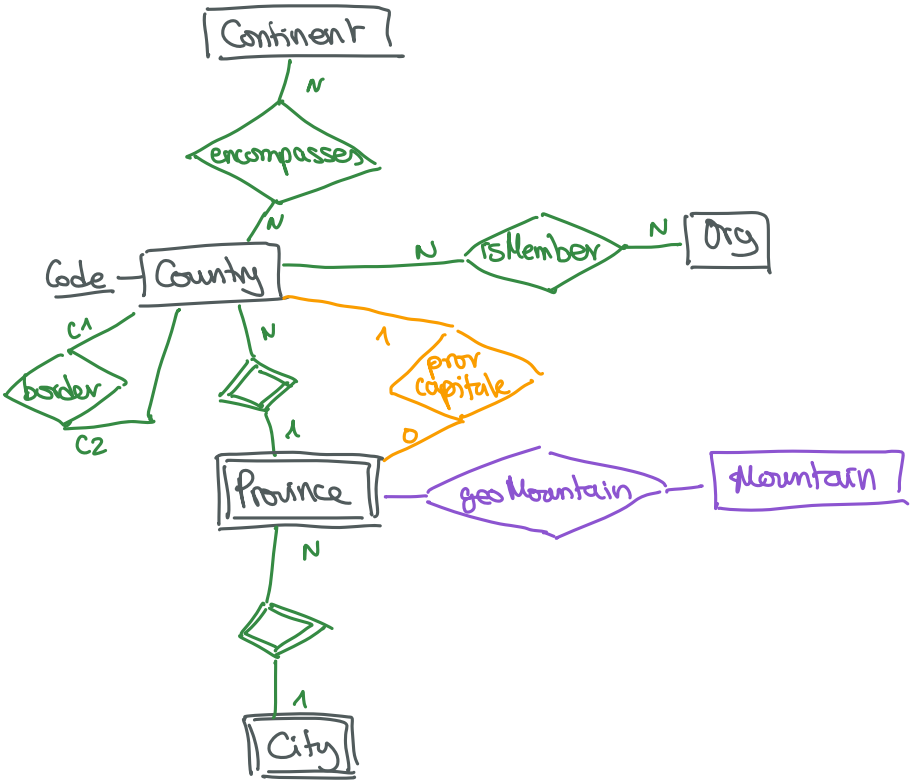
Diagramme EA :

1:N → ajouter clé étrangère
N:N → autre relation



(1) (2)
```
SELECT   c.name , c.population
FROM     Country c , Organization o , isMember m
WHERE    c.code = m.country
   and   o.abbreviation = m.organization
   and   o.name = 'United Nations'
ORDER BY c.name c.population DESC
```

simplification.

(3)
```
SELECT c.name
FROM   Country c
WHERE  c.code NOT IN | SELECT   m.country
                     | FROM     Organization o , isMember m
                     | WHERE    o.abbreviation = m.organization
                     |    and   o.name = 'United Nations'
```

(3b)

Avec une soustraction A - B
les éléments de A et B sont des identifiants de pays 'code'

A
```
SELECT c.code , c.name
FROM Country c
```

MINUS

B
```
SELECT   m.country, c.name
FROM     Country c , Organization o , isMember m
WHERE    c.code = m.country
   and   o.abbreviation = m.organization
   and   o.name = 'United Nations'
```

④

```
SELECT  b.country2
FROM    Country f , Borders b
WHERE   f.code = b.country1
  and   f.name = 'France'
UNION
SELECT  b.country1
FROM    Country f , Borders b
WHERE   f.code = b.country2
  and   f.name = 'France'
```

Si on veut le c.name, on ajoute un 2ème Country et b.country2 = c.code

UNION dans 2 requêtes

⑤

```
SELECT  v.code, v.name
FROM    Country f, Border b , Country v
WHERE   f.name = 'France'
and   ((f.code = b.country1  and  v.code = b.country2)
          or
       (f.code = b.country2  and  v.code = b.country1))
```

or c'est dans WHERE entre 2 booleens

⑥

```
SELECT  SUM (b.length)
FROM    Country f, Border b
WHERE   f.name = 'France'
  and   ( f.code = b.country1
          or
          f.code = b.country2 )
```

SELECT c.code, COUNT b.

⑦

```
SELECT   p.code, count(*), p.name
FROM     Border b, Country p
WHERE    p.code = b.country1
   or    p.code = b.country2

GROUP BY p.code, p.name   ← à ajouter en THE
```

GROUP BY c.code

⑧

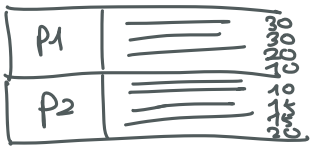
```
SELECT   p.code, p.name, SUM (v.population)
FROM     Border b, Country p , Country v
WHERE  ( p.code = b.country1  and  v.code = b.country2 )
   or  ( p.code = b.country2  and  v.code = b.country1)
GROUP BY p.code, p.name
```

⑨

```
SELECT   p.code, p.name, SUM (v.population)
FROM     Border b, Country p , Country v , Encompasses e
WHERE  ( ( p.code = b.country1  and  v.code = b.country2 )
         or
         ( p.code = b.country2  and  v.code = b.country1) )
GROUP BY p.code, p.name
HAVING   p.code in ( SELECT  e.country
                     From    Encompasses e
                     WHERE   e.continent = 'Europe' )
```

(10) (11)

```
SELECT  o.abbrev , sum(p.population) , count(*)
FROM    Organization o , Country p , isMember m
WHERE   m.country = p.code
   and  m.organization = o.abbrev
GROUP BY  o.abbrev
HAVING  COUNT(*) > 100
```



return p1, 80, 3
p2, 60, 4

RQ : on peut remplacer count(*) par count(1)
count(m.country) count(m.country, m.organization).
count(m.organization) mais pas ~~count(distinct m.organization)~~
count(Distinct country)

DISTINCT enlève les doublons

(12) les pays avec leur montagne.

tq la hauteur = [hauteur max des montagnes de ce pays]

ou

∄ autre montagne de ce pays dont la hauteur est > à cette montagne.