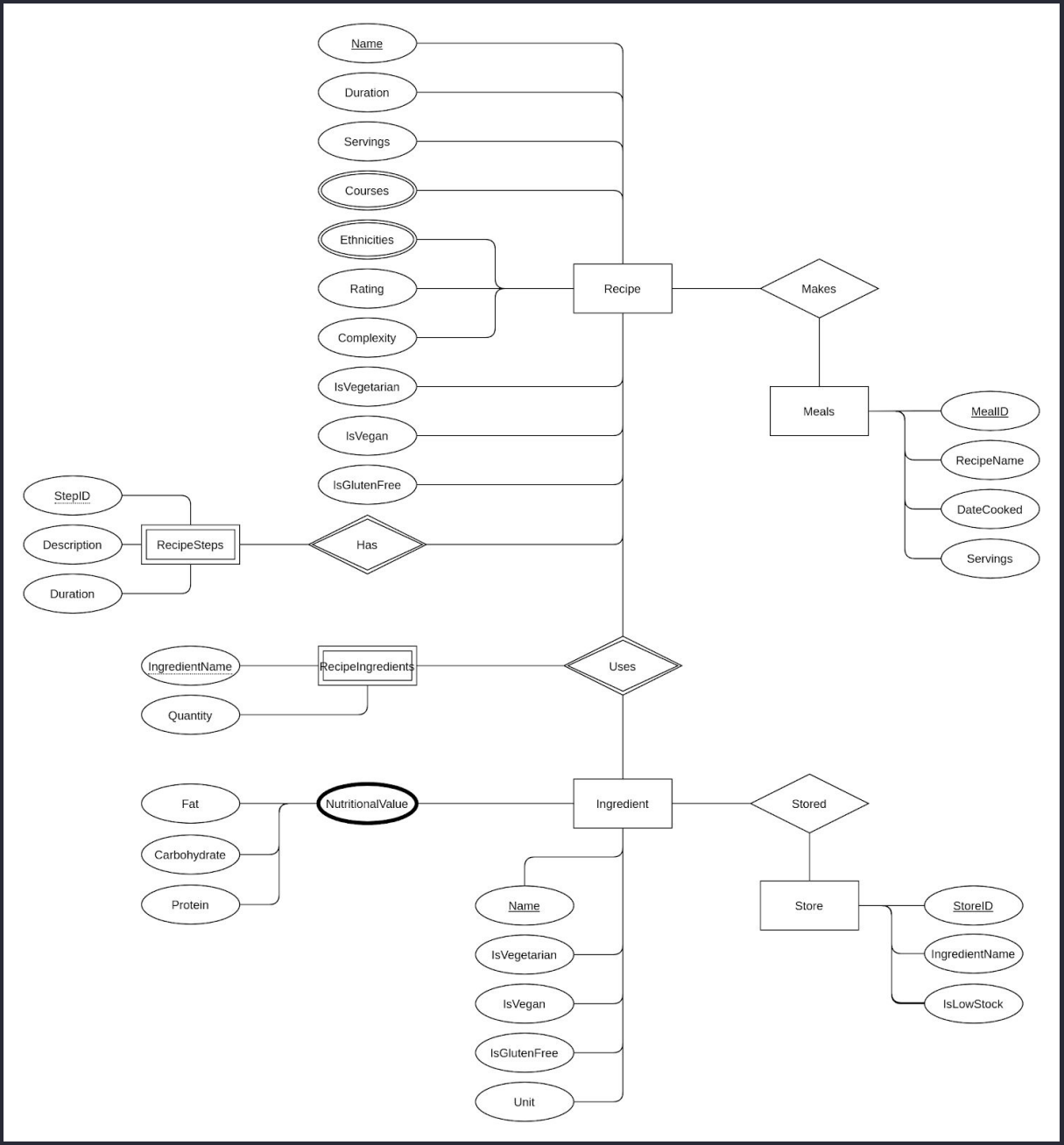
Databases : Assignment One

Q1.

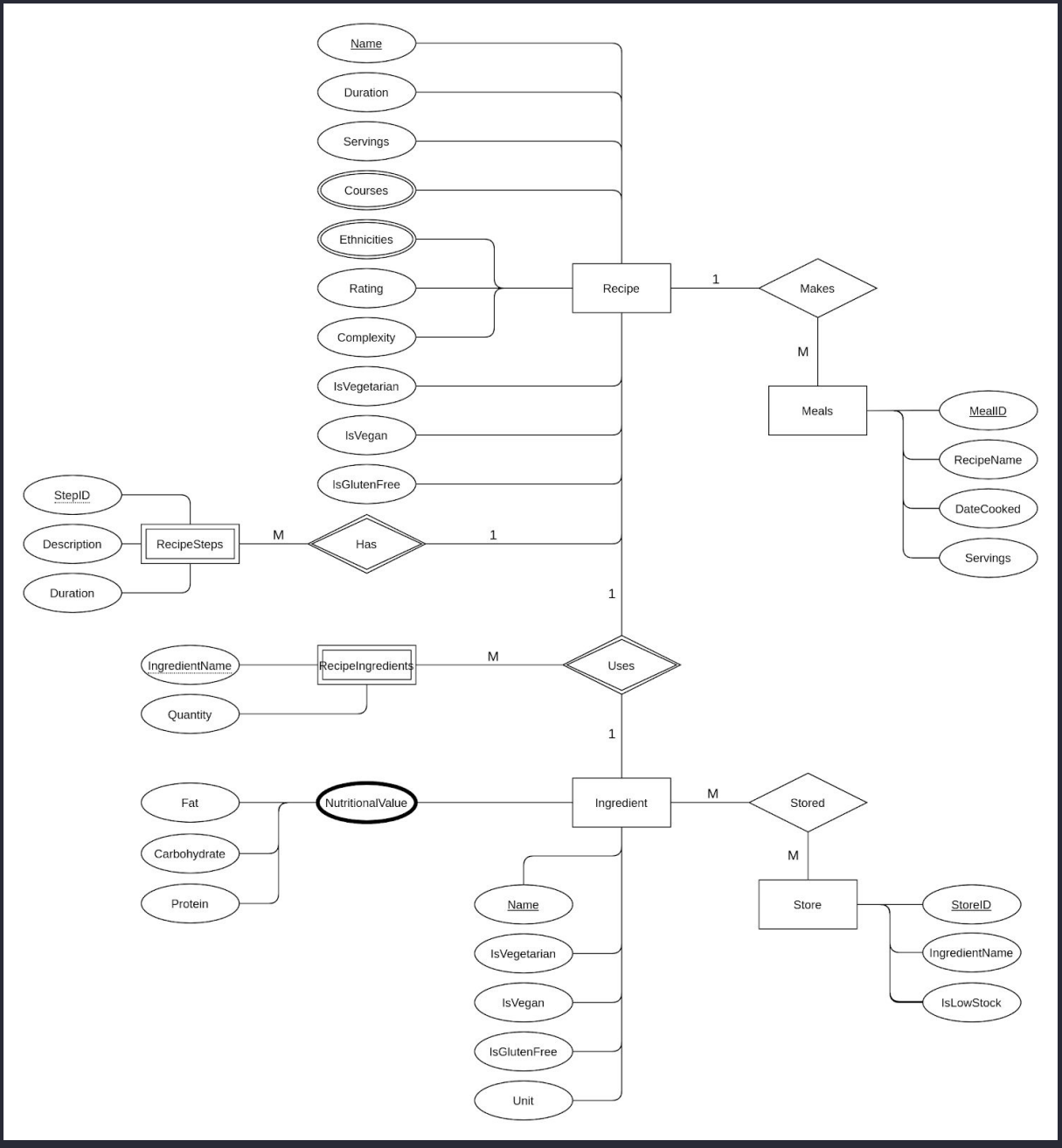
a)



Issues:

The meal entity does not have a unique identifier, the first thought was to make a combination of all attributes (recipe name, date cooked and servings) as a unique identifier however there is a possibility of someone cooking the same meal multiple times with the same servings in a day. The other solution to this was to create a new attribute which would be used to store a unique number.

b)



Relation “Recipe MAKES Meals” :

A recipe will MAKES multiple meals

A meal is MADE from one recipe

So cardinality is 1:M

Relation “Recipe USES RecipeIngredients” :

A recipe will USE multiple recipe ingredients

A recipe ingredient is USED in a recipe

So cardinality is 1:M

Relation “Recipe HAS RecipeSteps” :

A recipe HAS multiple recipe steps

A recipe step will HAVE one recipe

So cardinality is 1:M

Relation “RecipeIngredients USES Ingredients”

A recipe ingredients wille USE multiple ingredients

One ingredient will be USED in multiple recipes ingredients

So the cardinality is 1:M

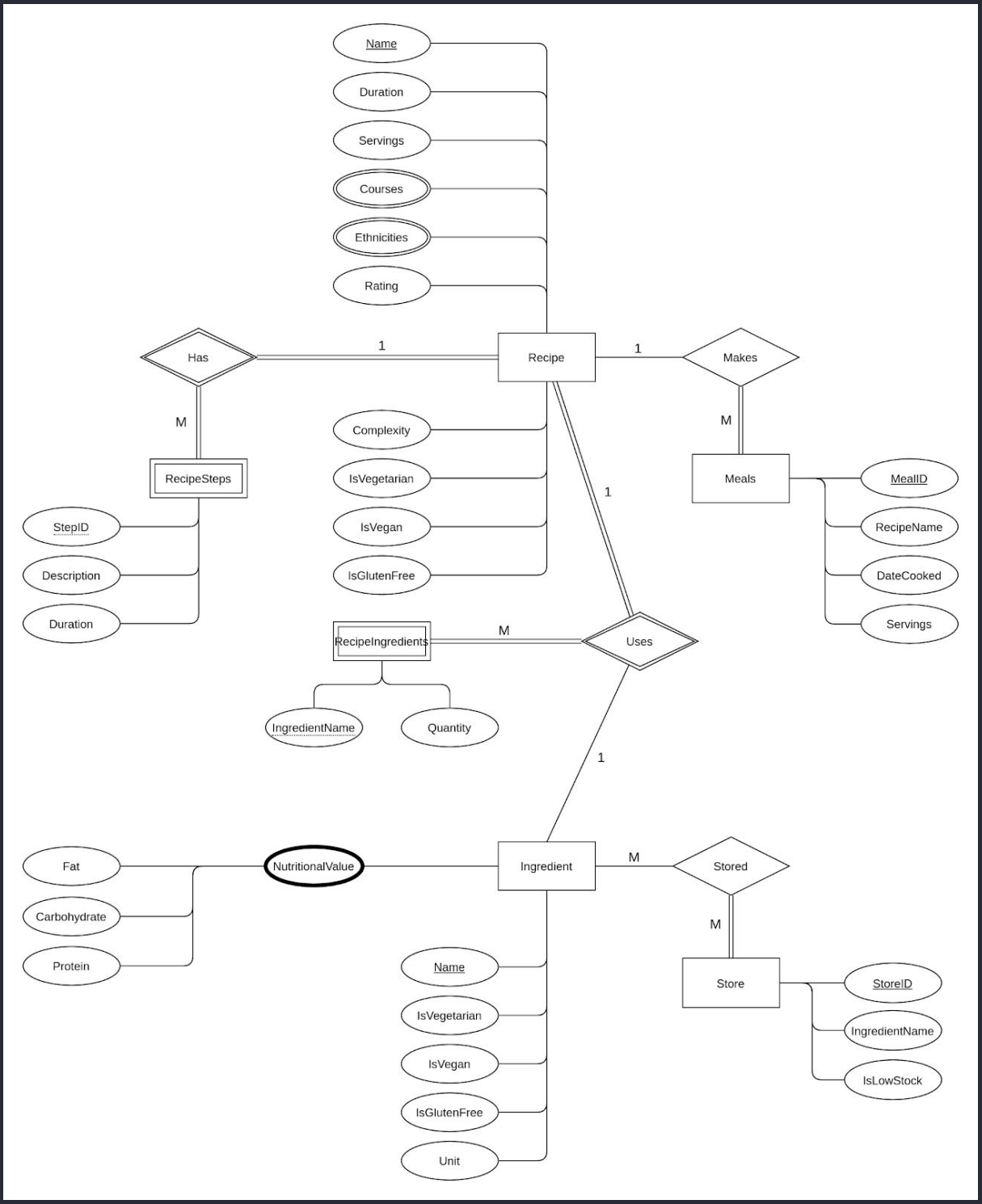
Relation “Ingredient STORED Store”

Many ingredients are STORED in a store

Many stores STORE ingredients

So the cardinality is M:M

c)


Name
Duration
Servings
Courses
Ethnicities
Rating
Has
1
Recipe
1
Makes
M
M
Complexity
IsVegetarian
IsVegan
IsGlutenFree
RecipeSteps
Meals
MealID
RecipeName
DateCooked
Servings
StepID
Description
Duration
1
RecipeIngredients
M
Uses
IngredientName
Quantity
1
Fat
Carbohydrate
Protein
NutritionalValue
Ingredient
M
Stored
M
Name
IsVegetarian
IsVegan
IsGlutenFree
Unit
Store
StoreID
IngredientName
IsLowStock

Relation “Recipe MAKES Meals” :

A meal has to be made from a recipe

Not every recipe is made into a meal

So there is partial participation for Recipe and total participation for Meal

Relation “Recipe USES RecipeIngredients” :

A recipe has to have recipe ingredients

Recipe ingredients has to come from a recipe

So there is total participation for both Recipe and RecipeIngredients

Relation “Recipe HAS RecipeSteps” :

A recipe has to have recipe steps

Recipe steps has to come from a recipe

So there is total participation for both Recipe and RecipeSteps

Relation “RecipeIngredients USES Ingredients”

Recipe ingredients have to come from ingredients

An ingredient doesn’t have to be in a recipe

So there is partial participation for Ingredients and total participation for RecipeIngredients

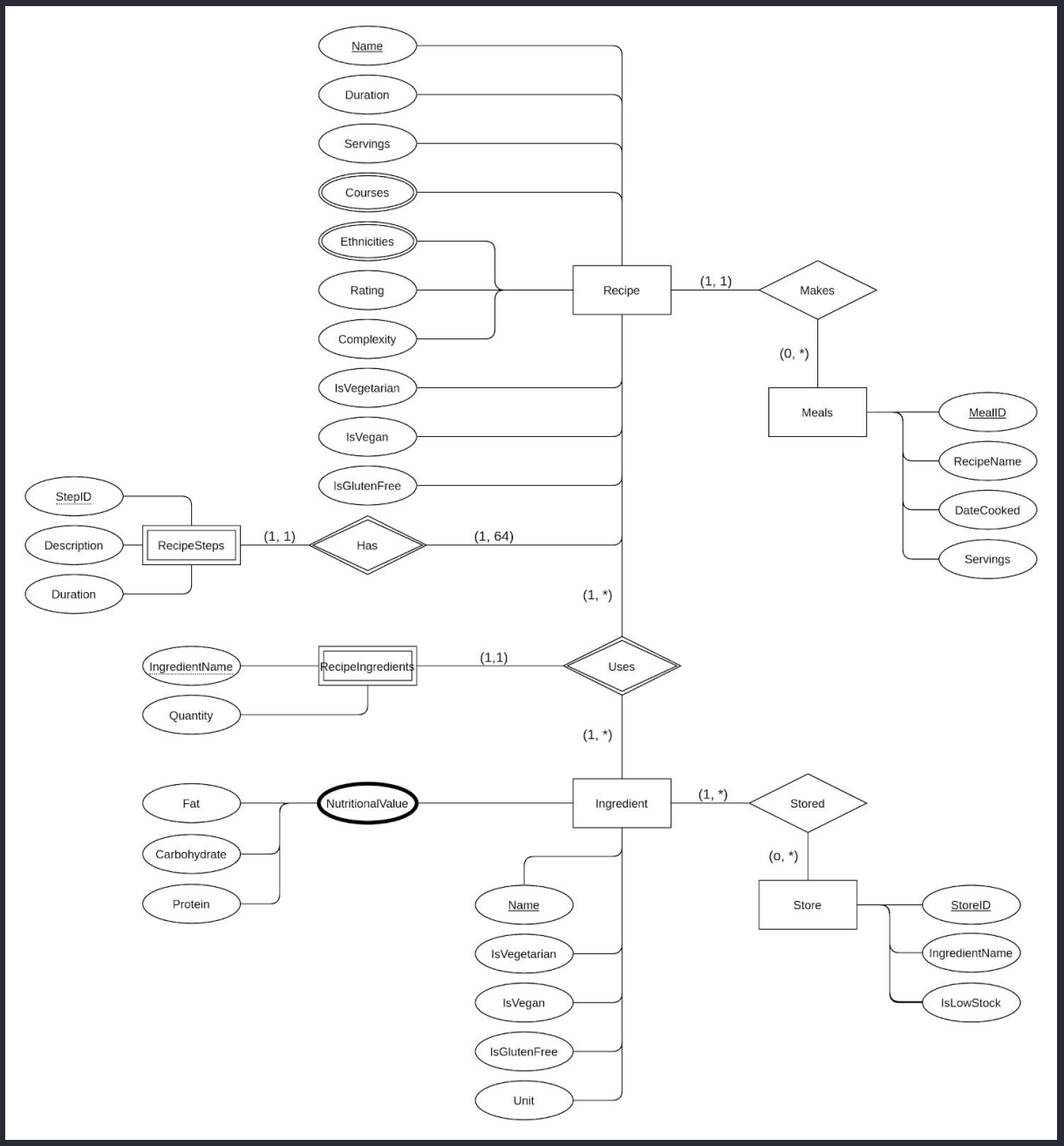
Relation “Ingredient STORED Store”

A store could store an ingredient

A ingredient could be stored in a store

So there is partial participation for both Ingredient and Store

d)


Name
Duration
Servings
Courses
Ethnicities
Rating
Complexity
IsVegetarian
IsVegan
IsGlutenFree
Recipe
(1, 1)
Makes
(0, *)
Meals
MealID
RecipeName
DateCooked
Servings
StepID
Description
Duration
RecipeSteps
(1, 1)
Has
(1, 64)
(1, *)
IngredientName
Quantity
RecipeIngredients
(1,1)
Uses
(1, *)
Fat
Carbohydrate
Protein
NutritionalValue
Ingredient
(1, *)
Stored
(o, *)
Store
StoreID
IngredientName
IsLowStock
Name
IsVegetarian
IsVegan
IsGlutenFree
Unit

Relation “Recipe MAKES Meals” :

A meal is made from 1 recipe only

A meal could never be made, or could be made many times with no bound for the maximum

Relation “Recipe USES RecipeIngredients” :

A recipe ingredient comes from 1 recipe only

A recipe has to have at minimum of 1 ingredient, with no bound for the maximum

Relation “Recipe HAS RecipeSteps” :

A recipe step comes from 1 recipe only

A recipe must have minimum of 1 step and maximum of 64

Relation “RecipeIngredients USES Ingredients”

A recipe ingredient can be 1 ingredient only

A ingredient is at minimum used 1 time in a recipe ingredients, with no bound for the maximum

Relation “Ingredient STORED Store”

A ingredient could never be stored, or could be stored in many locations with no bound for the maximum

A store could store ingredients minimum of 1 times, with no bound for the maximum

Q2

a)

Yes, this is due to the entity Order which is the super-class of both PhoneOrder and TableOrder which could be used instead as the participation of the sub-classes are optional.

b)

A primary key is an attribute or set of attributes which is used to identify a table if there is an attribute or set of attributes which create a unique value to be used as one.

c)

Yes, as PhoneOrder and TableOrder are both a sub-class of Order and the only types of order so each of the total attribute occurrences in Order could be added together as both sub-classes share the attribute.

Q3

a)

This is expressed through the cardinality constraints between Table and TableOrder with a one to many relationship translating to one table can have multiple table orders but a table order can only be for one table.

b)

This is not expressed as the details are too fine grained for an ER diagram as the constraints are constrained to only describing the maximum number of possible relationship occurrences not the details of two different table orders at roughly the same time come from different tables.

Q4

Restaurant (name)

Primary key name

PhoneNum (name, phone_num)

Primary key (name, phone_num)

Foreign key name references Restaurant (name)

Table (name, number)

Primary key (name, number)

Foreign key name references Restaurant (name)

Order (billingNo, dateTime, total, name)

Primary key (billingNo)

Foreign key name references Restaurant (name)

PhoneOrder (billingNo, street, postcode, city)

Primary key (billingNo)

Foreign key billingNo references Order (billingNo)

TableOrder (billingNo, waiter, number)

Primary key (billingNo)

Foreign key billingNo references Order (billingNo)

Foreign key number references Table (number)

number NOT NULL