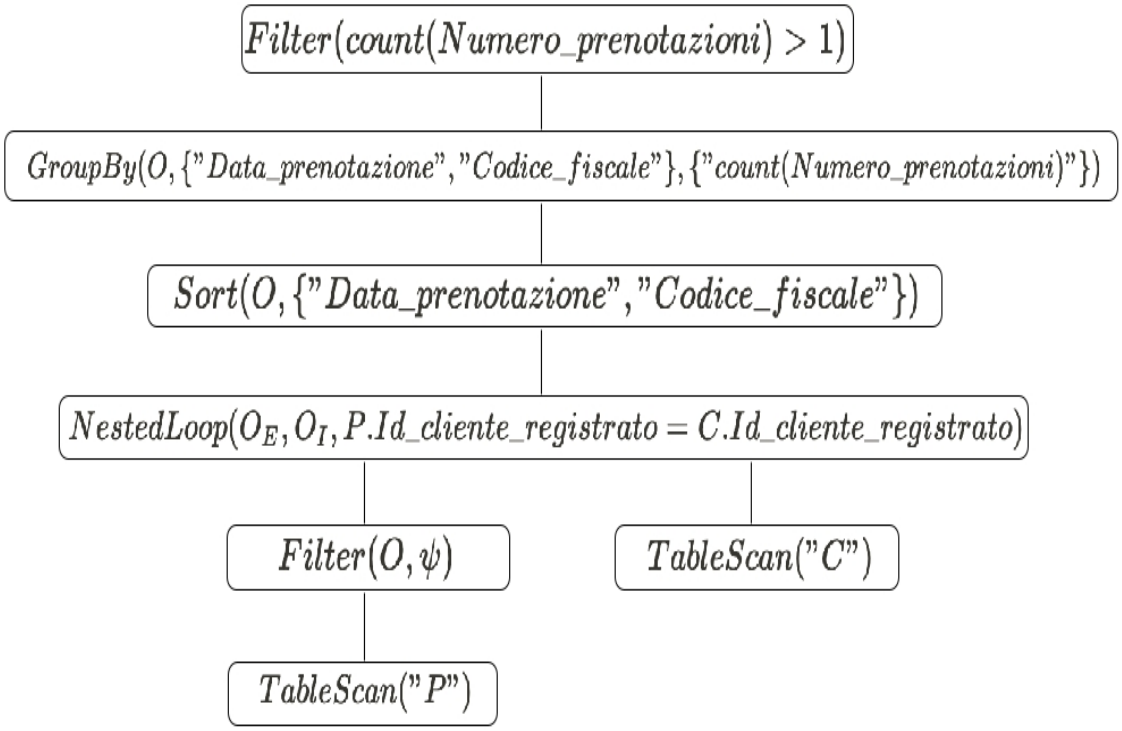
a)



b)



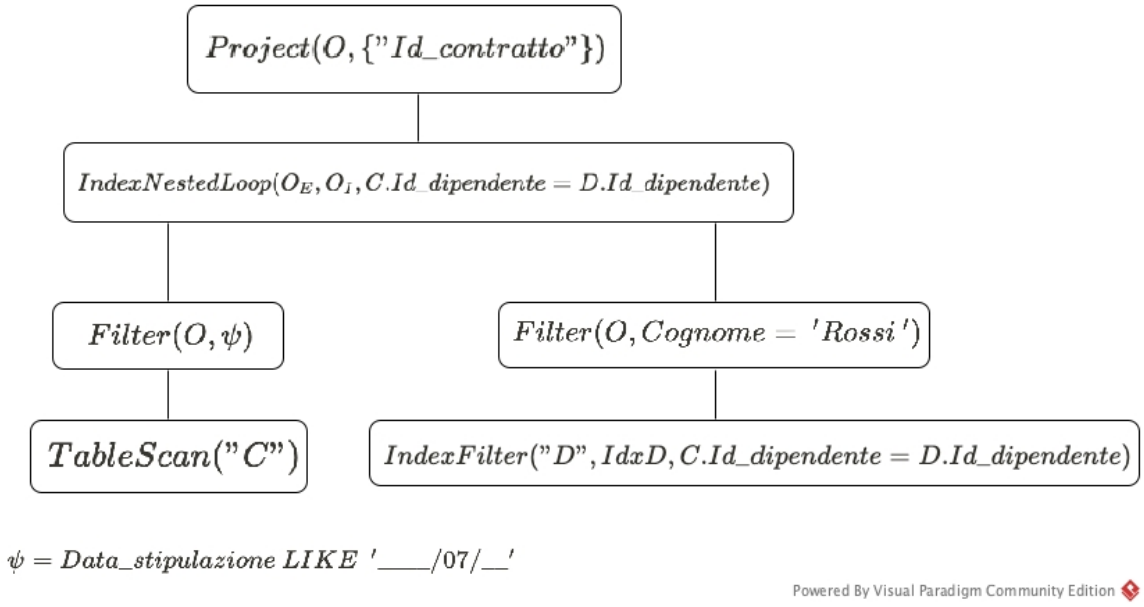
c)



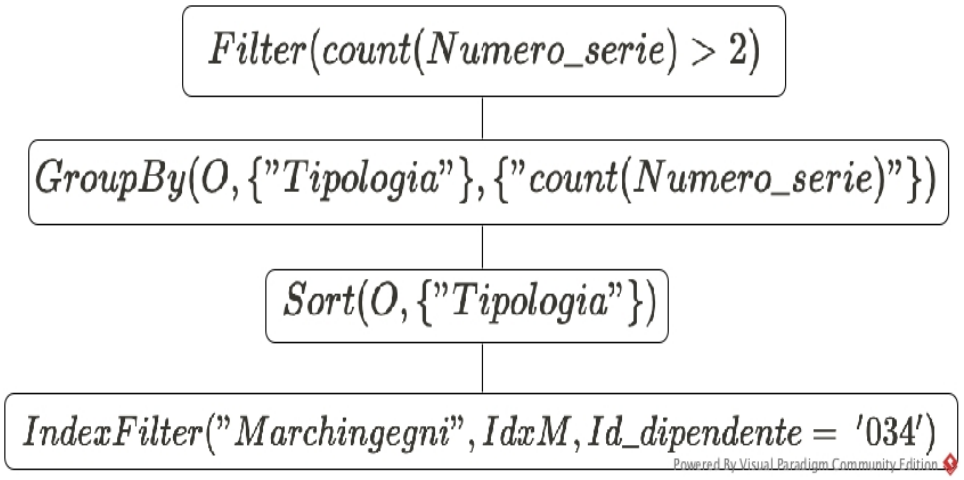
$\psi = Data\_stipulazione\ LIKE\ '2020/07/\_\_'$

III) Scrivere un piano di accesso fisico efficiente per i tre piani di accesso logico al punto I che fanno uso di due indici (o comunque del numero massimo di indici possibili), e (opzionale) verificare se la sort prima della Group By può essere evitata.

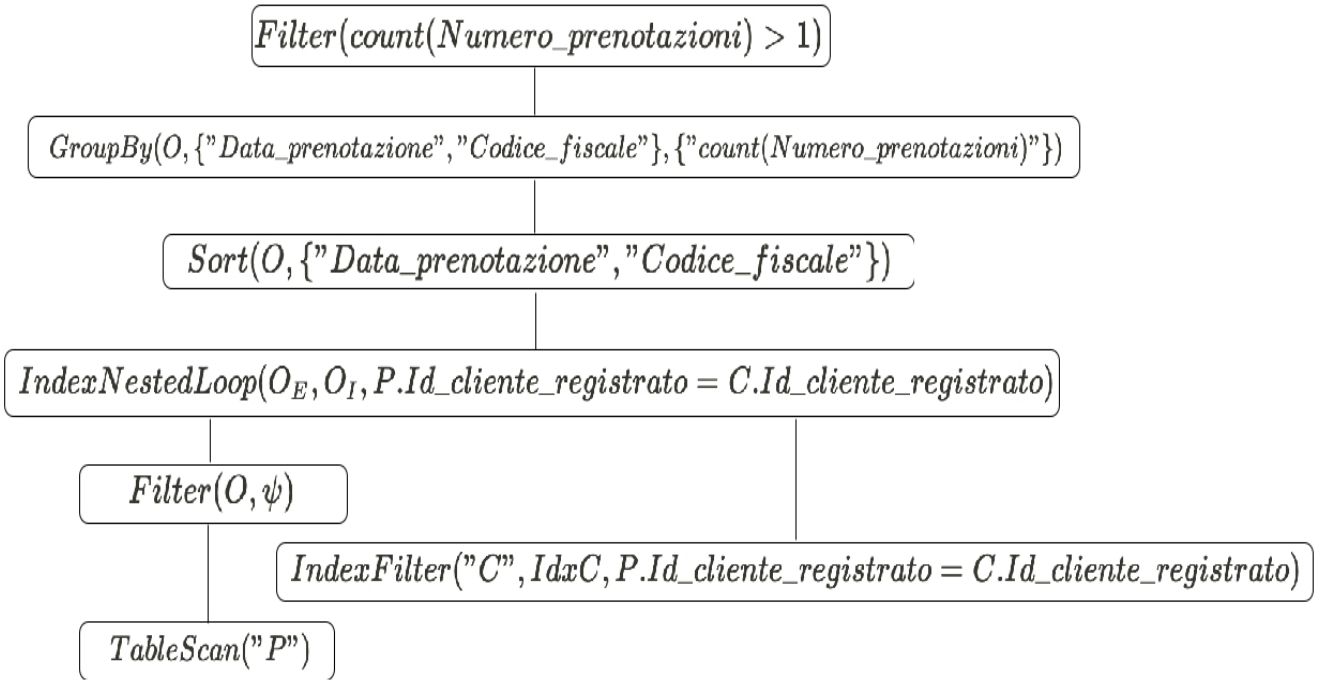
a)



b)



c)



$\psi = Data\_stipulazione\ LIKE\ '2020/07/\_\_'$