**The Competition**

Thornlie swim club has a summer series intra-club swimming Competition for its swimmers aged 13 to 16.

The competition takes place over six (6) Friday nights (called a Racemeet) during January and February each year.

Swimmers race in eight (8) Divisions according to their age and sex.

* Division 1: 13's - boys
* Division 2: 13's - girls
* Division 3: 14's - boys
* Division 4: 14's - girls
* Division 5: 15's - boys
* Division 6: 15's - girls
* Division 7: 16's - boys
* Division 8: 16's - girls

**The Races**

Races are classified as Events, of which there are five (5) in each division.

* Event 1: 50 metres Freestyle
* Event 2: 50 metres Breaststroke
* Event 3: 50 metres Butterfly
* Event 4: 50 metres Backstroke
* Event 5: 4 x 50 metres Medley

The events are either an individual stroke race (i.e., freestyle, breaststroke, butterfly, backstroke) or the combined strokes race (i.e., the medley - where swimmers swim all 4 strokes in one race).

The races take place by race-type/stroke, running through all divisions (i.e., all age-groups) in that race,   
for e.g., Event 1: 50 metres Freestyle; 5:30 - 6:00pm.

race 1: 50m Freestyle, 13's boys;

race 2: 50m Freestyle, 13's girls;

race 3: 50m Freestyle, 14's boys;

race 4: 50m Freestyle, 14's girls;

race 5: 50m Freestyle, 15's boys;

race 6: 50m Freestyle, 15's girls;

race 7: 50m Freestyle, 16's boys;

race 8: 50m Freestyle, 16's girls;

followed by

Event 2 (50 metres Breaststroke); 6:00pm - 6:45pm

Event 3 (50 metres Butterfly); 6:45pm - 7:15pm

Event 4 (50 metres Backstroke); 7:20pm - 7:50pm

Event 5 (4 x 50 metres Medley); 8:00pm - 9:00pm

This allows swimmers time to recover if they have nominated to swim a second event and/or medley.

**Swimmers and Races**

A swimmer can nominate to compete in up to three (3) events each week, provided one of those events is the combined stroke medley race. That is, two (2) individual stroke races each week, plus a third medley race. Thus, even though division have more than 8 swimmers within the division (most divisions have between 10 and 15 swimmers) do not need to run 'heats' for races, since swimmers often specialise in 1 or 2 strokes/events. Those swimmers who have ability in more than 2 strokes are the ones who nominate to also compete in their divisions medley.

There are eight (8) lanes in each event, and thus a maximum of 8 swimmers can nominate for a race. There are no races with less than five swimmers – and most races are full. Swimmers are allocated lane numbers in each of their races randomly by the officials running each racemeet.

**Race Results and Competition Points**

Swimmers are awarded points according to the position they finish in each race using a sliding scale:

for e.g., 1st place = 8 points.

2nd place = 7 points;

3rd place = 6 points;

4th place = 5 points;

5th place = 4 points;

6th place = 3 points;

7th place = 2 points;

8th place = 1 points;

**Extra/Specific Details to help with ERD design, Data design (what data needs to be typically captured)**

There is a coach for each of the eight (8) divisions in the competition

There are 7 officials at each racemeet (running the events)

* 1 x Starter who readies the swimmers at the beginning of the race and triggers the starter's beep.
* 2 x Scrutineers on each end of the pool who watch that each swimmer's turns and/or end-of-race touch are legal
* 2 x Admin/Results officials capturing racer's races, lanes and results etc.,

There are 40 races per night/racemeet (approx. times for events are listed on previous page), across 5 events.

Over the 6-week competition there are 30 events and 240 races.

## 1. Design / Conceptual DB

### 1.1 Entity Relationship Diagram

### 1.2 Business Rules

Instruction: Write all Business Rules according to the ERD you have designed for the case.

HELP: Each Business Rule describes a relationship between 2 entities in real everyday language, first describing the relationship from entity 1 to entity 2, and then describing the reverse (how entity 2 relates to entity 1).

HELP: There should be twice as many business rule statements as there are relationships in your ERD (with each relationship requiring two (2) business rule statements

HELP: Since there should be no M:N (many to many) relationships in your final ERD, do not describe M:N relationships in the final business rules

## 2. Schema / Logical DB Schema

### 2.1 DDL-Schema

Instruction: Develop a DDL-Schema that includes the names of all tables, columns, PKs and FKs.

HELP: The point of the DDL-schema is to write a schema that will inform your Data-Dictionary and CREATE TABLE SQL, Thus, present your tables in the DDL-schema in the correct CREATE TABLE order.

### 2.2 Data Dictionary

Instruction: Develop a Data-Dictionary that includes all the names of tables, columns, PKs, FKs and Constraints, and all the column and tables referred to when named in a Constraint.

HELP: Below are some suggested templates for a Data-Dictionary for students to use. 1-Remove any/all table-templates you don't use. 2-Remove example text inside the used table-template. 3-Remove this instruction

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table Name** | **Column Name** | **Description** | **Format** | **Opt/Require** | **PK Name** | **Column Reference** | **FK Name** | **Table Reference** |
| Staff | staffID | Unique identification of a Staff member | VARCHAR(7) | NOT NULL | StaffID\_PK | staffID |  |  |
|  | staffLastName | Surname/Family name of Staff member | VARCHAR(55) | NOT NULL |  |  |  |  |
| Event | eventID | Unique identification of an event | VARCHAR(7) | NOT NULL | eventID\_PK | eventID |  |  |
|  | staffID | Staff member affiliated with the event | VARCHAR(7) | NOT NULL |  | staffID | Event\_staffID\_FK | Staff |
|  | etc. |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table Name** | **Column Name** | **Description** | **Domain** | **Opt/Req** | **PK Name** | **Column Reference** | **FK Name** | **Table Reference** |
| Staff | staffID | Unique identification of a Staff member | VARCHAR(7) | NOT NULL | StaffID\_PK | staffID |  |  |
|  | staffPhone | mobile (24/7) contact of competitor | NUMBER(10) |  |  |  |  |  |
| Event | eventID | Unique identification of an event | VARCHAR(7) | NOT NULL | eventID\_PK | eventID |  |  |
|  | staffID | Staff member affiliated with the event | VARCHAR(7) | NOT NULL |  | staffID | Event\_staffID\_FK | Staff |
|  | etc. |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Table Name** | **Column Name** | **Description** | **Format/Domain** | **Opt/Require** | **PK Name** | **Column Reference** | **FK Name** | **Table Reference** |
| Staff | staffID | Unique identification of a Staff member | VARCHAR(7) | NOT NULL | StaffID\_PK |  |  |  |
|  | staffLastName | Surname/Family name of Staff member | VARCHAR(55) | NOT NULL |  |  |  |  |
| Event | eventID | Unique identification of an event | VARCHAR(7) | NOT NULL | eventID\_PK |  |  |  |
|  | staffID | Staff member affiliated with the event | VARCHAR(7) |  |  |  | Event\_staffID\_FK | Staff |
|  | etc. |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |

## 3. SQL Implementation

### 3.1 Create Table SQL Syntax (Heading 3, Segoe UI, size=11 RGB(204,102,0), before=8, after=0)

Instruction: Write and present the SQL of the CREATE TABLE syntax you develop in order to build your database in your Oracle profile.

HELP: Your tables should be created in an order that allows for the proper construction of FKs and FK constraints as you build your tables. Constraint names should match the data-dictionary, both in the SQL and when the marker checks each table's construction and constraints in the Oracle Object-Browser when being marked.

HELP: Template Optional – some optional table-name headings (Heading 4) have been included in the template below. If you use them, please ensure every table uses the Heading-4 format (open your "styles" window). If you do not use these sub-headings, please remove them all. PRESENTATION MARKS ARE AWARDED/REMOVED according to how well you use (or don't use) the template.

HELP: Each (and every) SQL script should be pasted into the spaces provided below. Use the "Syntax" Style in the Styles (open your "styles" window". The Syntax Style is demonstrated below.

This is the requested formatting for any syntax/SQL presented in this document. Use the "Syntax" style in Styles of this template. (Courier New, size=10.5, colour=RGB(0,0,153), before=0, after-1, indent=20)

HELP: Each SQL script should be accompanied by the appropriate (good quality, good sized/cropped) screen-grab of the result of the SQL. "Appropriate Screen-grab" in the case of CREATE TABLE syntax is the actual table that has been created (go to the Object-Browser to view the table you created and click on table tab. Use the provided "ISYS2007.Resource-ScreenGrabsOfSQLresults.pdf" for more help to create/present your screen-grabs

*Optional headings to separate tables (name the table) – remove this instruction*

#### Table name (Heading 4, Segoe UI, size 11, RGB (134, 67, 0), before=0, after=0)

#### Table name

### 2.2 INSERT data SQL Syntax (Heading 3, Segoe UI, size=11 RGB(204,102,0), before=8, after=0)

Instruction: Write and present the SQL of the CREATE TABLE syntax you develop in order to build your database in your Oracle profile.

HELP: Make sure you have thoroughly read Part 4 regarding what SQL SELECT are required so that you understand the type of data you need to create/capture. This is a 6-week competition involving 8 divisions, racing 5 events each racemeet. You will need to create the whole competition's (240 races) in your database to make the SQL SELECT queries work.

HELP: Your tables should be created in an order that allows for the proper construction of FKs and FK constraints as you build your tables. Constraint names should match the data-dictionary, both in the SQL and when the marker checks each table's construction and constraints in the Oracle Object-Browser when being marked.

HELP: Each SQL script should be accompanied by the appropriate (good quality, good sized/cropped) screen-grab of the result of the SQL. "Appropriate Screen-grab" in the case of INSERT INTO syntax is the actual table of data/rows created as a result of the script. Either go to the Object-Browser to view the table you created and click on data tab, or write SELECT \* FROM TabeNameHere; Use the provided "ISYS2007.Resource-ScreenGrabsOfSQLresults.pdf" for more help to create/present your screen-grabs

*Optional headings to separate tables (name the table) – remove this instruction*

#### Table name

#### Table name

## 4. SQL Statements/Exercises

### 4.1 SQL testing/statements

#### (1) Whole Season/Competition Query

HELP: Each SQL script should be accompanied by the appropriate (good quality, good sized/cropped) screen-grab of the result of the SQL. "Appropriate Screen-grab" in the case of SELECT syntax is the actual table of results of the query, including ALL ROWS of the result and including the statement at the bottom of the table indicating how many results/rows there were to the query. You WILL LOSE MARKS without this statement at the bottom of the result AND all the rows.

There are 40 races each Friday evening (called a "Racemeet") and six (6) weeks of racemeets for a total of 240 races across the season/ competition.

1. Write a select query that will generate a table to display every race that was swum across the whole 6-week competition. The result needs to include the following meaningful information:
2. which racemeet the race was swum (i.e., which week of the competition);
3. the raceID/Number of the evening/racemeet the race took place;
4. time the race took place;
5. race division (age-group and sex);
6. race stroke/type;
7. winner of the race

List the races in race-meet, followed by race number order.

1. Run the same query, this time list the races by division, race-meet, race number

#### (2) One Swimmer's and one Racemeet's story:

During Racemeet No.2, a 13 y/o boy nominated to swim in his division's Freestyle, Butterfly and Medley races:

1. Write a query so he knows what time each of his races will be.
2. Write a query that lists his results for the evening (list his name, the positions he came in each of his races, the times he swam and how many points he scored for each race).
3. Write a query to learn which 13 y/o boy performed the best (i.e., scored the most points in the Division) during raceMeet/week-2.   
   List the boy's name, number of races swum (by the boy) that night, which strokes were swum and the total number of points the boys scored for the evening).

Your SQL should be sophisticated enough that it finds out and lists this information/data as part of the same query. That is, if you changed the results within table-results tables within the database, the answer in the developed SQL (asking for the best 13 y/o boy at racemeet/week-2) would also change. Thus, do not rely on going to find the answer first and then use the numbers/answer to then list details about the specific boy.

HINTS: This is likely a nested and/or sub-query. Use the Bordoloi & Bock textbook to help you develop the SQL to answer this query.

HINT: (data population) - within the 13 y/o boys division there are 11 boys. A couple of boys specialise in one stroke (only swim one race per Racemeet/week). Most boys swim at least 2 races/strokes, and some swim 2 stokes plus the medley. No race each Reacemeet/week has less than 5 swimmers. No race has more than 8 swimmers (there are only 8 lanes in any race).

#### (3) Division winners across the whole competition.

Write a query that will provide the winners of each division across the whole 6 week competition.

There are eight (8) divisions, who won each division? The 8 results should include

* The swimmer's (who won) details
* The Division they won
* The total number of points scored for the competition
* The number of races the swimmer swam over the competition

#### (4) Season's BEST PERFORMANCES (best times for events/strokes)

Write a query that will provide the following - Across the course of the season/competition

HINT: (data population). Times for swims need to be realistic. It will be worth using google to find some typical swim times for each of the four (4) strokes and the medley across Australia/World - and make your times a little slower - - this is only a local competition for 12 to 16 year old's, not the world championships etc.

1. Season's best Freestyle result.

include:

* who swam the fastest freestyle race of the whole season
* which racemeet/week it occurred at
* what division the swimmer is from
* what the time was

HINT: These bullet points are in the same query, and thus the same resulting table of information from the query.

1. Season's best Breaststroke result.

include:

* who swam the fastest breaststroke race of the whole season
* which racemeet /week it occurred at
* what division the swimmer is from
* what the time was

1. Season's best Butterfly result:

include:

* who swam the fastest butterfly race of the whole season
* which racemeet /week it occurred at
* what division the swimmer is from
* what the time was

1. Season's best Backstroke result:

include:

* who swam the fastest backstroke race of the whole season
* which racemeet /week it occurred at
* what division the swimmer is from
* what the time was

1. Season's best Medley result

include:

* who swam the fastest butterfly race of the whole season
* which racemeet /week it occurred at
* what division the swimmer is from
* what the time was

# ~ End of Assessment ~