Ex5

First question:

1. In order to find all keys of athlete_events, we will run the algorithm we saw at class:
   ID, Year, Season, Event.
2. Finding normal form of athlete_events:
   Let us look at the first relation of F. ID → Name, Sex. ID is not a superkey, and Name is not an attribute in a key.
   It’s neither BCNF or 3NF.
3. Check if the decompression is lossless

$$R_1 = (\mathrm{ID}, \mathrm{Year}, \mathrm{Season}, \mathrm{Name}, \mathrm{Sex}, \mathrm{Age}, \mathrm{Height}, \mathrm{Weight})$$
$$R_2 = (\mathrm{ID}, \mathrm{Year}, \mathrm{Season}, \mathrm{City}, \mathrm{Team}, \mathrm{Event}, \mathrm{Sport}, \mathrm{Medal})$$
$$R_3 = (\mathrm{Team}, \mathrm{NOC})\ .$$

First we will build the row table:

	id	name	sex	age	height	weight	team	noc	year	season	city	sport	event	medal
R1	a1	a2	a3	a4	a5	a6	b17	b18	a9	a10	b111	b112	b113	b114
R2	a1	b22	b23	b24	b25	b26	a7	b28	a9	a10	a11	a12	a13	a14
R3	b31	b32	b33	b34	b35	b36	a7	a8	b39	b310	b311	b312	b313	b314

Now we will fix the contradictions:

ID → Name, Sex:

	id	name	sex	age	height	weight	team	noc	year	season	city	sport	event	medal
R1	a1	a2	a3	a4	a5	a6	b17	b18	a9	a10	b111	b112	b113	b114
R2	a1	a2	b23	b24	b25	b26	a7	b28	a9	a10	a11	a12	a13	a14
R3	b31	b32	b33	b34	b35	b36	a7	a8	b39	b310	b311	b312	b313	b314

Year, season → city:

	id	name	sex	age	height	weight	team	noc	year	season	city	sport	event	medal
R1	a1	a2	a3	a4	a5	a6	b17	b18	a9	a10	a11	b112	b113	b114
R2	a1	a2	b23	b24	b25	b26	a7	b28	a9	a10	a11	a12	a13	a14
R3	b31	b32	b33	b34	b35	b36	a7	a8	b39	b310	b311	b312	b313	b314

ID, Year, season, city → name, sex, age, height, weight, team, noc:

	id	**name**	**sex**	**age**	**height**	**weight**	**team**	**noc**	**year**	**season**	**city**	**sport**	**event**	**medal**
R1	a1	a2	a3	a4	a5	a6	a7	b18	a9	a10	a11	b112	b113	b114
R2	a1	a2	a3	a4	a5	a6	a7	b28	a9	a10	a11	a12	a13	a14
R3	b31	b32	b33	b34	b35	b36	a7	a8	b39	b310	b311	b312	b313	b314

team → noc:

	id	**name**	**sex**	**age**	**height**	**weight**	**team**	**noc**	**year**	**season**	**city**	**sport**	**event**	**medal**
R1	a1	a2	a3	a4	a5	a6	a7	a8	a9	a10	a11	b112	b113	b114
R2	**a1**	**a2**	**a3**	**a4**	**a5**	**a6**	**a7**	**a8**	**a9**	**a10**	**a11**	**a12**	**a13**	**a14**
R3	b31	b32	b33	b34	b35	b36	a7	a8	b39	b310	b311	b312	b313	b314

As we can see, row 2 has all attributes with an ‘a’, which means that this decomposition is lossless.

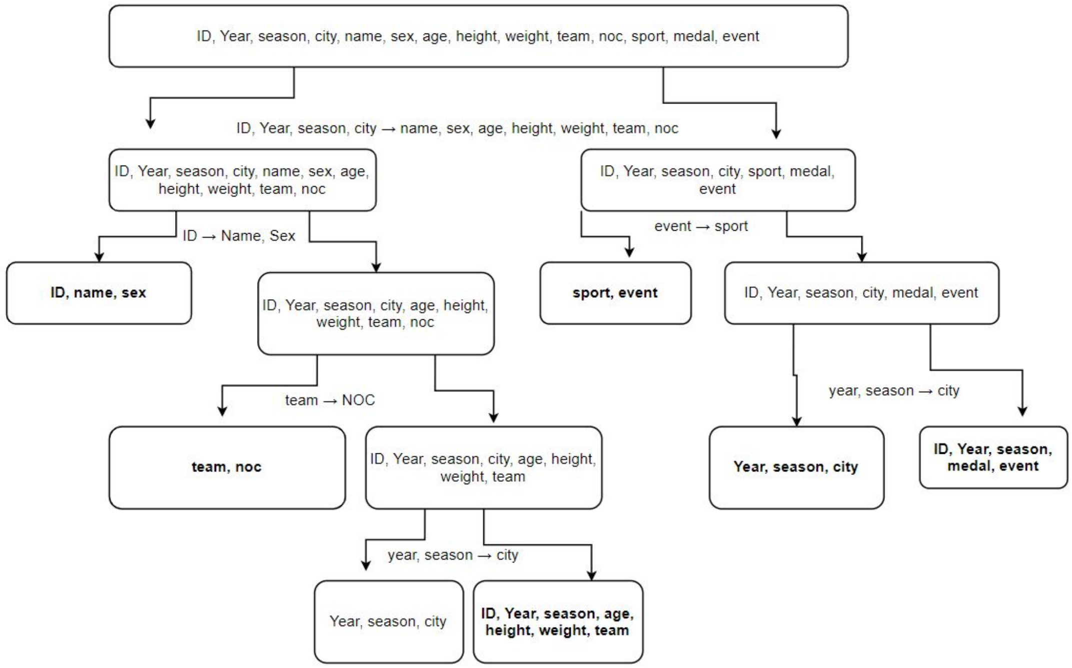
4. Step1:
   ID → Name; ID → Sex;
   Year, season → city;
   ID, Year, season, city → name; ID, Year, season, city → sex; ID, Year, season, city → age; ID, Year, season, city → height; ID, Year, season, city → weight; ID, Year, season, city → team; ID, Year, season, city → noc;
   event → sport
   team→ noc
   Noc, year→ team
   ID, Year, season, team, noc, event → sport
   ID, Year, season, team, noc, event → medal
   Step 2:
   Year+ = ø, season+ = ø
   So: **ID → Name; ID → Sex; Year, season → city;** are part of the minimal cover in this step
   (ID, year, season)+ = city, therefore:
   ID, Year, season → age; ID, Year, season → height; ID, Year, season → weight; ID, Year, season → team; ID, Year, season → noc; also part in this step.
   event → sport
   team→ noc
   noc+ = ø

Noc, year→ team
Event+ = sport, therefore: event → sport.
Team+ = noc, therefore: **ID, Year, season, event → medal**
<u>Step 3:</u>
Erase the redundancy, and the dependencies that can be conclude from others:
ID → Name;
ID → Sex;
Year, season → city;
ID, Year, season → age;
ID, Year, season → height;
ID, Year, season → weight;
ID, Year, season → team;
event → sport
team→ noc
Noc, year→ team
ID, Year, season, event → medal

5.

תת סכמה באנגלית	צורה נורמלית
ID, name	BCNF
ID,sex	BCNF
Year, season,city	BCNF
ID, Year, season,age	BCNF
ID, Year, season,height	BCNF
ID, Year, season,weight	BCNF
ID, Year, season,team	BCNF
Noc, year, team	3NF
event,sport	BCNF
Team, noc	BCNF
ID, year, season, event,medal	**BCNF**

6. According to the algorithm we have seen in class, The decomposition is:
R1 = (id, name, sex), R2 = (sport, event), R3 = (team, noc), R4 = (year, season, city), R5 = (ID, Year, season, medal, event), R6 = (ID, Year, season, age, height, weight, team).



7. No. let us take a look at the dependency **Noc, year→ team.**
   Define Z = Noc, year.
   R1; Z = Noc, year ∪ ((Noc, year ∩ id, name, sex)+ ∩ id, name, sex) = noc, year.
   We can see that in any scheme in the decomposition, (noc)+ = and (year)+ both empty, and those attributes don't appear together. beside the original dependency, there is no other dependency that determines 'team'.
   So we can infer that this decomposition is not dependency preserving.

Second question:

1. name → author, name → genre, name,year → price, name,year → reviews, name,year → user rating
2. (Name, year) is a key for this scheme because together it determines all parameters as we can conclude from the assumptions.
3. By checking the first dependency (name → author) we can disqualify the BCNF form, because it is not trivial, and 'name' is not a superkey. (name)+ does not include the attribute 'year'.
   'Author' is not part of any key, so this scheme is not 3NF as well.
4. Submitted separately
5. has a contradiction: (name, year) → price, 3 rows
   Does not has a contradiction: name → author, name → genre, name,year → reviews, name,year → user rating.
   R1 = (name, author) - every book can have a single author

R2 = (name, genre) - every book can have a single genre
R3=(name, year, price) - every book for a certain year can only have a certain price
R4=(name, year, reviews) -
every book for a certain year can only have a amount of review
R5=(name, year, user rating) - every book for a certain year can only have a certain amount of ratings

Third question:

1. Let us choose (A,B,C) as the attributes of relation R, and define F as F = (A → B). We will choose the following decomposition: R1 = (A,B) and R2 = (C). This decomposition does keep F, because all pairs of (a,b) that appear at A, wil be at R1. But it is not lossless:

	A	B	C
R1	a1	a2	b13
R2	b21	b22	a3

   There is no row filled with ‘a’s, therefore this decomposition is not lossless.

2. Let us choose (A,B,C) as the attributes of relation R, and define F as F = (A → B, C → B).
   We will choose the following decomposition: R1 = (A,B) and R2 = (A,C). This decomposition does not keep F,

	A	B	C
R1	a1	a2	b13
R2	b21	b22	a3

   There is no row filled with ‘a’s, therefore this decomposition is not lossless.

3. The claim is True
   => Let R be in BCNF
   as we have seen previously if it is in BCNF therefore it must be in 3NF
   => Let R be in 3NF
   Let us split this into cases
    1. F is in an empty group therefore everything is trivial and therefore R belongs to BCNF
    2. F is not an empty group and therefore has at least one dependency such that for each dependency X-> A

For all X it can be one of two cases

1. X is a SuperKey
   (since we are in 3NF) and therefore it meets the terms of BCNF
2. A is an attribute in a Key
   let us prove that if this if the case A must belong to X
   We know from the details given to us that B,C,D,E cannot be on the right hand side of a dependency therefore they must be part of the key
   Therefore, A,B,C,D,E is a key Lets say that A does not belong to X and X is composed from letters in B,C,D,E
   Therefore, since X->A a key can be composed from B,C,D,E which contradicts the fact that A belongs to a key
   From this we get that A must belong to X which is trivial and therefore it is in the BCNF form