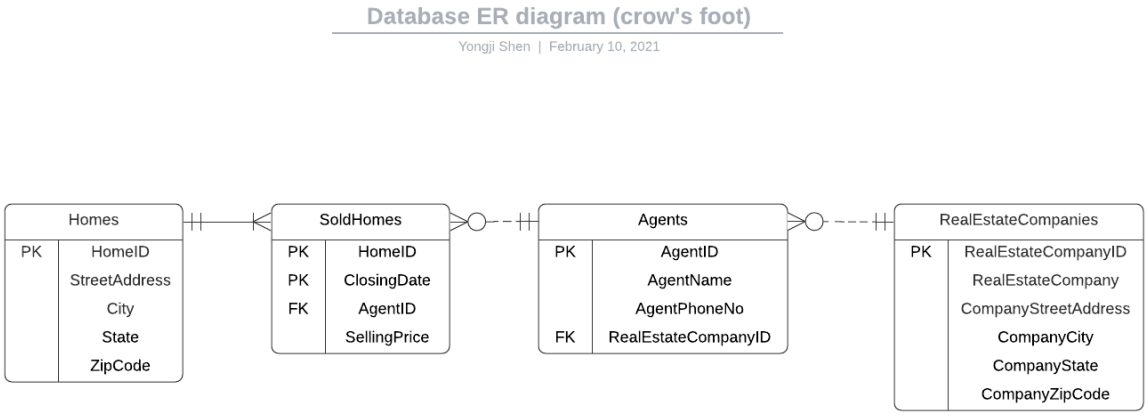
Yongji Shen

001531346

INFO 6210

Lab1

Part 1



Part 2

In Design B, because each table has non-identifying relationship, when a row of data inserts into Enrollment table, we have to ensure that the StudentID, CourseID, and TermID are both validated (should not be NULL, because if CourseID is NULL which does not make sense that a student can register a course which not exists, same idea for StudentID and TermID). At the same time, we also have to check the parent entities to make sure the parent entities have exactly same value, such as when the Enrollment table has a data like [001, INFO6210, Spring2020], we also need to look up the StudentID table to ensure that a student has 001 as his/her student ID, same for CourseID, and TermID. Finally, we need to identify the primary key in the parent entities is unique and the combination of the value of foreign key in the child table is unique, such as EnrollmentID is the primary key in the Enrollment table and searching by EnrollmentID should identify one and only one record, same idea for StudentID, TermID and CourseID.
As long as we can ensure the above requirements, we can say that the data integrity maintained in design A is still enforced in design B

Part 3

Code to insert all data into MongoDB

```
JS Lab1-3.js > ...
1   const MongoClient = require('mongodb').MongoClient;
2   const uri = "mongodb+srv://dbYS:11111@cluster0.frvi4.mongodb.net/Lab1-Part3?retryWrites=true&w=majority";
3   const client = new MongoClient(uri, { useNewUrlParser: true,useUnifiedTopology: true });
4
5   var list = [
6       {SalesPersonsID: 274, LastName: "Jiang",FirstName:"Stephen",TotalOrderCount: 48},
7       {SalesPersonsID: 275, LastName: "Blythe",FirstName:"Michael",TotalOrderCount: 450},
8       {SalesPersonsID: 276, LastName: "Mitchell",FirstName:"Linda",TotalOrderCount: 418},
9       {SalesPersonsID: 277, LastName: "Carson",FirstName:"Jillian",TotalOrderCount: 473},
10      {SalesPersonsID: 278, LastName: "Vargas",FirstName:"Garrett",TotalOrderCount: 234},
11      {SalesPersonsID: 279, LastName: "Reiter",FirstName:"Tsvi",TotalOrderCount: 429},
12      {SalesPersonsID: 280, LastName: "Ansman-Wolfe",FirstName:"Pamela",TotalOrderCount: 95},
13  ];
14  client.connect(err => {
15    const collection = client.db("Lab1")
16      collection.collection("Order").insertMany(list,function (err,res){
17          if (err) throw err;
18          console.log("inserted");
19          client.close();
20        });
21  });
```

Calculate the average of Total Order Count