

```

CREATE TABLE ruolo (
    id_ruolo SERIAL PRIMARY KEY,          -- ID univoco per ogni ruolo
    nome_ruolo VARCHAR(50) UNIQUE         -- Nome del ruolo, es. 'Amministratore', 'Gestore', 'Agente', etc.
);

CREATE TABLE utente (
    username_utente VARCHAR(15) NOT NULL PRIMARY KEY, -- Definisce username_utente come chiave primaria
    password VARCHAR(15) NOT NULL,
    id_ruolo INT NOT NULL, -- Colonna che rappresenta il ruolo dell'utente (riferimento alla tabella ruolo_utente)
    nome VARCHAR(20) NOT NULL,
    cognome VARCHAR(20) NOT NULL,
    email VARCHAR(255) NOT NULL,
    FOREIGN KEY (id_ruolo) REFERENCES ruolo(id_ruolo) ON DELETE CASCADE -- Legame con la tabella dei ruoli
);

CREATE TABLE indirizzo (
    id_indirizzo SERIAL PRIMARY KEY,
    città VARCHAR(255),
    provincia VARCHAR(255),
    via VARCHAR(255),
    cap VARCHAR(10)
);

CREATE TABLE filtro_avanzato (
    id_filtro_avanzato SERIAL PRIMARY KEY,
    tipologia_immobile VARCHAR(255),
    stanza INT CHECK (stanza >= 0),
    piano INT CHECK (piano >= 0),
    bagno INT CHECK (bagno >= 0),
    parcheggio VARCHAR(255),
    classe_energetica VARCHAR(10)
);

CREATE TABLE servizio_ulteriore (
    id_servizio_ulteriore SERIAL PRIMARY KEY,
    climatizzatore BOOLEAN,
    balcone BOOLEAN,
    portineria BOOLEAN,
    giardino BOOLEAN,
    ascensore BOOLEAN,
    arredato BOOLEAN,
    id_filtro_avanzato INT,
    FOREIGN KEY (id_filtro_avanzato) REFERENCES filtro_avanzato(id_filtro_avanzato) ON DELETE CASCADE
);

CREATE TABLE immobile (
    id_immobile SERIAL PRIMARY KEY,
    data_creazione DATE DEFAULT CURRENT_DATE,
    ora_creazione TIME DEFAULT CURRENT_TIME,
    username_agente VARCHAR(15), -- Username dell'agente che gestisce l'immobile
    tipo_contratto VARCHAR(20) CHECK (tipo_contratto IN ('Vendita', 'Affitto')),
    tipologia_immobile VARCHAR(255),
    titolo VARCHAR(255),
    testo TEXT,
    superficie DOUBLE PRECISION CHECK (superficie > 0), -- Superficie in metri quadrati
    prezzo DOUBLE PRECISION CHECK (prezzo > 0), -- Prezzo dell'immobile
    id_indirizzo_immobile INT, -- Riferimento alla tabella INDIRIZZO
    id_filtro_avanzato INT,
    id_servizio_ulteriore INT,
    FOREIGN KEY (username_agente) REFERENCES utente(username_utente) ON DELETE CASCADE,
    FOREIGN KEY (id_indirizzo_immobile) REFERENCES indirizzo(id_indirizzo) ON DELETE CASCADE,
    FOREIGN KEY (id_filtro_avanzato) REFERENCES filtro_avanzato(id_filtro_avanzato) ON DELETE CASCADE,
    FOREIGN KEY (id_servizio_ulteriore) REFERENCES servizio_ulteriore(id_servizio_ulteriore) ON DELETE CASCADE
);

CREATE TABLE foto (
    id_foto SERIAL PRIMARY KEY,
    id_immobile INT NOT NULL,
    percorso_file TEXT NOT NULL, -- Salviamo l'URL completo o il percorso
    ordine INT DEFAULT 1, -- Per gestire l'ordine delle foto
    FOREIGN KEY (id_immobile) REFERENCES immobile(id_immobile) ON DELETE CASCADE
);

CREATE TABLE modifica (
    id_modifica SERIAL PRIMARY KEY, -- ID univoco per ogni modifica
    id_immobile INT NOT NULL, -- ID dell'immobile che è stato modificato
    username_utente VARCHAR(15) NOT NULL, -- Username dell'utente che ha effettuato la modifica
    data_modifica DATE DEFAULT CURRENT_DATE, -- Data della modifica
    ora_modifica TIME DEFAULT CURRENT_TIME, -- Ora della modifica
    nuovo_tipo_contratto VARCHAR(20), -- Nuovo tipo di contratto (vendita/affitto)
    nuova_tipologia_immobile VARCHAR(255), -- Nuova tipologia dell'immobile
    nuovo_titolo VARCHAR(255), -- Nuovo titolo dell'immobile
    nuovo_testo TEXT, -- Nuovo testo descrittivo dell'immobile
    nuova_superficie DOUBLE PRECISION CHECK (nuova_superficie > 0), -- Nuova superficie
    nuovo_prezzo DOUBLE PRECISION CHECK (nuovo_prezzo > 0), -- Nuovo prezzo
    nuovo_id_indirizzo_immobile INT, -- Nuovo indirizzo dell'immobile
    nuovo_id_filtro_avanzato INT, -- Nuovo filtro avanzato applicato
    nuovo_id_servizio_ulteriore INT, -- Nuovo servizio aggiuntivo applicato
    FOREIGN KEY (id_immobile) REFERENCES immobile(id_immobile) ON DELETE CASCADE,
    FOREIGN KEY (username_utente) REFERENCES utente(username_utente) ON DELETE CASCADE,
    FOREIGN KEY (nuovo_id_indirizzo_immobile) REFERENCES indirizzo(id_indirizzo) ON DELETE CASCADE,
    FOREIGN KEY (nuovo_id_filtro_avanzato) REFERENCES filtro_avanzato(id_filtro_avanzato) ON DELETE CASCADE,
    FOREIGN KEY (nuovo_id_servizio_ulteriore) REFERENCES servizio_ulteriore(id_servizio_ulteriore) ON DELETE CASCADE
);

```

```

CREATE TABLE proposta (
    id_proposta SERIAL PRIMARY KEY,          -- ID della proposta, chiave primaria
    id_immobile_proposta INT,                -- Prezzo precedente
    vecchio_prezzo DOUBLE PRECISION,         -- Nuovo prezzo proposto
    nuovo_prezzo DOUBLE PRECISION,           -- Stato della proposta (ad esempio, 'accettata', 'in attesa', ecc.)
    stato_proposta VARCHAR(50),              -- Data della proposta
    data_proposta DATE,                      -- Ora della proposta
    ora_proposta TIME,                       -- Username dell'utente che ha fatto la proposta
    username_utente_proposta VARCHAR(15),    -- Username dell'agente che fa la controproposta (se presente)
    username_agente_controproposta VARCHAR(15), -- Prezzo della controproposta fatta dall'agente (se presente)
    controproposta DOUBLE PRECISION,         -- Stato della controproposta (ad esempio 'accettata', 'rifiutata')
    stato_controproposta VARCHAR(50),
    FOREIGN KEY (id_immobile_proposta) REFERENCES immobile(id_immobile) ON DELETE CASCADE,
    FOREIGN KEY (username_utente_proposta) REFERENCES utente(username_utente) ON DELETE CASCADE,
    FOREIGN KEY (username_agente_controproposta) REFERENCES utente(username_utente) ON DELETE CASCADE
);

CREATE TABLE visita (
    id_visita SERIAL PRIMARY KEY,
    id_immobile INT,
    username_utente VARCHAR(15),
    data_visita DATE,
    ora_visita TIME,
    stato_visita VARCHAR(255),                -- Stato della visita (es. 'in attesa', 'completata', ecc.)
    stato_approvazione_agente VARCHAR(50),    -- Stato dell'approvazione dell'agente (es. 'accettata', 'rifiutata', 'in attesa')
    username_agente_approvazione VARCHAR(15), -- Username dell'agente che ha approvato o rifiutato l'appuntamento
    FOREIGN KEY (id_immobile) REFERENCES immobile(id_immobile) ON DELETE CASCADE,
    FOREIGN KEY (username_utente) REFERENCES utente(username_utente) ON DELETE CASCADE,
    FOREIGN KEY (username_agente_approvazione) REFERENCES utente(username_utente) ON DELETE CASCADE
);

CREATE TABLE ricerca (
    id_ricerca SERIAL PRIMARY KEY,
    id_indirizzo INT,
    id_filtro_avanzato INT,
    FOREIGN KEY (id_indirizzo) REFERENCES indirizzo(id_indirizzo) ON DELETE CASCADE,
    FOREIGN KEY (id_filtro_avanzato) REFERENCES filtro_avanzato(id_filtro_avanzato) ON DELETE CASCADE
);

```