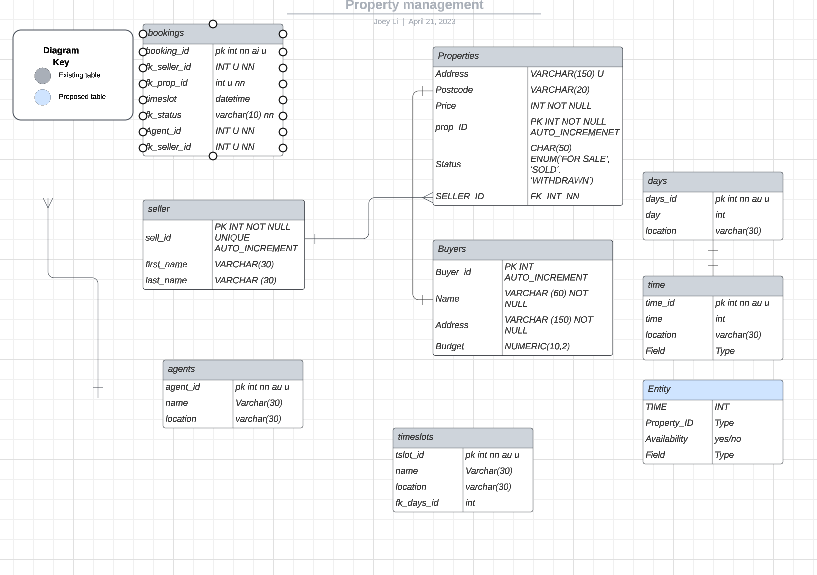
**EstateAgents Database Project**

***Firstly, I created the database ERD to ensure the correct amount of tables and data required would be present. This also helped me to visualise the relationships I would need to create between tables to query the data specific to the users requirements.***



***Once the schema was created, I then went on to create my database and the tables within. I placed this all into one query and then copy and pasted them individually into another query dependent on the foreign keys the table required (bookings was last due to requiring the most foreign keys)***

USE EstateAgents;

CREATE TABLE seller (

sell\_id INT UNIQUE PRIMARY KEY AUTO\_INCREMENT,

first\_name VARCHAR(20),

last\_name Varchar(30)

);

CREATE TABLE buyer (

buyer\_id INT UNIQUE PRIMARY KEY AUTO\_INCREMENT,

name VARCHAR(60) NOT NULL,

budget NUMERIC(10,2)

);

CREATE TABLE agents (

agent\_id INT UNIQUE PRIMARY KEY AUTO\_INCREMENT,

name VARCHAR(60) NOT NULL,

location VARCHAR(30) NOT NULL

);

CREATE TABLE properties (

prop\_id INT UNIQUE PRIMARY KEY AUTO\_INCREMENT,

p\_seller\_id INT NOT NULL,

address VARCHAR(120) UNIQUE,

postcode VARCHAR(10),

price INT NOT NULL,

status ENUM('FOR SALE', 'SOLD', 'WITHDRAWN'),

CONSTRAINT fk\_seller\_id

FOREIGN KEY (fk\_seller\_id) REFERENCES seller(seller\_id))

;

CREATE TABLE bookings (

book\_id INT UNIQUE PRIMARY KEY AUTO\_INCREMENT,

fk\_seller\_id INT IS NOT NULL,

fk\_prop\_id INT IS NOT NULL,

fk\_buyer\_id INT IS NOT NULL,

timeslot DATETIME,

fk\_agent\_id INT IS NOT NULL,

CONSTRAINT fk\_seller\_id

FOREIGN KEY (seller\_id) REFERENCES seller(seller\_id),

CONSTRAINT fk\_prop\_id

FOREIGN KEY (prop\_id) REFERENCES properties(prop\_id),

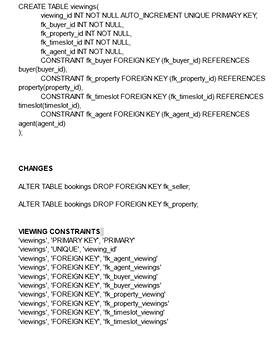
CONSTRAINT fk\_buyer\_id

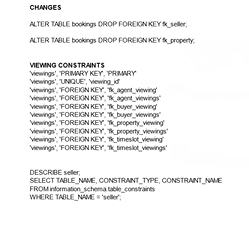
FOREIGN KEY (buyer\_id) REFERENCES buyer(buyer\_id),

CONSTRAINT fk\_agent\_id

FOREIGN KEY (agent\_id) REFERENCES agents(agent\_id)

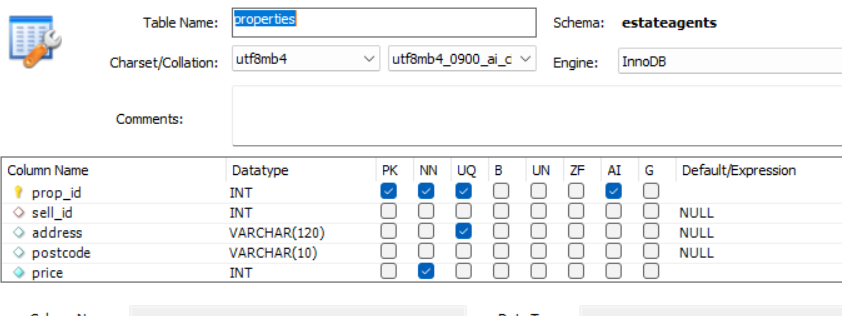
);

****

****

**CASE STUDY**

**Manage Property**

****

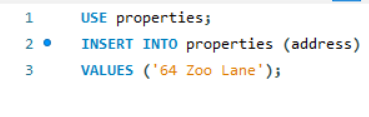
**1.A property can have the following status: FOR SALE, SOLD WITHDRAWN**

****

*I have used the enum datatype with specific values.*

**Add Property**

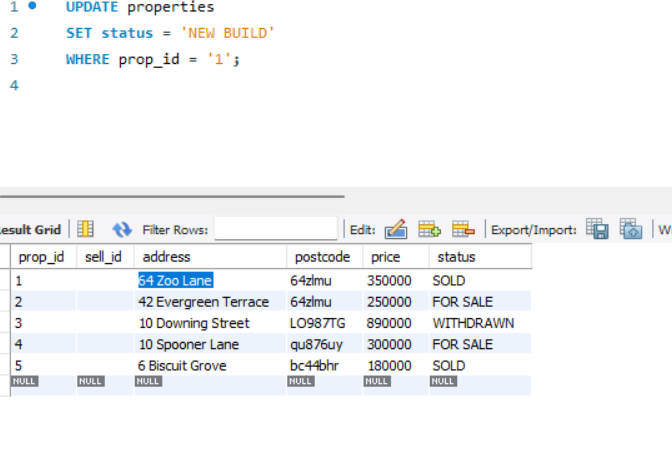
**2.**

****

****

*I have given the property address a unique constraint to ensure no duplicates of property address*

***3.***

**

**

***PROCEDURE CODE FOR ADDING PROPERTIES, MAKING SURE THEY ARE CHECKED AND UPDATED ON THE PROPERTY TABLE.***

*USE estateagents;*

*DELIMITER //*

*CREATE PROCEDURE add\_property(IN p\_address VARCHAR(255), IN p\_postcode VARCHAR(9), IN p\_seller\_id INT)*

*BEGIN*

*DECLARE address\_exists INT;*

*SELECT COUNT(\*) INTO address\_exists*

*FROM properties*

*WHERE address = p\_address AND postcode = p\_postcode;*

*IF address\_exists = 0 THEN*

*INSERT INTO properties (address, postcode, seller\_id)*

*VALUES (p\_address, p\_postcode, p\_seller\_id);*

*ELSE*

*SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'The property with this address already exists.';*

*END IF;*

*END //*

*DELIMITER ;*

***PROCEDURE CODE FOR SEARCHING PROPERTIES BY POSTCODE AND MIN/MAX PRICE***

*USE estateagents;*

*DELIMITER //*

*CREATE PROCEDURE search\_prop(*

*IN p\_postcode VARCHAR(9),*

*IN p\_price INT,*

*IN p\_min\_price INT,*

*IN p\_max\_price INT*

*)*

*BEGIN*

*DECLARE prop\_search INT;*

*SELECT COUNT(\*) INTO prop\_search*

*FROM properties*

*WHERE postcode = p\_postcode*

*AND price BETWEEN p\_min\_price AND p\_max\_price;*

*IF prop\_search = 0 THEN*

*SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'There are no properties that fit your requirements';*

*ELSE*

*SELECT \**

*FROM properties*

*WHERE postcode = p\_postcode*

*AND price BETWEEN p\_min\_price AND p\_max\_price;*

*END IF;*

*END //*

*DELIMITER ;*

***Procedure for withdrawing a property that is not sold***