Lab 1

Exercise 1

I decided to give the Toys a type to accommodate for different toys that can exist. I also gave them optional attributes in the form of Attribute1, Attrubte2 etc. This way different type of Toys can be added to one table and they can have maximum of 3 attributes which can be different or similar or do not need to be added.

The table would be very easy to maintain also very efficient considering it's only 1 table that'll needed to be updated instead of many more.

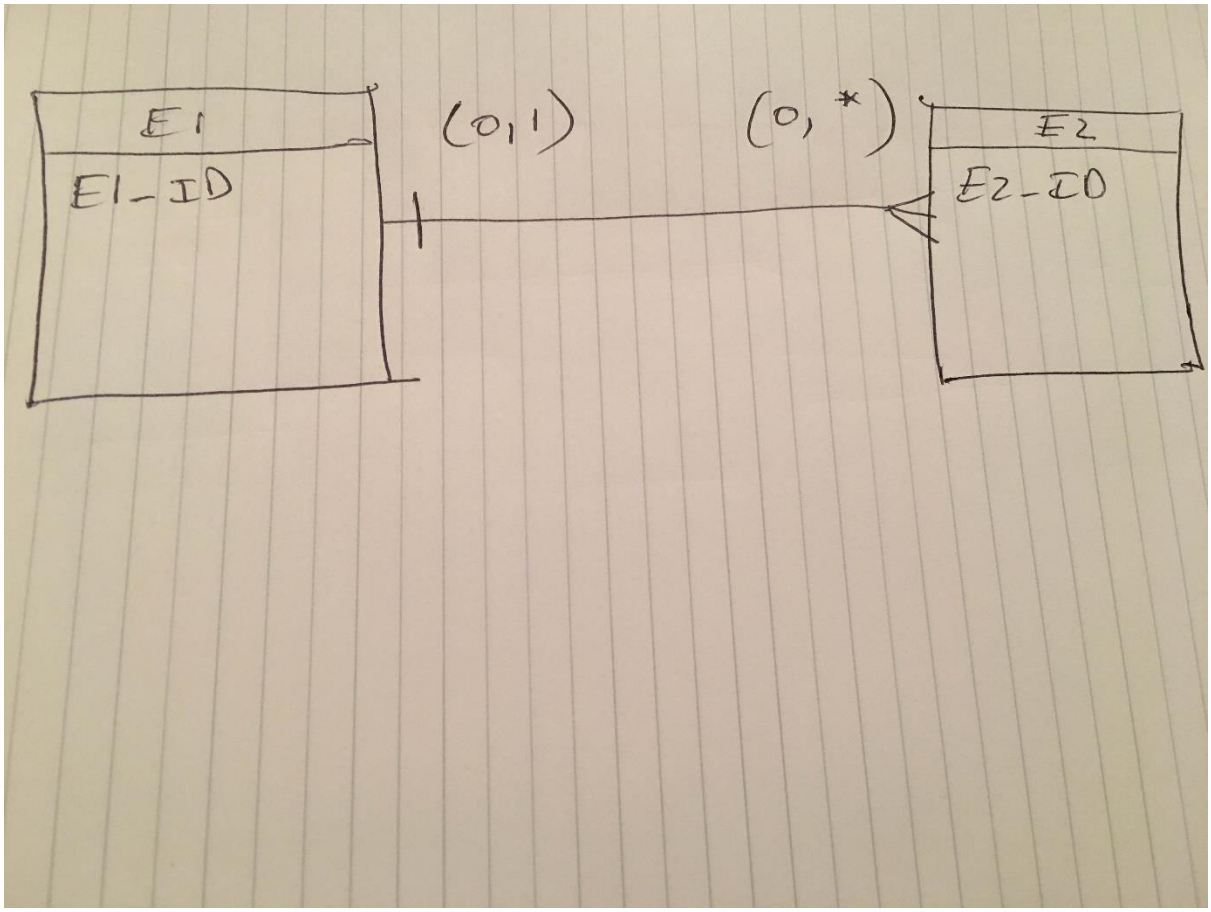
It's also flexible because all toys don't have to have all 3 attributes, some toys might have only 1 or 2 unique attributes or maybe none.

If new attributes were added, they would go in the Attribute1 etc. fields.

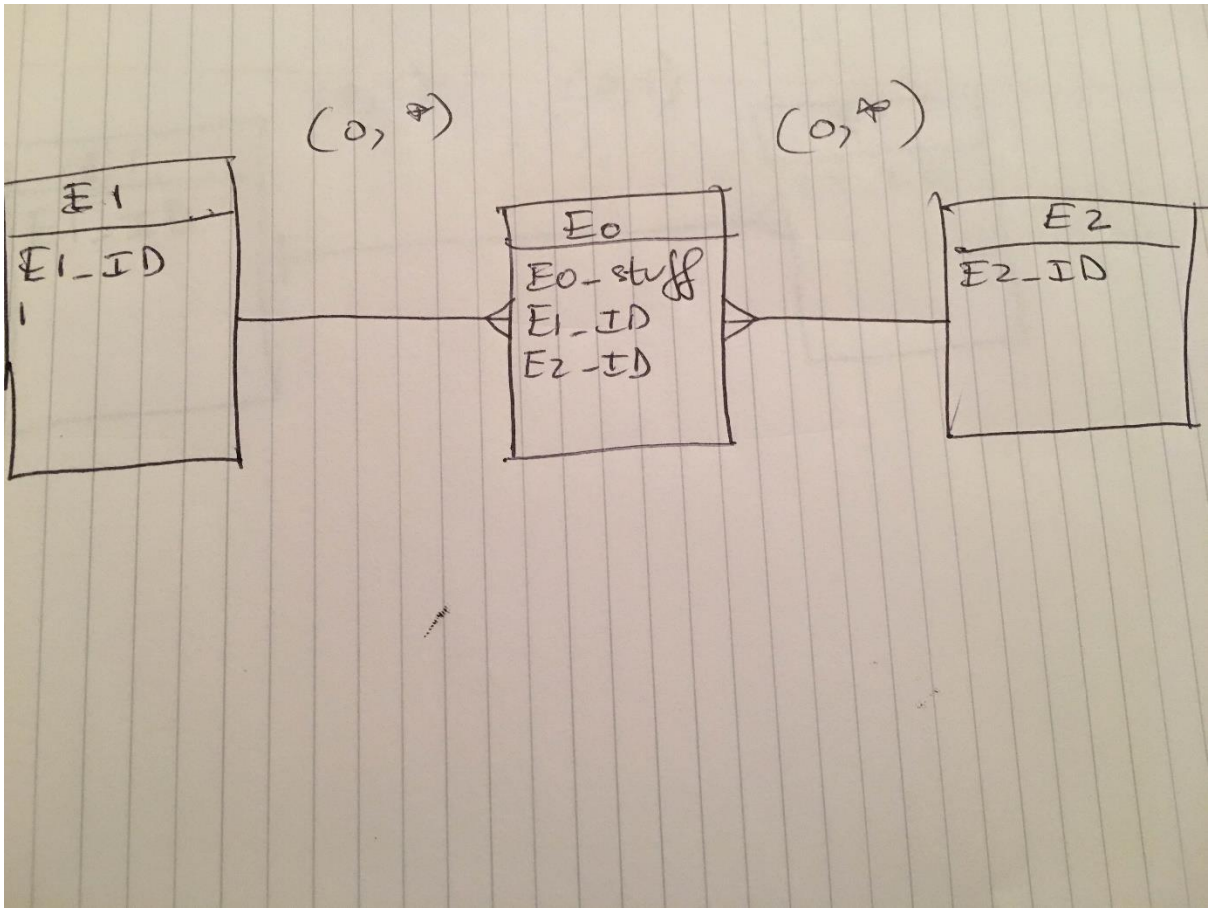
Toy

Toy_ID	
Name	
Price	
Type	
Attribute1	(O)
Attribute2	(O)
Attribute3	(O)

Exercise 2 – 1.

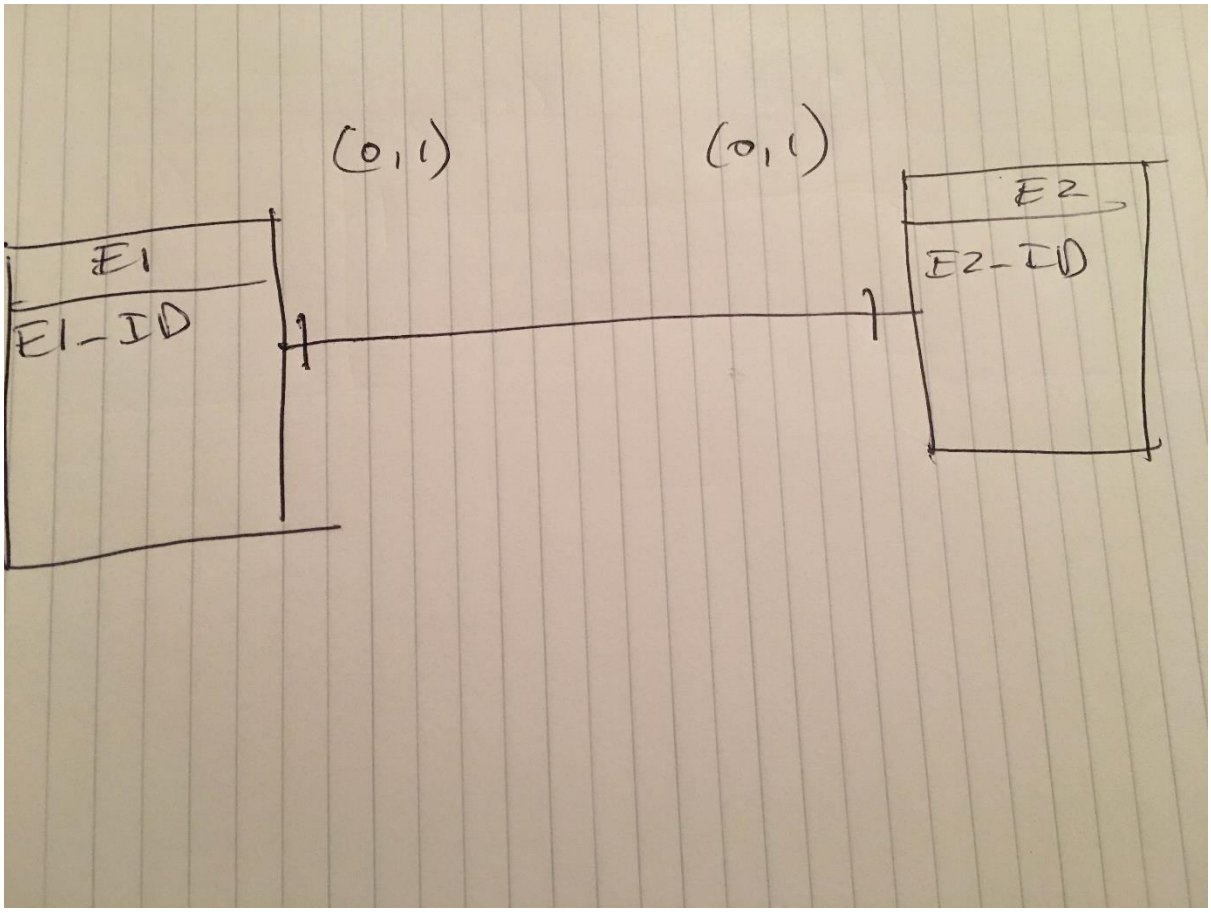


2.



Lab 1

3.



Last Q:

It is impossible to implement that because E1 states it needs to have 1 of E2, whereas E2 needs to have either many or 1 of E1. This does not allow E1 to be null at all and E1 always needs to have something in the table.

Exercise 3

A Cinema can have multiple screens. Screen can have multiple performances on one day. A movie can have multiple performances on in one day. A customer can have multiple bookings. A person can make multiple bookings and hence a performance can have multiple sales.