

# Project for NEW GENERATION DATA MODELS AND DBMSs

A.A. 2020/2021

## 1. Domain description

Suppose you wish to manage the participation of players to golf competitions. For each golf player you wish to store the federal card number, name, surname, sex, age, golf club to which he is a member. Each golfer can participate in competitions only if he/she is member of a golf club (in reality a player could be registered in more than one club, but for simplicity we do not consider this possibility). Each player is assigned a playing handicap. The playing handicap value ranges from 0 (excellent player) to 18 (very poor). For each golf club we wish to store the name, the city, the telephone numbers and the number of members registered in the club. A golf course usually consists of 9 or 18 holes (in our case we consider golf courses of 9 holes). The aim of the game is to throw the ball into the hole with fewer strokes as possible (a better description of the golf game can be found at <https://en.wikipedia.org/wiki/Golf>). Each hole has a "par" that represents the "correct" number of strokes that need to be done for the hole. Moreover, each hole has an index (from 1 to 9) which represents the level of difficulty of the hole. This index is used along with the golfer handicap for determining the "player par" of the hole. Suppose that a player has an handicap of 18, it means that for each hole of the course, his player par for the hole is the par plus 2, whereas if the handicap is 5, the player has a par plus 1 for the first 5 most difficult holes, whereas for the others he has the hole par. There are many scoring systems that can be adopted for ranking the players of a competition. In our case, we adopt the "stableford" scoring system. In this system, rather than counting the total number of strokes in the entire course, involves scoring points based on the number of strokes taken at each hole. The number of points awarded on each hole is determined based on the comparison of the number of strokes taken to the par. The par is then adjusted in relation to the player's handicap. Once players have taken two strokes more than the "player par", they may abandon the hole and move on to the next, as it is then not possible to score any points on that hole. At the end of the round, the number of points scored on each hole is summed up to give a final score. The winner of a Stableford competition is the player with the highest point total. The points to be assigned depends on the player par of each hole according to the following table:

5 point	3 strokes under the player par
4 point	2 strokes under the player par
3 point	1 stroke under the player par
2 points	Player par
1 point	1 stroke over the player par
0	2 or more strokes over the player par

Clubs use an "electronic score" to storing the number of strokes a player takes for each hole. A player has the option to withdraw at any hole in the course. In this case the score is missing the strokes of the hole in which he retired and also in the following ones. Moreover, it is marked that the player has retired. A retired player cannot participate in the ranking. At the end of each match, the player delivers the electronic score that produces an XML document that is used for the management of the match. An example of this document is shown in Figure 1 (the relative XML schema is attached to this document).

A competition is characterized by the name (which is unique in a given year), the sponsor, the date in which the competition is taken. The points of the various players that participate to the competition are determined relying on the scoring system previously described that determines the ranking of the competition based on the category they belong to. There are usually 3 categories "first", "second", "third" based on the handicap value; there may be transversal categories such as the "women" category or the "over" category of a certain age. The structure of the categories of a competition are established on the basis of the information

---

```

<?xml version="1.0" encoding="UTF-8"?>
<Golfscore xsi:noNamespaceSchemaLocation="golfscore.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Name Id="324">President's Cup</Name>
  <Player>FC1001</Player>
  <Results>
    <Hole Number="1" par="3" difficulty="9">3</Hole>
    ... <Hole Number="2" par="4" difficulty="3">5</Hole>
    <Hole Number="3" par="4" difficulty="4">4</Hole>
    ... <Hole Number="4" par="5" difficulty="2">6</Hole>
    <Hole Number="5" par="3" difficulty="7">4</Hole>
    ... <Hole Number="6" par="4" difficulty="8">5</Hole>
    <Hole Number="7" par="4" difficulty="6">4</Hole>
    ... <Hole Number="8" par="5" difficulty="1">3</Hole>
    <Hole Number="9" par="3" difficulty="5">4</Hole>
  </Results>
</Golfscore>

```

---

*Figure 1 Electronic score of the player with Federal identifier FC1001 for the competition "President's Cup"*

contained in the XML document shown in Figure 2 (the relative XML schema is attached to this document). This XML document is provided by the sponsor of the competition and allows the management of the competition according to the parameters specified therein. When the XML document is introduced in the system, it is mandatory to check the consistency with respect to the XML schema. Specifically, the document in Figure 2 indicates that this is the "President's Cup" competition sponsored by Lavazza SpA for which there are three categories with 3 prizes for the first and second category and only 2 prizes for the third category. There is also the lady category for ladies and the over category for all players over the age of 40. Note that if the "reserved" element is present it indicates that the competition is restricted only to the club members. The category (s) they belong to is determined automatically based on the characteristics of the player.

We are interested in storing the player results in the context of the participation to a competition. For each competition in which a player participates, it is important to store the strokes made in each hole of the course and the position in the ranking. A ranking position is a pair (category, number). For example ('first', 2) indicates that the player qualified second in the first category.

---

```

<?xml version="1.0" encoding="UTF-8"?>
<GolfCompetition xsi:noNamespaceSchemaLocation="GolfCompetition.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Name> President's cup</Name>
  <Date>2009-12-25</Date>
  <Sponsor>Lavazza S.p.A</Sponsor>
  <Category NumPrize="3" From="0" To="12" Type="First"/>
  <Category NumPrize="3" From="13" To="24" Type="Second"/>
  <Category NumPrize="2" From="25" To="36" Type="Third"/>
  <Category NumPrize="1" Type="Lady"/>
  <Category NumPrize="1" Type="Over" Age="40"/>
</GolfCompetition>

```

---

*Figure 2 XML document containing the organization of the competition*

## 2. What to do

The following items should be carried out.

- A. Develop a conceptual design of the domain proposed above, creating a UML class diagram. The diagram must represent in the most appropriate way the constraints imposed by the text. Furthermore, all constraints that cannot be expressed with the UML schema must be highlighted in natural language. Report all the assumptions / simplifications that you intend to adopt.
- B. Propose a translation of the conceptual model into the Oracle object-relational model by exploiting the best characteristics of the system in dealing with objects and XML documents. The following workload should be taken into account in the development of the logical model:
  - a. Create the calendar of the competitions of a club in a given year. The calendar must be an XML document and contain all the information about the competitions. The calendar should be organized in ascending order with respect to the date of the competitions.
  - b. Determine the information about the club a given golfer is enrolled in.
  - c. Create a competition starting from the XML document contained in Figure 2.
  - d. Determine golfers who only participate in competitions organized by their club for their members only.
  - e. For each competition, determine the golfers who did not finish the game. Sort the list according to the number of holes that have been completed.
  - f. Create an XML document containing the ranking of a competition. The document must contain the winners for each category that exist in the competition. It is possible that a player can be a winner in several categories (for example, a lady with a handicap of 5 over 40 years old can be second in the first category and win the Lady and Over 40 prize). It is necessary to indicate which players have played below the "par" (this is determined by considering the players that provide at least a point for each hole and that the total number of points is above 18).
- C. Write the SQL statements for the creation of the database and for its population with enough data to be able to verify the developed system.
- D. Implement the queries of point b in SQL. Make the most of SQL / XML and XQuery language. Note that a single query may not be enough.

## 3. What to deliver

The project can be delivered at any time of the academic year and is valid until April 2022. After this date a new project will be assigned.

The project must be carried out by groups of at most 2 people.

For students who submit the project by March 2021, a bonus of 1 point will be given.

You have to submit a zip file containing the following documents:

1. A technical documentation containing:
  - a. The UML diagram complete with explanations, motivations, constraints and any other information that allows understanding the design carried out. The diagram must be labeled with the direction of traversal of the associations which takes into account the assigned workload.
  - b. The translation of the object model into Oracle's object-relational model c. The SQL queries required in the workload and the implementation of the main constraints
2. A script for the creation of the database, its population, the implementation of the query associated with the main constraints (i.e. there is no need to develop triggers but just the queries for checking the constraints). Finally; the SQL queries required in the workload.

## 4. Alternative project

Students, interested in making experience in other advanced data models, can decide to develop the same project by exploiting:

1. Mongo DB (in this case scores are represented as JSON documents)
2. Cassandra
3. Neo4j