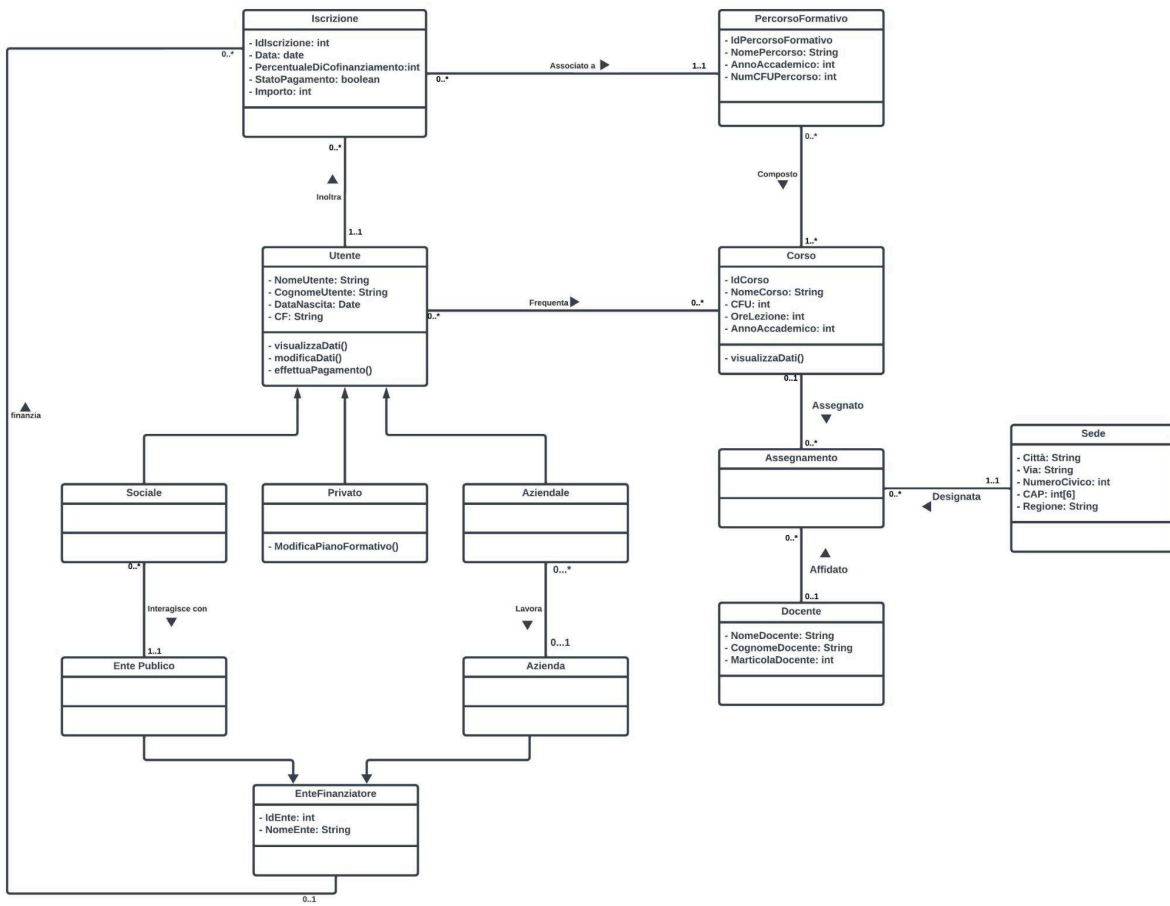


# Progetto di Sistemi Informativi

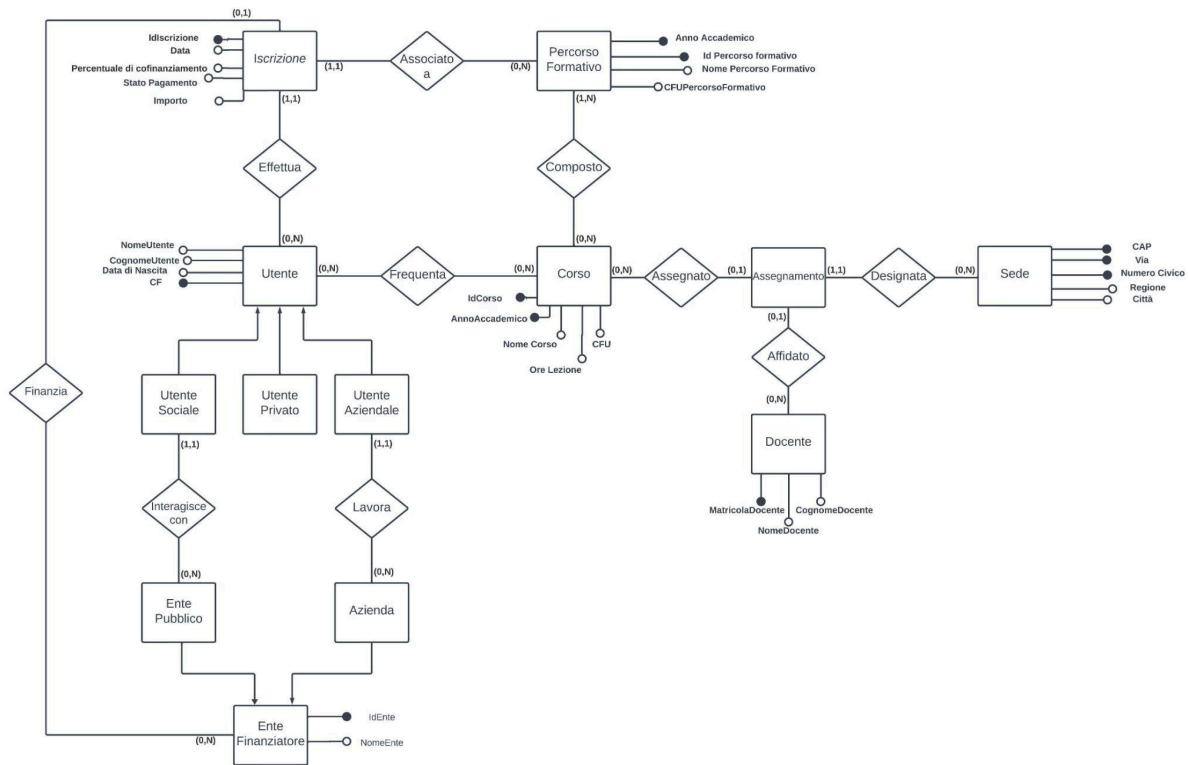
## Indice

UML	2
ER (Entity-Relationship)	3
SR (Schema Relazionale)	4
BPMN (Business Process model & Notation)	5
DFM (Dimensional Fact Model)	6
SS (Star Schema)	7
QUERY SQL OLAP	8

# UML



# ER(Entity Relationship)



# SCHEMA RELAZIONALE

**ISCRIZIONE** (IdIscrizione, Data, PercentualeDiCofinanziamento, StatoPagamento, Importo, IdPercorsoFormativo:PERCORSO FORMATIVO, AnnoAccademico:PERCORSO FORMATIVO, CF:UTENTE, idEnte\*:ENTEFINANZIATORE)

**UTENTE** (CF, NomeUtente, CongnomeUtente, DataDiNascita, IdEnte\*: ENTEFINANZIATORE)

**PERCORSO FORMATIVO**(IdPercorsoFormativo, AnnoAccademico, NomePercorso Formativo, CFUPercorsoFormativo)

**CORSO**(IdCorso, AnnoAccademico, NomeCorso, OreLezione, CFU)

**SEDE**(CAP, Via, NumeroCivico, Città, Regione)

**DOCENTE** (MatricolaDocente, NomeDocente, CognomeDocente)

**ENTE FINANZIATORE**(IdEnte, NomeEnte)

**FREQUENTA**(CF:UTENTE, IdCorso:CORSO, AnnoAccademico:CORSO)

**COMPOSTO**(IdPercorsoFormativo: PERCORSO FORMATIVO, AnnoAccademico: PERCORSO FORMATIVO, IdCorso:CORSO, AnnoAccademico:CORSO)

**ASSEGNAIMENTO**(MatricolaDocente: DOCENTE, Cap:SEDE, Via:SEDE, NumeroCivico:SEDE, IdCorso:CORSO, AnnoAccademico:CORSO)

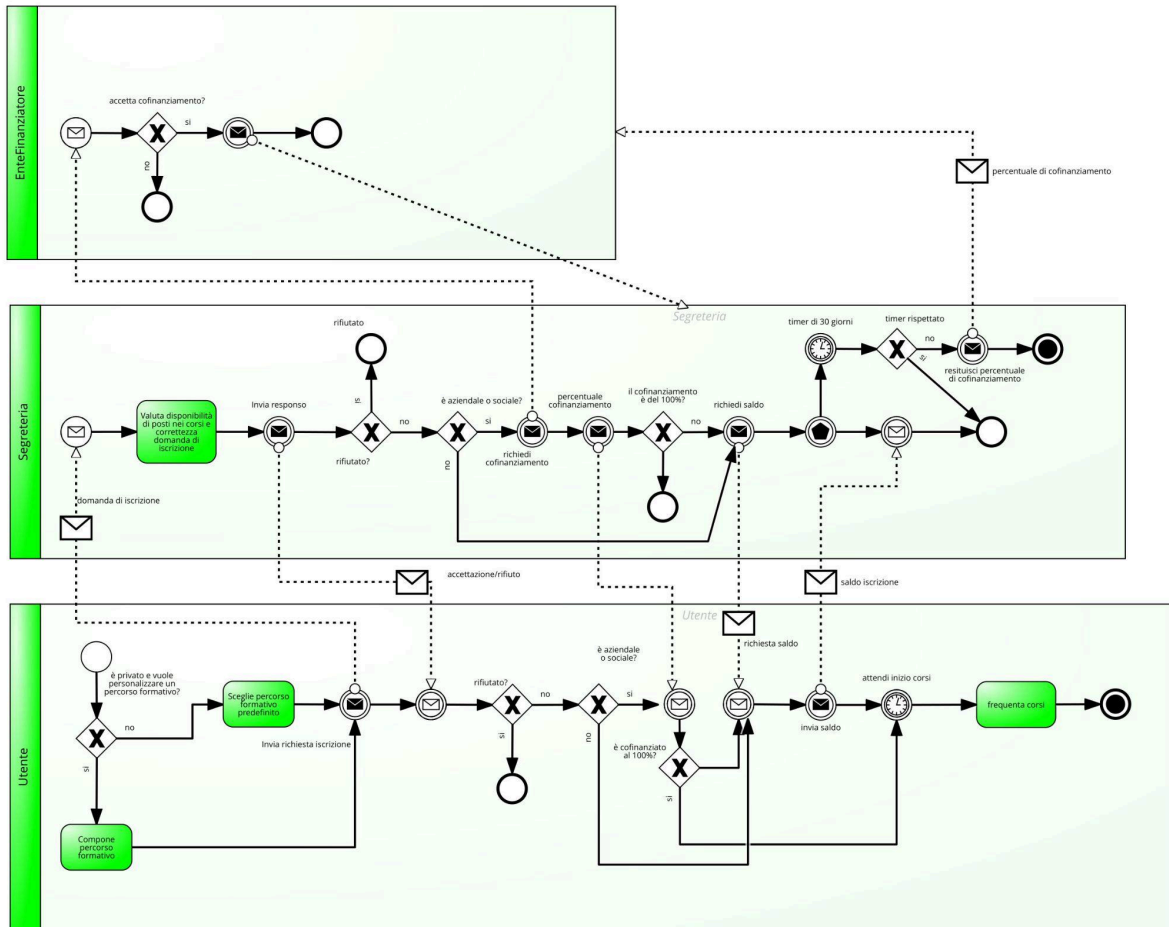
Gli attributi sottolineati sono le chiavi primarie delle tabelle

Gli attributi etichettati con l'asterisco sono annullabili

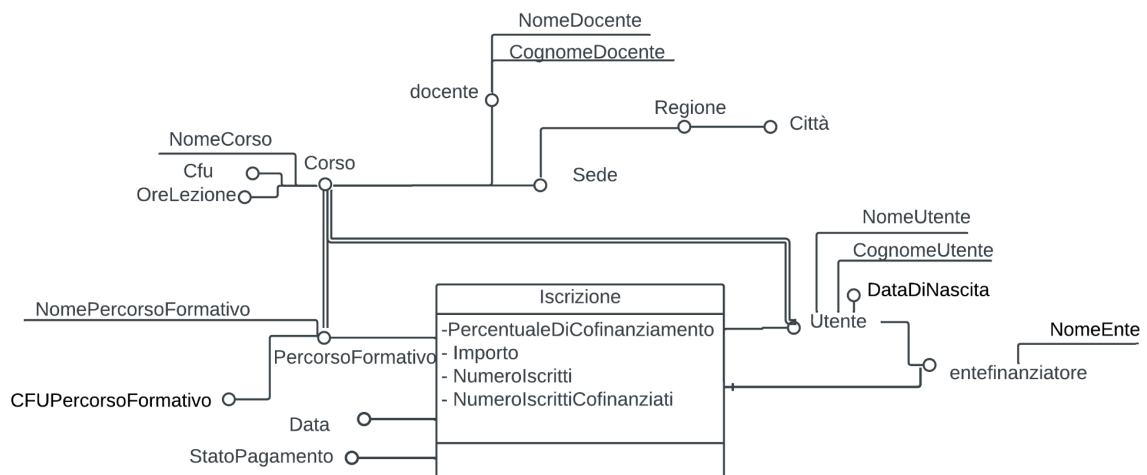
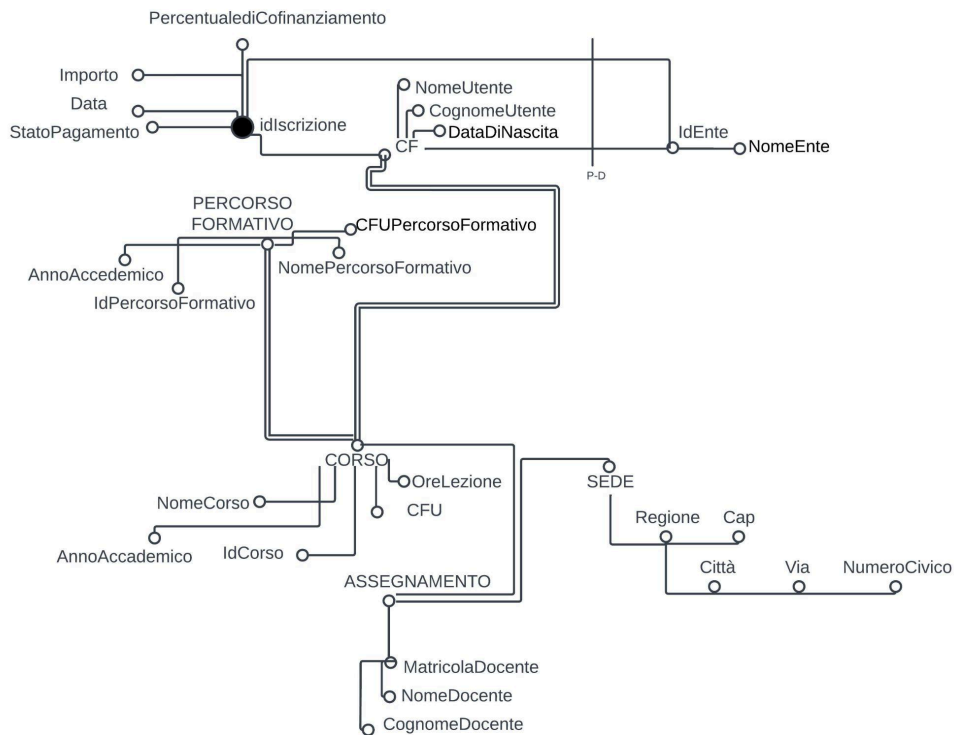
## BUSINESS RULES

In utente e iscrizione se l'Id dell'Ente non è nullo ci troviamo nel caso di un utente di tipo sociale, nel caso di un ente pubblico, oppure di un utente aziendale, nel caso di un'azienda, nel caso l'attributo sia nullo l'utente è di tipo privato quindi non viene finanziato da nessun ente, quindi la percentuale di cofinanziamento è pari a 0.

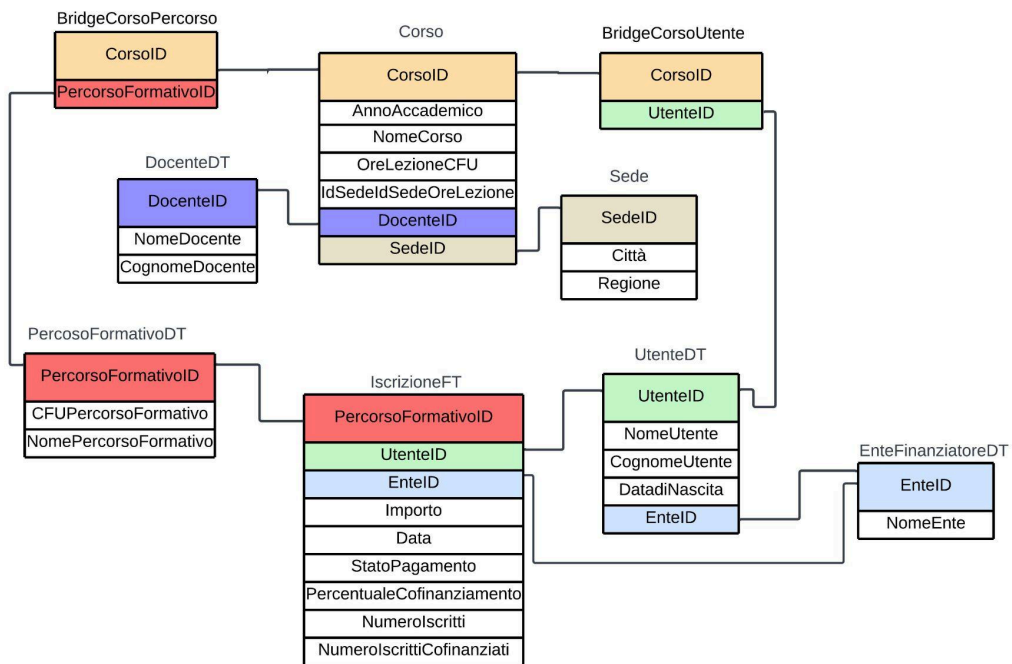
# BPMN (Business Process Model & Notation)



# DFM (Dimensional Fact Model)




# STAR SCHEMA



# QUERY

ammontare del cofinanziamento aziendale e istituzionale per trimestre nell'arco dell'ultimo anno

```
SELECT
  EXTRACT(YEAR FROM Data) AS Anno,
  EXTRACT(QUARTER FROM Data) AS Trimestre,
  ROUND(SUM(CASE WHEN StatoPagamento = 'Pagato' THEN
    (PercentualeCofinanziamento/100 * Importo) ELSE
    0 END) ,2) AS AmmontareCofinanziamento
FROM ISCRIZIONE
WHERE EXTRACT(YEAR FROM Data) = EXTRACT(YEAR FROM CURRENT_DATE)
  AND Data >= CURRENT_DATE - INTERVAL '1 YEAR'
GROUP BY (Anno, Trimestre);
```

	anno numeric 	trimestre numeric 	ammontarecofinanziamento numeric 
1	2024	1	847.50
2	2024	2	337.50
3	2024	3	1022.50
4	2024	4	0.00

numero di iscritti per insegnamento nell'a.a. corrente

```
SELECT
  C.IdCorso,
  C.AnnoAccademico,
  C.NomeCorso,
  COUNT(I.IdIscrizione) AS NumeroIscrizioni
FROM ISCRIZIONE I
JOIN FREQUENTA F ON I.CF = F.CF AND I.AnnoAccademico = F.AnnoAccademico
JOIN CORSO C ON F.IdCorso = C.IdCorso AND F.AnnoAccademico = C.AnnoAccademico
GROUP BY C.IdCorso, C.AnnoAccademico, C.NomeCorso;
```



	idcorso [PK] integer	annoaccademico [PK] character varying (10)	nomecorso character varying (50)	numeroiscrizioni bigint
1	1	2023-2024	Programmazione	4
2	2	2023-2024	Economia Aziendale	3
3	3	2023-2024	Ingegneria del Software	3

contributo (in termini di incasso) di ogni singolo insegnamento nell'ultimo a.a.

```

WITH NumeroCorsi AS (
    SELECT
        F.CF,
        COUNT(*) AS NumeroCorsiFrequentati
    FROM FREQUENTA F
    GROUP BY F.CF
)

SELECT
    C.IdCorso,
    C.AnnoAccademico,
    C.NomeCorso,
    ROUND(SUM(I.Importo / NC.NumeroCorsiFrequentati),2) AS ContributoIncassoCorso
FROM ISCRIZIONE I
JOIN FREQUENTA F ON I.CF = F.CF AND I.AnnoAccademico = F.AnnoAccademico
JOIN CORSO C ON F.IdCorso = C.IdCorso AND F.AnnoAccademico = C.AnnoAccademico
JOIN NumeroCorsi NC ON F.CF = NC.CF
WHERE EXTRACT(YEAR FROM I.Data) = EXTRACT(YEAR FROM CURRENT_DATE)
GROUP BY C.IdCorso, C.AnnoAccademico, C.NomeCorso;

```

	idcorso [PK] integer	annoaccademico [PK] character varying (10)	nomecorso character varying (50)	contributoincassocorso numeric
1	1	2023-2024	Programmazione	2500.00
2	2	2023-2024	Economia Aziendale	1650.00
3	3	2023-2024	Ingegneria del Software	1650.00

numero di insegnamenti e di ore erogate per regione nell'arco degli ultimi tre a.a.

```

SELECT
  S.Regione,
  A.IdCorso,
  C.NomeCorso,
  COUNT(DISTINCT A.IdCorso) AS NumeroInsegnamenti,
  SUM(C.OreLezione) AS OreErogate
FROM ASSEGNAmento A
JOIN SEDE S ON A.Cap = S.CAP AND A.Via = S.Via AND A.NumeroCivico = S.NumeroCivico
JOIN CORSO C ON A.IdCorso = C.IdCorso AND A.AnnoAccademico = C.AnnoAccademico
WHERE
  SUBSTRING(A.AnnoAccademico FROM 1 FOR 4)::INT > TO_CHAR(CURRENT_DATE, 'YYYY')::INT
- 3
  AND SUBSTRING(A.AnnoAccademico FROM 6 FOR 4)::INT <= TO_CHAR(CURRENT_DATE,
'YYYY')::INT
GROUP BY GROUPING SETS ((S.Regione, A.IdCorso, C.NomeCorso), (S.Regione))

```

	regione character varying (50) 🔒	idcorso integer 🔒	nomecorso character varying (50) 🔒	numeroinsegnamenti bigint 🔒	oreerogate bigint 🔒
1	Calabria	1	Programmazione	1	60
2	Calabria	[null]	[null]	1	60
3	Emilia-Romagna	3	Ingegneria del Software	1	50
4	Emilia-Romagna	[null]	[null]	1	50
5	Friuli Venezia Giulia	3	Ingegneria del Software	1	50
6	Friuli Venezia Giulia	[null]	[null]	1	50
7	Lazio	1	Programmazione	1	60
8	Lazio	[null]	[null]	1	60
9	Lombardia	2	Economia Aziendale	1	45
10	Lombardia	[null]	[null]	1	45
11	Piemonte	2	Economia Aziendale	1	45
12	Piemonte	3	Ingegneria del Software	1	50
13	Piemonte	[null]	[null]	2	95
14	Puglia	2	Economia Aziendale	1	45
15	Puglia	[null]	[null]	1	45
16	Toscana	1	Programmazione	1	60
17	Toscana	[null]	[null]	1	60
18	Veneto	1	Programmazione	1	60
19	Veneto	[null]	[null]	1	60

```
CREATE TABLE UTENTE (  
    CF VARCHAR(16) PRIMARY KEY,  
    NomeUtente VARCHAR(50),  
    CongnomeUtente VARCHAR(50),  
    DataDiNascita DATE,  
    IdEnte INT,  
    FOREIGN KEY (IdEnte) REFERENCES ENTEFINANZIATORE(IdEnte)  
);
```

```
CREATE TABLE PERCORSOFORMATIVO (  
    IdPercorsoFormativo INT,  
    AnnoAccademico VARCHAR(10),  
    NomePercorsoFormativo VARCHAR(50),  
    CFUPercorsoFormativo VARCHAR(16) REFERENCES UTENTE(CF),  
    PRIMARY KEY(IdPercorsoFormativo, AnnoAccademico)  
);
```

```
CREATE TABLE CORSO (  
    IdCorso INT,  
    AnnoAccademico VARCHAR(10),  
    NomeCorso VARCHAR(50),  
    OreLezione INT,  
    CFU INT,  
    PRIMARY KEY(IdCorso, AnnoAccademico)  
);
```

```
CREATE TABLE SEDE (  
    CAP VARCHAR(10),  
    Via VARCHAR(50),  
    NumeroCivico VARCHAR(10),  
    Città VARCHAR(50),  
    Regione VARCHAR(50),  
    PRIMARY KEY (CAP, Via, NumeroCivico)
```

);

```
CREATE TABLE DOCENTE (  
    MatricolaDocente INT PRIMARY KEY,  
    NomeDocente VARCHAR(50),  
    CognomeDocente VARCHAR(50)  
);
```

```
CREATE TABLE ISCRIZIONE (  
    IdIscrizione INT PRIMARY KEY,  
    Data DATE,  
    PercentualeDiCofinanziamento DECIMAL(5,2),  
    StatoPagamento VARCHAR(20),  
    Importo DECIMAL(10,2),  
    IdPercorsoFormativo INT,  
    AnnoAccademico VARCHAR(10),  
    CF VARCHAR(16),  
    IdEnte INT,  
    FOREIGN KEY (IdPercorsoFormativo, AnnoAccademico) REFERENCES  
PERCORSOFORMATIVO(IdPercorsoFormativo, AnnoAccademico),  
    FOREIGN KEY (CF) REFERENCES UTENTE(CF),  
    FOREIGN KEY (IdEnte) REFERENCES ENTEFINANZIATORE(IdEnte)  
);
```

```
CREATE TABLE FREQUENTA (  
    CF VARCHAR(16),  
    IdCorso INT,  
    AnnoAccademico VARCHAR(10),  
    PRIMARY KEY (CF, IdCorso, AnnoAccademico),  
    FOREIGN KEY (CF) REFERENCES UTENTE(CF),  
    FOREIGN KEY (IdCorso, AnnoAccademico) REFERENCES CORSO(IdCorso,  
AnnoAccademico)  
);
```

```
CREATE TABLE COMPOSTO (  
    IdPercorsoFormativo INT,  
    AnnoAccademicoPF VARCHAR(10),  
    IdCorso INT,  
    AnnoAccademicoC VARCHAR(10),  
    PRIMARY KEY (IdPercorsoFormativo, AnnoAccademicoPF, IdCorso, AnnoAccademicoC),  
    FOREIGN KEY (IdPercorsoFormativo, AnnoAccademicoPF) REFERENCES  
PERCORSOFORMATIVO(IdPercorsoFormativo, AnnoAccademico),  
    FOREIGN KEY (IdCorso, AnnoAccademicoC) REFERENCES CORSO(IdCorso,  
AnnoAccademico)  
);
```

```
CREATE TABLE ASSEGNAIMENTO (  
    MatricolaDocente INT,
```

Cap VARCHAR(10),  
Via VARCHAR(50),  
NumeroCivico VARCHAR(10),

```
INSERT INTO ENTEFINANZIATORE (IdEnte, NomeEnte)
VALUES
(1, 'Ente A'),
(2, 'Ente B'),
(3, 'Ente C');
```

-- Inserimento di 10 tuple di sedi

```
INSERT INTO SEDE (CAP, Via, NumeroCivico, Città, Regione)
VALUES
('00100', 'Via Roma', '1', 'Roma', 'Lazio'),
('20121', 'Via della Moscova', '10', 'Milano', 'Lombardia'),
('40126', 'Via San Vitale', '15', 'Bologna', 'Emilia-Romagna'),
('50123', 'Via de Tornabuoni', '20', 'Firenze', 'Toscana'),
('70122', 'Via Sparano', '25', 'Bari', 'Puglia'),
('10123', 'Piazza Castello', '30', 'Torino', 'Piemonte'),
('30174', 'Corso del Popolo', '35', 'Venezia', 'Veneto'),
('12100', 'Via Po', '40', 'Cuneo', 'Piemonte'),
('34100', 'Piazza del Duomo', '45', 'Trieste', 'Friuli Venezia Giulia'),
('88100', 'Corso Garibaldi', '50', 'Catanzaro', 'Calabria');
```

-- Inserimento di 10 tuple di docenti

```
INSERT INTO DOCENTE (MatricolaDocente, NomeDocente, CognomeDocente)
VALUES
(1001, 'Professore', 'Uno'),
(1002, 'Professoressa', 'Due'),
(1003, 'Professore', 'Tre'),
(1004, 'Professoressa', 'Quattro'),
(1005, 'Professore', 'Cinque'),
(1006, 'Professoressa', 'Sei'),
(1007, 'Professore', 'Sette'),
(1008, 'Professoressa', 'Otto'),
(1009, 'Professore', 'Nove'),
(1010, 'Professoressa', 'Dieci');
```

-- Inserimento di 10 tuple di utenti

```
INSERT INTO UTENTE (CF, NomeUtente, CognomeUtente, DataDiNascita, IdEnte)
VALUES
('CF1000000001', 'Mario', 'Rossi', '1990-01-15', 1),
('CF1000000002', 'Laura', 'Bianchi', '1992-02-20', 2),
('CF1000000003', 'Luigi', 'Verdi', '1988-03-10', 3),
```

```

('CF1000000004', 'Sara', 'Neri', '1995-04-05', 1),
('CF1000000005', 'Giovanni', 'Gialli', '1993-05-15', 2),
('CF1000000006', 'Paola', 'Rosa', '1991-06-20', 3),
('CF1000000007', 'Marco', 'Azzurri', '1987-07-10', 1),
('CF1000000008', 'Elena', 'Marroni', '1994-08-05', 2),
('CF1000000009', 'Andrea', 'Arancioni', '1989-09-15', 3),
('CF1000000010', 'Valentina', 'Viola', '1996-10-20', 1);

```

-- Inserimento di 10 tuple di percorsi formativi

```

INSERT INTO PERCORSOFORMATIVO (IdPercorsoFormativo, AnnoAccademico,
NomePercorsoFormativo, CFUPercorsoFormativo)

```

VALUES

```

(1, '2023-2024', 'Informatica', 'CF1000000001'),
(2, '2023-2024', 'Economia', 'CF1000000002'),
(3, '2023-2024', 'Ingegneria', 'CF1000000003'),
(4, '2023-2024', 'Informatica', 'CF1000000004'),
(5, '2023-2024', 'Economia', 'CF1000000005'),
(6, '2023-2024', 'Ingegneria', 'CF1000000006'),
(7, '2023-2024', 'Informatica', 'CF1000000007'),
(8, '2023-2024', 'Economia', 'CF1000000008'),
(9, '2023-2024', 'Ingegneria', 'CF1000000009'),
(10, '2023-2024', 'Informatica', 'CF1000000010');

```

-- Inserimento di 10 tuple di corsi

```

INSERT INTO CORSO (IdCorso, AnnoAccademico, NomeCorso, OreLezione, CFU)

```

VALUES

```

(1, '2023-2024', 'Programmazione', 60, 6),
(2, '2023-2024', 'Economia Aziendale', 45, 4),
(3, '2023-2024', 'Ingegneria del Software', 50, 5),
(4, '2023-2024', 'Basi di Dati', 48, 5),
(5, '2023-2024', 'Marketing', 42, 4),
(6, '2023-2024', 'Reti di Calcolatori', 54, 6),
(7, '2023-2024', 'Sicurezza Informatica', 45, 5),
(8, '2023-2024', 'Finanza Aziendale', 48, 4),
(9, '2023-2024', 'Progettazione Architetture Software', 60, 6),
(10, '2023-2024', 'Sistemi Operativi', 54, 5);

```

-- Inserimento di 10 tuple di iscrizione con stato di pagamento "Pagato" o "In attesa"

```

INSERT INTO ISCRIZIONE (IdIscrizione, Data, PercentualeDiCofinanziamento, StatoPagamento,
Importo, IdPercorsoFormativo, AnnoAccademico, CF, IdEnte)

```

VALUES

```

(1010, '2024-01-15', 85.00, 'Pagato', 550.00, 1, '2023-2024', 'CF1000000001', 1),
(1011, '2024-02-20', 70.00, 'In attesa', 700.00, 2, '2023-2024', 'CF1000000002', 2),
(1012, '2024-03-10', 95.00, 'Pagato', 400.00, 3, '2023-2024', 'CF1000000003', 3),
(1013, '2024-04-05', 80.00, 'In attesa', 600.00, 1, '2023-2024', 'CF1000000004', 1),
(1014, '2024-05-15', 75.00, 'Pagato', 450.00, 2, '2023-2024', 'CF1000000005', 2),
(1015, '2024-06-20', 90.00, 'In attesa', 800.00, 3, '2023-2024', 'CF1000000006', 3),
(1016, '2024-07-10', 85.00, 'Pagato', 700.00, 1, '2023-2024', 'CF1000000007', 1),

```

```
(1017, '2024-08-05', 70.00, 'In attesa', 500.00, 2, '2023-2024', 'CF1000000008', 2),
(1018, '2024-09-15', 95.00, 'Pagato', 450.00, 3, '2023-2024', 'CF1000000009', 3),
(1019, '2024-10-20', 80.00, 'In attesa', 650.00, 1, '2023-2024', 'CF1000000010', 1);
```

-- Inserimento di 10 tuple di frequenza

```
INSERT INTO FREQUENTA (CF, IdCorso, AnnoAccademico)
VALUES
('CF1000000001', 1, '2023-2024'),
('CF1000000002', 2, '2023-2024'),
('CF1000000003', 3, '2023-2024'),
('CF1000000004', 1, '2023-2024'),
('CF1000000005', 2, '2023-2024'),
('CF1000000006', 3, '2023-2024'),
('CF1000000007', 1, '2023-2024'),
('CF1000000008', 2, '2023-2024'),
('CF1000000009', 3, '2023-2024'),
('CF1000000010', 1, '2023-2024');
```

-- Inserimento di 10 tuple di composizione

```
INSERT INTO COMPOSTO (IdPercorsoFormativo, AnnoAccademicoPF, IdCorso, AnnoAccademicoC)
VALUES
(1, '2023-2024', 1, '2023-2024'),
(1, '2023-2024', 2, '2023-2024'),
(1, '2023-2024', 3, '2023-2024'),
(2, '2023-2024', 1, '2023-2024'),
(2, '2023-2024', 2, '2023-2024'),
(2, '2023-2024', 3, '2023-2024'),
(3, '2023-2024', 1, '2023-2024'),
(3, '2023-2024', 2, '2023-2024'),
(3, '2023-2024', 3, '2023-2024'),
(4, '2023-2024', 1, '2023-2024');
```

-- Inserimento di 10 tuple di assegnamento

```
INSERT INTO ASSEGNAMENTO (MatricolaDocente, Cap, Via, NumeroCivico, IdCorso,
AnnoAccademico)
VALUES
(1001, '00100', 'Via Roma', '1', 1, '2023-2024'),
(1002, '20121', 'Via della Moscova', '10', 2, '2023-2024'),
(1003, '40126', 'Via San Vitale', '15', 3, '2023-2024'),
(1004, '50123', 'Via de Tornabuoni', '20', 1, '2023-2024'),
(1005, '70122', 'Via Sparano', '25', 2, '2023-2024'),
(1006, '10123', 'Piazza Castello', '30', 3, '2023-2024'),
(1007, '30174', 'Corso del Popolo', '35', 1, '2023-2024'),
(1008, '12100', 'Via Po', '40', 2, '2023-2024'),
(1009, '34100', 'Piazza del Duomo', '45', 3, '2023-2024'),
(1010, '88100', 'Corso Garibaldi', '50', 1, '2023-2024');
```