CoAssignment 1
Entity Relationship Modeling
Points: 50

Instructions: For this assignment you must submit a **picture** file(s) of the ER diagram that must be drawn using any ER modeling tool or any digital drawing tool. “Handmade” diagrams won’t be accepted. **I strongly recommend using TerraER but others can be used.**

We organize cycling races, for example, the “Tour de France” or “La Vuelta a España”.

We need to keep a database with the races, the teams, the riders, the stages and some details about the stages, including positions of the riders and times.

For those that don’t know how the cycling races and the classifications work let’s explain everything with detail:

- A race is defined by its name (unique) and a description
Example: Name: “Tour of France” Description: “The most famous race in the world, but it is more boring than the Giro and the Vuelta”

- A race can have different editions in different years (no more than one per year), The description will NOT change based on the edition.
Example: We can have the ”Tour de France” in 2020 and the “Tour deFrance” in 2021.

- A race will keep the information about the principal country where it happens. In case the race passes by more than one country, we will store just one (*for example, the Tour de France, sometimes goes by Italy or Spain for some miles, but we will just keep France as the main one*). It is weird, but it could change on a different edition. Maybe we have a “Red Bull Xtreme Race” that one year is in Spain and the next one in Sweden.

- A race has a category. *Examples of categories are “1.UWT”, “2.UWT”, “1.WWT” …* and we will store them just as a text. A race can belong to different categories in different editions.

- A race will keep the information about the number of riders allowed by team and the number of teams that can participate. This can change on each edition too.

- A race can have one or more stages. Each stage has a date and a unique order inside the

race. *(For example, Stage order: 7, Stage date: 12/08/2020)* We can infer the starting and ending date of the race based on the dates for the first and last stage.
Note: No need to store start and end date of the races… just the stages and the order in the race

-Each stage has a departing point and an arriving point. *For example, the Stage 9 in the Tour of France in 2020 started in PAU and ended in LARUNS.*

- Each stage has a length in kms. For example this one is 153 kms.

- Each stage can have (or not) some climbs, maybe a lot of them. For example, we can see in the previous image that the stage 9 in the Tour de France in 2020 had 5 climbs.
  - The Cote d'Arteguelouve. Height: 273m. Length: 2.3kms. Percent: 4.5%. Category: 4, Position: at the km 9.5
  - The Col de la Hourcere. 1440m, 11.1kms, 8,8%. Category 1 in the km 69
  - The Col de Soudet, Category 3, 1540m, 3.8km, 8,5%, in the km 78
  - The Col d'Ichere, Category 3, 674m, 4.2kms, 7%, in the km 115.5
  - The Col de Marie Blanque, Category 1, 1035m, 7.7km. 8.6% in the km 135

- The database will need to keep the information about those climbs: Height, Length, Percent, Category, Name (unique) and we will need to keep where they happen on the stage (the position). Important, those climbs can appear in different stages in different races, so we don't want to repeat that information each time it is used… ! *For example, The Col de Marie Blanque could be climbed in 2020 in the Tour of France, stage 9 in the km 135, and in the*

“Quebrantrahuesos, Stage 1, 2006” in the km 104.

- We can have a climb that maybe has never belonged to any stage.

- We will have cycling teams (with a team_id that is unique, a name, country and category).
For example: Name: Movistar, Country: Spain, Category: UWT.
Note: We will store all those fields as just text, we don’t need a countries table or categories table)

- A team has “persons” on it (at least one). A person can belong or not to a team.

- A person can be a rider or staff, and the staff people can have several roles. It is weird but maybe a rider can be staff too at the same time, so we will need to keep that in mind when designing the database.

- For any person (rider and/or staff) we will keep the person_id (unique), name, country, birth date, and we will need to keep (in some place) the date they start in a team and the date they end in a team. That means that a person can be in different teams during their career.

- If it is a rider we will keep the licenseID (unique for each rider), the height and weight. But we don’t need that if the person is only staff.

- A team must have at least one person.

- A team can participate in a race if it is invited. In the case it is invited, a team can maybe have 20 riders in total, but maybe the race only allows 8 of them, so we need to keep, in some way, the riders that have been enrolled in an specific race for that team. A race must have at least one team

- Only riders can ride in the races, not the staff (and the ER diagram must show that).

- We will keep 2 different classifications for each race.

1. Stage classification. For each rider in the race we will need to store if they started the stage, if they finished it, the position and the time it took for them to finish it.
For example, a rider with license ES30061976 started the stage 8 in the Tour of France 2020 and ended the stage 8 in the Tour of France 2020 in the position 23 with a time of 2h 34m 23s 123ms.

2. Climbing positions. For each of the climbs in the stage we will need to keep the first 10 riders and the order that they crossed the line at the top of the climbing.
Note, in the database design we don’t care if we store 10 or 100 or all of them… we just care that we need “more than one”. The software will insert 10, but as said, we don’t care about the

exact number.

NOTE: In the case that you know about cycling, maybe you are wondering about the general classification, or the total points classifications for the climbs or sprints… Well… the software will calculate it based on the stage times or the climb/sprint positions, ***so we won't need to store it or worry about it.*** *In a real system we surely would want to keep that to avoid complex queries to display the classifications, but not on this one so we can keep it simple and just store the "stage classifications".*

Design an entity-relationship diagram for the cycling races database described above. Along with all the relevant entities, relationships, and attributes make sure to specify the following:

- **Key** attributes of entities (remember, key attributes are underlined)
- **Cardinality ratio and participation** constraints for relationships (Preferred in
- Chen notation, that means double line or single line for participation and a number for cardinality, but other notations can be accepted too if your ER diagram tool has not Chen notation)
- **IMPORTANT: Any assumptions you make that are not in the requirements, but affect your design must be written in the document.**

RUBRIC:

20 points: for entities and their attributes

- -4 points for missing entity;
- -1 for missing attributes or incorrect attribute type
- -2 for missing keys
- -1 point if any weak entity is not defined correctly

15 points: for relationships with correct participating entities and attributes where applicable

- -3 points for missing relationship
- -1 for missing attributes or incorrect attribute type
- -2 for missing keys

10 points: for cardinality

- -2 points for missing or wrong cardinality

5 points: for participation constraints

- -2 points for missing or wrong participation constraint