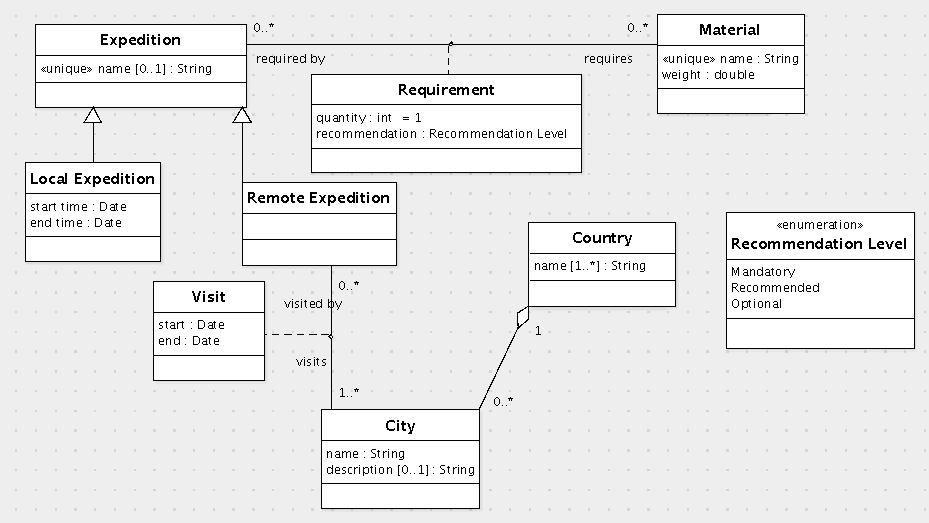
**CS 5200 Introduction to Database Management Homework #2**

Compile the following UML diagram to the relational model:



Write the translation in SQL.

create table Expedition (

id int primary key,

name varchar(500) unique

) engine=InnoDB;

create table Material (

id int primary key,

name varchar(500) not null unique,

weight double not null

) engine=InnoDB;

create table Requirement (

requiredBy int references Material

/\* If some expedition is using a material, then one should not delete it. \*/

on update cascade,

requires int references Expedition

/\* If an expedition is deleted, then it no longer requires any materials. \*/

on update cascade on delete cascade,

quantity int not null default 1,

recommendation enum ('Mandatory', 'Recommended', 'Optional') not null,

primary key(requiredBy, requires)

) engine=InnoDB;

create table Country (

id int primary key

) engine=InnoDB;

create table CountryName (

country int references Country(id)

/\* If a country is deleted, then its attributes should also be deleted. \*/

on update cascade on delete cascade,

name varchar(500),

primary key(country, name)

) engine=InnoDB;

create table City (

id int primary key,

name varchar(500) not null,

description varchar(5000),

country int not null references Country(id)

/\* If a country is deleted, then its cities are no longer needed in the database. \*/

on update cascade on delete cascade

) engine=InnoDB;

create table LocalExpedition (

id int primary key references Expedition(id)

/\* Subclass records are about the same object, so if an expedition is deleted,

then all of its records must be deleted. \*/

on update cascade on delete cascade,

startTime Date not null,

endTime Date not null

) engine=InnoDB;

create table RemoteExpedition (

id int primary key references Expedition(id)

/\* Subclass records are about the same object, so if an expedition is deleted,

then all of its records must be deleted. \*/

on update cascade on delete cascade

) engine=InnoDB;

create table Visit (

visits int references City(id)

/\* If any expedition is visiting a city, then the city should not be deleted. \*/

on update cascade,

visitedBy int references RemoteExpedition(id)

/\* If an expedition is deleted, then it is no longer visiting any city. \*/

on update cascade on delete cascade,

start Date not null,

end Date not null,

primary key(visits, visitedBy)

) engine=InnoDB;

alter table Country

add constraint fk\_CountryName foreign key(id) references CountryName(country);

alter table RemoteExpedition

add constraint fk\_RemoteExpedition foreign key(id) references Visit(visits);

## Grading criteria

(70 items)

1. Expedition table
2. Expedition primary key (cannot be name column)
3. Expedition.name column of type varchar
4. Expedition.name column is unique
5. Expedition.name column may be null
6. Material table
7. Material primary key (can be name column)
8. Material.name column of type varchar
9. Material.name column is unique
10. Material.name column may not be null
11. Material.weight column of type double
12. Material.weight column may not be null
13. Requirement table
14. Requirement.requires column of same type as Material primary key
15. Requirement.requires column references Material primary key
16. Requirement.requires column has commented enforcement step (See Note [2])
17. Requirement.requiredBy column of same type as Expedition primary key
18. Requirement.requiredBy column references Expedition primary key
19. Requirement.requiredBy column has commented enforcement step
20. Requirement primary key is (requires, requiredBy)
21. Requirement.quantity column of type int
22. Requirement.quantity column may not be null
23. Requirement.quantity column has default value 1
24. Requirement.recommendation column of enum type
25. The enum type has the literal values 'Mandatory', 'Recommended', 'Optional'
26. Requirement.recommendation column may not be null
27. LocalExpedition table (See Note [1])
28. LocalExpedition primary key of same type as Expedition primary key
29. LocalExpedition primary key references Expedition primary key
30. LocalExpedition primary key has commented enforcement step
31. LocalExpedition.startTime column of type Date
32. LocalExpedition.startTime column may not be null
33. LocalExpedition.endTime column of type Date
34. LocalExpedition.endTime column may not be null
35. Remote Expedition table (See Note [1])
36. RemoteExpedition primary key of same type as Expedition primary key
37. RemoteExpedition primary key references Expedition primary key
38. RemoteExpedition primary key has commented enforcement step
39. RemoteExpedition primary key references Visit.visitedBy column (preferably using alter table with no enforcement step, but this is not required)
40. Country table
41. Country primary key
42. Country primary key references CountryName.country column (preferably using alter table with no enforcement step, but this is not required)
43. CountryName table
44. CountryName.country column of same type as Country primary key
45. CountryName.country column references Country primary key
46. CountryName.country column has commented enforcement step
47. CountryName.name column of type varchar
48. CountryName primary key is (country, name)
49. City table
50. City primary key may not be name
51. City.name column of type varchar
52. City.name column may not be null
53. City.description column of type varchar
54. City.description column may be null
55. City.country column of same type as Country primary key
56. City.country column references Country primary key
57. City.country column has commented enforcement step "on update cascade on delete cascade"
58. City.country column may not be null
59. Visit table
60. Visit.visits column of same type as City primary key
61. Visit.visits column references City primary key
62. Visit.visits column has commented enforcement step
63. Visit.visitedBy column of same type as RemoteExpedition primary key
64. Visit.visitedBy column references RemoteExpedition primary key
65. Visit.visitedBy column has commented enforcement step
66. Visit primary key (visits, visitedBy)
67. Visit.start column of type Date
68. Visit.start column may not be null
69. Visit.end column of type Date
70. Visit.end column may not be null

Notes:

 The commented enforcement steps are required, but they need not be the same as what I wrote as long as their comments are more or less consistent with their enforcement steps. The only exception is the enforcement step for City.country. This must be "on update cascade on delete cascade".

 InnoDB is only for MySQL. This is not required.