32.-

SELECT P.cod\_pais, P.nombre

FROM pais P

WHERE NOT EXISTS (SELECT \*

FROM actor A

WHERE P.cod\_pais = A.cod\_pais

AND NOT (EXTRACT(YEAR FROM A.fecha\_nac) BETWEEN 1900 AND 1999)

)

AND EXISTS (SELECT \*

FROM actor A

WHERE P.cod\_pais = A.cod\_pais

)

ORDER BY P.nombre

33.-

SELECT A.cod\_act, A.nombre

FROM actor A

WHERE NOT EXISTS (SELECT \*

FROM actua AC

WHERE AC.cod\_act = A.cod\_act

AND NOT AC.papel = 'Secundario'

)

AND EXISTS (SELECT \*

FROM actua AC

WHERE AC.cod\_act = A.cod\_act

)

ORDER BY A.nombre

34.-

SELECT DISTINCT A.cod\_act, A.nombre

FROM actor A, actua AC

WHERE A.cod\_act = AC.cod\_act

AND NOT EXISTS (SELECT \*

FROM pelicula P

WHERE AC.cod\_peli = P.cod\_peli

AND NOT P.director = 'Guy Ritchie'

)

AND EXISTS (SELECT \*

FROM pelicula P

WHERE P.director = 'Guy Ritchie'

)

ORDER BY A.cod\_act

35.-

SELECT DISTINCT A.cod\_act, A.nombre

FROM actor A, actua AC

WHERE A.cod\_act = AC.cod\_act

AND NOT EXISTS (SELECT \*

FROM pelicula P

WHERE AC.cod\_peli = P.cod\_peli

AND NOT P.director = 'John Steel'

)

AND EXISTS (SELECT \*

FROM pelicula P

WHERE P.director = 'John Steel'

)

ORDER BY A.cod\_act;

36.-

SELECT P.cod\_peli, P.titulo

FROM pelicula P

WHERE P.duracion < 100

AND 1 = (SELECT COUNT(DISTINCT A.cod\_pais)

FROM actua AC, actor A

WHERE AC.cod\_peli = P.cod\_peli

AND A.cod\_act = AC.cod\_act

)

37.-

SELECT P.cod\_peli, P.titulo, P.anyo

FROM pelicula P

WHERE NOT EXISTS (SELECT A.cod\_act

FROM actua AC, actor A

WHERE AC.cod\_peli = P.cod\_peli

AND A.cod\_act = AC.cod\_act

AND EXTRACT(YEAR FROM A.fecha\_nac) >= 1943

)

AND EXISTS (SELECT \*

FROM actua AC

WHERE AC.cod\_peli = P.cod\_peli

)

ORDER BY P.titulo

38.-

SELECT P.cod\_pais, P.nombre

FROM pais P

WHERE NOT EXISTS(SELECT \*

FROM actor A

WHERE A.cod\_pais = P.cod\_pais

AND NOT EXISTS(SELECT \*

FROM pelicula PE, actua AC

WHERE AC.cod\_peli = PE.cod\_peli

AND AC.cod\_act = A.cod\_act

AND PE.duracion > 120)

)

AND EXISTS(SELECT \*

FROM actor A, actua AC

WHERE A.cod\_pais = P.cod\_pais

)

ORDER BY P.nombre

39.-

SELECT L.cod\_lib, L.titulo, COUNT(\*)

FROM libro\_peli L, pelicula P

WHERE L.cod\_lib = P.cod\_lib

GROUP BY L.cod\_lib, L.titulo

HAVING COUNT(\*) > 1

40.-

SELECT G.cod\_gen, G.nombre, COUNT(\*), ROUND(AVG(P.duracion))

FROM genero G, pelicula P, clasificacion C

WHERE C.cod\_gen = G.cod\_gen

AND P.cod\_peli = C.cod\_peli

GROUP BY G.cod\_gen, G.nombre

HAVING COUNT(\*) > 5

41.-

SELECT P.cod\_peli, P.titulo, COUNT(\*)

FROM genero G, pelicula P, clasificacion C

WHERE C.cod\_gen = G.cod\_gen

AND P.cod\_peli = C.cod\_peli

AND P.anyo > 2000

GROUP BY P.cod\_peli, P.titulo

HAVING COUNT(\*) > 0

ORDER BY P.titulo

42.-

SELECT P.director

FROM pelicula P

WHERE P.director LIKE 'George%'

GROUP BY P.director

HAVING COUNT(\*) = 2

43.-

SELECT P.cod\_peli, P.titulo, COUNT(\*)

FROM pelicula P, actua AC

WHERE P.cod\_peli = AC.cod\_peli

AND NOT AC.cod\_act IS NULL

AND 1 = (SELECT COUNT(\*)

FROM clasificacion C

WHERE C.cod\_peli = P.cod\_peli

)

GROUP BY P.cod\_peli, P.titulo

ORDER BY P.titulo

44.-

SELECT P.cod\_pais, P.nombre, COUNT(DISTINCT A.nombre)

FROM pais P, actor A, actua AC, pelicula P

WHERE P.cod\_pais = A.cod\_pais

AND AC.cod\_act = A.cod\_act

AND P.cod\_peli = AC.cod\_peli

AND P.anyo LIKE '196%'

GROUP BY P.cod\_pais, P.nombre

ORDER BY P.nombre

45.-

SELECT G.cod\_gen, G.nombre

FROM genero G, clasificacion C

WHERE G.cod\_gen = C.cod\_gen

GROUP BY G.cod\_gen, G.nombre

HAVING COUNT(C.cod\_peli) = (SELECT MAX(COUNT(C1.cod\_peli))

FROM clasificacion C1

GROUP BY C1.cod\_gen

)

46.-

SELECT L.cod\_lib, L.titulo, L.autor

FROM libro\_peli L, pelicula P

WHERE L.cod\_lib = P.cod\_lib

GROUP BY L.cod\_lib, L.titulo, L.autor

HAVING COUNT(P.cod\_peli) = (SELECT MAX(COUNT(P1.cod\_peli))

FROM pelicula P1

GROUP BY P1.cod\_lib

)

47.-

SELECT P.cod\_pais, P.nombre

FROM pais P, actor A

WHERE P.cod\_pais = A.cod\_pais

AND A.cod\_act IN (SELECT AC.cod\_act

FROM actua AC

GROUP BY AC.cod\_act

HAVING COUNT(AC.cod\_peli)=2

)

GROUP BY P.cod\_pais, P.nombre

HAVING COUNT(A.cod\_act) = (SELECT MAX(COUNT(A2.cod\_act))

FROM actor A2, pais P2

WHERE P2.cod\_pais = A2.cod\_pais

AND A2.cod\_act IN (SELECT AC2.cod\_act

FROM actua AC2

GROUP BY AC2.cod\_act

HAVING COUNT(AC2.cod\_peli)=2

)

GROUP BY P2.cod\_pais

)

48.-

SELECT EXTRACT(YEAR FROM A.fecha\_nac), COUNT(\*)

FROM actor A

GROUP BY EXTRACT(YEAR FROM A.fecha\_nac)

HAVING COUNT(\*) > 3

49.-

SELECT P.cod\_peli, P.titulo

FROM pelicula P, actua AC, actor A

WHERE P.duracion < 100

AND AC.cod\_act = A.cod\_act

AND P.cod\_peli = AC.cod\_peli

GROUP BY P.cod\_peli, P.titulo

HAVING COUNT(DISTINCT A.cod\_pais) = 1

50.-

SELECT P.cod\_pais, P.nombre, COUNT(A.cod\_act)

FROM pais P LEFT JOIN actor A ON P.cod\_pais = A.cod\_pais

GROUP BY P.cod\_pais, P.nombre

ORDER BY P.nombre

51.-

SELECT L.cod\_lib, L.titulo, COUNT(P.cod\_lib)

FROM libro\_peli L LEFT JOIN pelicula P ON L.cod\_lib = P.cod\_lib

WHERE L.anyo > 1980

GROUP BY L.cod\_lib, L.titulo

ORDER BY L.titulo

52.-

SELECT P.cod\_pais, P.nombre, COUNT(T.cod\_pais)

FROM pais P LEFT JOIN (SELECT DISTINCT A.cod\_pais, A.cod\_act

FROM actor A, actua AC

WHERE A.cod\_act = AC.cod\_act

AND AC.papel = 'Secundario'

) T ON P.cod\_pais = T.cod\_pais

GROUP BY P.cod\_pais, P.nombre

ORDER BY P.nombre

53.-

SELECT P.cod\_peli, P.titulo, COUNT(DISTINCT C.cod\_gen), COUNT(DISTINCT AC.cod\_act)

FROM pelicula P LEFT JOIN clasificacion C ON C.cod\_peli = P.cod\_peli LEFT JOIN actua AC ON AC.cod\_peli = P.cod\_peli

WHERE P.duracion > 140

GROUP BY P.cod\_peli, P.titulo

ORDER BY P.titulo

54.-

SELECT anyo

FROM (SELECT anyo FROM libro\_peli UNION SELECT anyo FROM pelicula)

WHERE NOT anyo LIKE '%9%'

ORDER BY anyo

55.-

SELECT G.cod\_gen, G.nombre

FROM genero G, pelicula P, clasificacion C

WHERE P.cod\_peli = C.cod\_peli

AND C.cod\_gen = G.cod\_gen

AND P.duracion = (SELECT MAX(P2.duracion)

FROM pelicula P2

)

ORDER BY G.cod\_gen

56.-

SELECT A.cod\_act, A.nombre, A.fecha\_nac, COUNT(T.papel)

FROM actor A LEFT JOIN (SELECT AC.papel, AC.cod\_act, AC.cod\_peli

FROM actua AC

) T ON T.cod\_act = A.cod\_act AND T.cod\_act = 'Principal'

WHERE EXTRACT(YEAR FROM A.fecha\_nac) < 1948

GROUP BY A.cod\_act, A.nombre, A.fecha\_nac

HAVING COUNT(T.cod\_peli) > 1

ORDER BY A.nombre

MUSIC SCHEMA

17.-

SELECT DISTINCT C.nombre

FROM companyia C

WHERE NOT EXISTS (SELECT \*

FROM grupo G, disco D

WHERE G.cod = D.cod\_gru

AND G.pais = 'España'

AND D.cod\_comp = C.cod

)

18.-

SELECT DISTINCT C.nombre

FROM companyia C, disco D

WHERE D.cod\_comp = C.cod

AND NOT EXISTS (SELECT \*

FROM grupo G

WHERE G.cod = D.cod\_gru

AND G.pais <> 'España'

)

19.-

SELECT DISTINCT C.nombre, C.dir

FROM companyia C, disco D, grupo G

WHERE G.cod = D.cod\_gru

AND D.cod\_comp = C.cod

AND NOT EXISTS (SELECT \*

FROM grupo G2, disco D2

WHERE G2.cod = G.cod

AND G2.cod = D2.cod\_gru

AND D2.cod\_comp <> C.cod

)

AND EXISTS (SELECT \*

FROM grupo G3, disco D3

WHERE G3.cod = G.cod

AND G3.cod = D3.cod\_gru

)

20.-

SELECT G.nombre, SUM(C.num)

FROM grupo G, club C

WHERE G.cod = C.cod\_gru

AND G.pais = 'España'

GROUP BY G.nombre

21.-

CYCLIST RACE SCHEMA

29.-

SELECT DISTINCT J.code, J.color

FROM jersey J, wear W, cyclist C

WHERE J.code = W.code

AND C.cnum = W.cnum

AND NOT EXISTS (SELECT \*

FROM cyclist C1, wear W1

WHERE C1.cnum = W1.cnum

AND W.code = W1.code

AND C1.teamname <> C.teamname

)

30.-

SELECT DISTINCT T.teamname

FROM team T, climb CL

WHERE NOT EXISTS (SELECT \*

FROM cyclist C2, climb CL

WHERE C2.teamname = T.teamname

AND CL.cnum = C2.cnum

AND CL.category <> '1'

)

AND EXISTS (SELECT \*

FROM cyclist C2, climb CL

WHERE C2.teamname = T.teamname

AND CL.cnum = C2.cnum

)

36.-

SELECT C.teamname, AVG(C.age)

FROM cyclist C

GROUP BY C.teamname

HAVING AVG(C.age) = (SELECT MAX(AVG(C2.age))

FROM cyclist C2

GROUP BY C2.teamname

)

40.-

SELECT DISTINCT C.cnum, C.name

FROM cyclist C

WHERE EXISTS (SELECT \*

FROM wear W

WHERE W.cnum =20

AND NOT EXISTS (SELECT \*

FROM wear W2

WHERE W2.cnum = C.cnum

AND W2.code = W.code

)

)

ORDER BY C.cnum;