

UNIVERSITY AT BUFFALO  
DATA SCIENCES AND APPLICATIONS

**PROJECT IMPLEMENTATION**

"The Super Awesome Swimmers Club".

Database System

**COURSE:** CDA502 - Database Management Systems

**SEMESTER:** 1<sup>st</sup> Semester - Fall 2025

**TEAM NAME:** Team 8

**TEAM MEMBERS:**

- ◆ Rameen Mufti – Presentation Lead - [rameenmu@buffalo.edu](mailto:rameenmu@buffalo.edu)
- ◆ Dharshini Venkataramani – Technical Lead – [dvenkata@buffalo.edu](mailto:dvenkata@buffalo.edu)
- ◆ Meena Murugesan – Team Coordinator - [meenamur@buffalo.edu](mailto:meenamur@buffalo.edu)

## TABLE OF CONTENTS

INTRODUCTION .....	2
Summary.....	2
DEFINING ENTITIES AND ATTRIBUTES.....	2
METADATA TABLES.....	4
Employee Entity .....	4
Pool Entity .....	5
Maintenance_Schedule.....	6
Course Entity .....	6
Registration Entity .....	7
Swimmer Entity.....	8
Swimmer_Guardian Entity .....	9
Guardian_Info Entity.....	9
Charity Entity .....	10
Charity_Neighborhood Entity .....	10
Charity_Manager Entity.....	11
Competition Entity .....	11
Competition_Schedule Entity.....	12
Supplier Entity .....	12
Supplier_Specialty Entity .....	12
Bill Entity .....	13
Product Entity .....	13
Supply Schedule Entity .....	14
Order Line Entity .....	14
Order Entity .....	15
Developer_Profile Entity .....	15
Developer_Contract Entity.....	16
BUSINESS RULES.....	17
IDENTIFYING PKs and FKs.....	18
ENHANCED ENTITY RELATIONSHIP DIAGRAM .....	19
RELATIONAL SCHEMA .....	20
IDENTIFYING FULL, PARTIAL AND TRANSITIVE DEPENDENCIES.....	21
ORACLE SCREENSHOTS:.....	22
Table and SQL statements .....	22
Section 2: Query 1 .....	32
Section 2: Query 2 .....	32

Section 2: Query 3 .....	32
Section 3: Advanced Query .....	33
INTEGRITY CONSTRAINTS.....	33
Domain Integrity .....	33
Referential Integrity .....	33
Entity Integrity.....	34
CONCLUSION .....	34

# INTRODUCTION

## Summary

This project focuses on designing, normalizing, and implementing a complete relational database system for *The Super Awesome Swimmers Club*. We began by identifying all business rules, defining the necessary entities, creating detailed metadata tables, identifying keys, checking dependencies, and constraints. Using this information, we constructed an Enhanced Entity Relationship (EER) diagram and relational Schema that capture all major processes of the club, including swimmer registration, course scheduling, maintenance, supplier relations, billing, competitions, and charity-based discounts.

During normalization (up to 3NF), two new entity groups, **Developer** and **Charity**, were added to support pool construction, contract tracking, population-based discount rules, and charity management. The final normalized schema was then implemented using SQL in Oracle APEX, enforcing primary keys, foreign keys, data types, and constraints to ensure full domain, referential, and entity integrity.

# DEFINING ENTITIES AND ATTRIBUTES

## EMPLOYEE

An Employee is a staff member responsible for supporting the club's operations, which may include sub-roles such as trainer, accountant or maintenance person. It includes details such as employee id, first name, last name, primary contact number, job role, and a coaching certificate in case of a trainer.

## MAINTENANCE\_SCHEDULE

A Maintenance contract is an agreement for a maintenance person to handle regular inspections and repairs. It outlines the pool number, maintenance person's employee id along with contract date and time to ensure the pool remains safe and functional.

## POOL

A Pool is the main facility used for swimming training, practice, and competitions. It is designed with maintenance contracts to provide a working environment for swimmers. This includes the details like pool number, capacity, condition of the pool - whether indoor or outdoor, who maintains the pool and the contract number.

## COURSE

A Course is a training program developed to teach or improve swimming skills. It includes session timings from a qualified trainer with a proper coaching certificate. We have the following details for a course such as course id, course name, course day & time, course's maximum capacity (typically 20), instructor id along with the pool number for the class.

## REGISTRATION

Course Registration represents the enrollment of a swimmer in a specific swimming course offered by the club. It includes the details of swimmers by their id and the course id offerings they registered for.

## ORDER

An Order represents an official request made by the club's maintenance staff to purchase one or more products required for cleaning or repair activities. It includes details such as the order date, the products requested, quantities, and the supplier from whom the items are ordered.

### ORDER\_LINE

An Order\_Line represents the individual products included within a specific order placed by a maintenance employee. Each line records the product requested and the quantity needed, allowing an order to contain multiple different products.

### PRODUCT

A Product refers to any item, material, or equipment used for swimming pool cleaning, maintenance, or repair. Each product record includes attributes such as product name, description, unit price, and supplier information.

### SUPPLIER

A Supplier is an external business partner that provides the club with products or materials required for maintenance operations. Suppliers receive purchase orders from the club and are responsible for delivering the requested products.

### BILL

A Bill (or Invoice) is a financial document issued by a supplier to the club after delivering ordered products. It specifies the products supplied, quantities, prices, applicable taxes, and total payment due, and is processed by the club's accountant.

### SUPPLY\_SCHEDULE

A Supply Schedule records the delivery of arrangements between suppliers and products, including the date and product supplied.

It helps track when and by which supplier a product was delivered.

### SWIMMER

A Swimmer represents an individual who registers to participate in the swimming club's programs. Each swimmer can enroll in one or more courses offered by the club, may participate in competitions if over 18 years of age and can refer other swimmers to the club for discounts. Swimmers under 18 need guardian approval.

### SWIMMER GUARDIAN

Swimmer\_Guardian records the authorized parents or guardians assigned to a minor swimmer. It ensures that swimmers under 18 have a designated guardian, supporting the rule that minors must be accompanied when attending classes.

### GUARDIAN\_INFO

Guardian\_Info stores personal details of each parent or guardian responsible for minor swimmers. It contains identifying and contact information used to verify authorized guardians in the system.

### COMPETITION

A Competition represents an official swimming event or contest that eligible swimmers (aged 18 and above) can participate in. Each competition record includes details such as the competition name and the award or prize associated with it.

### COMPETITION\_SCHEDULE

Competition\_Schedule represents the participation of adult swimmers in various competitions organized or endorsed by the club. It links each competition to the swimmers who are eligible (18+), capturing when and where they participate.

### SUPPLIER\_SPECIALTY

Supplier\_Specialty identifies the specific product types or service categories that each supplier is qualified to provide. It supports proper supplier-product matching by recording each supplier's area of specialization.

## DEVELOPER PROFILE

A Developer Profile represents a company or organization responsible for constructing swimming pools for the club. It contains key information about the developer, such as the company name and primary contact details.

## DEVELOPER CONTRACT

A Developer Contract represents a formal agreement between the swimming club and a developer company for building a specific pool. It captures contract details, warranty coverage, and the items covered under the warranty.

## CHARITY

The Charity entity stores information about each charity organization partnered with the club, including, charity name, neighborhoods served, and assigned manager.

## CHARITY\_NEIGHBORHOOD

Stores each neighborhood served by a charity and the approximate population of that area. Used to support population-based discount rules and validate charity service locations.

## CHARITY\_MANAGER

The Charity\_Manager entity records details about managers overseeing each charity, including their unique manager ID, name and email address.

# METADATA TABLES

## Employee Entity

Data Item Name	Data Item Type	Fixed/ Variable	Max Data Item Length	Required/ Optional	Metadata Min	Metadata Max	Metadata Description	Source
Employee_ID	Numeric	Fixed	4	Required	0001	9999	Unique identification number for each employee	HR database
Employee_First_Name	Char	Variable	30	Required	NA	NA	First name of an employee	HR database
Employee_Last_Name	Char	Variable	30	Required	NA	NA	Last name of an employee	HR database
Employee_Primary_Contact	Char	Fixed	11	Required	NA	NA	Primary contact number of an employee	HR database
Coaching_Certificate	Char	Variable	20	Optional	NA	NA	Certificate level of an employee who is	HR database

							eligible for teaching	
Job_Role	Char	Variable	15	Required	NA	NA	Designation of an employee at the club	HR database

### Pool Entity

Data Item Name	Data Item Type	Fixed or Variable	Max Data Item Length	Required or Optional	Metadata Min	Metadata Max	Metadata Description	Source
Pool_Number	Numeric	Fixed	3	Required	001	500	Unique number of a pool to be identified	Club's Facility database
Pool_Capacity	Numeric	Fixed	2	Required	01	50	Maximum number capacity a pool can accommodate	Club's Facility database
Pool_Condition	Char	Variable	15	Required	NA	NA	Working condition of a pool whether usable or under repair	Club's Facility database
Pool_Feature	Char	Variable	10	Required	NA	NA	Special feature of pool whether indoor or outdoor	Club's Facility database

### Maintenance\_Schedule

Data Item Name	Data Item Type	Fixed/Variable	Max Data Item Length	Required/Optional	Metadata Min	Metadata Max	Metadata Description	Source
Maintenance_Record_ID	Numeric	Fixed	10	Required	NA	NA	Maintenance schedule unique identification	Club's Maintenance system

							ion number	
Maintenance_Employee_ID	Numeric	Fixed	4	Required	0001	9999	Unique identification number for each maintenance employee	HR database
Pool_Number	Numeric	Fixed	3	Required	001	500	Unique number of a pool to be identified	Club's Facility system

### Course Entity

Data Item Name	Data Item Type	Fixed/ Variable	Max Data Item Length	Required/ Optional	Metadata Min	Metadata Max	Metadata Description	Source
Course_ID	Numeric	Fixed	2	Required	01	75	Unique identification for a course	Club's Courses or Training list system
Course_Name	Alphanumeric	Variable	25	Required	NA	NA	Name of the swimming course	Club's Courses or Training list system
Course_Day	Char	Variable	10	Required	NA	NA	Days of the week when the course is available	Club's Courses or Training list system
Course_Time	Char	Variable	15	Required	NA	NA	Time of the course when its available	Club's Courses or Training list system

Course_capacity	Numeric	Fixed	2	Required	01	20	Maximum number capacity of a course that swimmers can take	Club's Courses or Training list system
Pool_Number	Numeric	Fixed	3	Required	001	500	Unique number of a pool to be identified	Club's Facility system
Instructor_Employee_ID	Numeric	Fixed	4	Required	0001	9999	Unique identification number for each instructor	HR database

### Registration Entity

Data Item Name	Data Item Type	Fixed/ Variable	Max Data Item Length	Required/ Optional	Metadata Min	Metadata Max	Metadata Description	Source
Course_ID	Numeric	Fixed	2	Required	01	75	Unique identification for a course	Club's Courses or Training list system
Swimmer_ID	Numeric	Fixed	4	Required	0001	9999	Unique identification number for a swimmer	Club's Membership database

### Swimmer Entity

Data Item Name	Data Item Type	Fixed/ Variable	Max Data Item Length	Required/ Optional	Metadata Min	Metadata Max	Metadata Description	Source
Swimmer_ID	Numeric	Fixed	4	Required	0001	9999	Unique identification number for a swimmer	Club's Membership database

Swimmer_First_Name	Char	Variable	30	Required	NA	NA	First name of a swimmer	Club's Membership database
Swimmer_Last_Name	Char	Variable	30	Required	NA	NA	Last name of a swimmer	Club's Membership database
Swimmer_Age	Numeric	Fixed	2	Required	10	60	Age of a swimmer	Club's Membership database
Courses	Numeric	Fixed	2	Required	01	75	Unique identification for a course	Club's Membership database
Reference_ID	Char	Fixed	4	Optional	0001	9999	Unique identification number of a swimmer who gave referral	Club's Membership database
Swimmer_Type	Char	Variable	15	Required	NA	NA	Type of swimmer whether Adult or Minor based on age	Club's Membership database
Charity_Name	Char	Variable	50	Required	NA	NA	Official name of the charity organization	Charity organization database or portal.
Charity_Contact	Char	Variable	30	Optional	NA	NA	Primary contact number for the charity	Charity organization database or portal.
Charity_Neighborhood	Char	Variable	50	Required	NA	NA	Neighborhood(s) that the charity provides support.	Charity organization database or portal.

### Swimmer\_Guardian Entity

Data Item Name	Data Item Type	Fixed/Variable	Max Data Item Length	Required/Optional	Metadata Min	Metadata Max	Metadata Description	Source
Minor_Swimmer_ID	Numeric	Fixed	4	Required	0001	9999	Unique identification	Club's Members

							number for a swimmer	hip database
Guardian_ID	Numeric	Fixed	4	Required	0001	9999	Unique identification number for a swimmer	Club's Membership database

### Guardian\_Info Entity

Data Item Name	Data Item Type	Fixed/Variable	Max Data Item Length	Required/Optional	Metadata Min	Metadata Max	Metadata Description	Source
Guardian_ID	Numeric	Fixed	4	Required	0001	9999	Unique identification number for a swimmer	Club's Membership database
Guardian1_First_Name	Char	Variable	30	Optional	NA	NA	First name of Guardian 1	Club's Membership database
Guardian1_Last_Name	Char	Variable	30	Optional	NA	NA	Last name of Guardian 1	Club's Membership database
Guardian2_First_Name	Char	Variable	30	Optional	NA	NA	First name of Guardian 2	Club's Membership database
Guardian2_Last_Name	Char	Variable	30	Optional	NA	NA	Last name of Guardian 2	Club's Membership database
Guardian1_Phone #	Char	Fixed	11	Optional	NA	NA	Primary phone number of Guardian 1	Club's Membership database
Guardian2_Phone #	Char	Fixed	11	Optional	NA	NA	Primary phone number of Guardian 2	Club's Membership database

## Charity Entity

Data Item Name	Data Item Type	Fixed/ Variable	Max Data Item Length	Required/ Optional	Metadata Min	Metadata Max	Metadata Description	Source
Charity_Name	Char	Variable	50	Required	NA	NA	Official name of the charity organization	Charity organization database or portal.
Charity_Neighborhood	Char	Variable	50	Required	NA	NA	Neighborhood(s) that the charity provides support.	Charity organization database or portal.
Charity_Contact	Char	Variable	30	Optional	NA	NA	Primary contact number for the charity	Charity organization database or portal.
Manager_ID	Numeric	Fixed	4	Required	0001	9999	Unique identification number assigned to the charity manager organization	Charity organization's HR database

## Charity\_Neighborhood Entity

Data Item Name	Data Item Type	Fixed / Variable	Max Data Item Length	Required/ Optional	Metadata Min	Metadata Max	Metadata Description	Source
Neighborhood	Char	Variable	50	Required	NA	NA	Neighborhood(s) that the charity provides support.	Charity organization database or portal.
Approx_Neighb_Population	Numeric	Fixed	5	Optional	00001	99999	Approximate population size of the neighborhood(s) served	Charity organization database or portal.

## **Charity\_Manager Entity**

Data Item Name	Data Item Type	Fixed/ Variable	Max Data Item Length	Required/ Optional	Metadata Min	Metadata Max	Metadata Description	Source
Manager_ID	Numeric	Fixed	4	Required	0001	9999	Unique identification number assigned to the charity manager organization	Charity organization's HR database
Manager_Name	Char	Variable	40	Required	NA	NA	Full legal name of the charity manager	Charity organization's HR database
Manager_Email	Char	Variable	30	Optional	NA	NA	Email address of the charity manager	Charity organization's HR database

## Competition Entity

Data Item Name	Data Item Type	Fixed/ Variable	Data Item Length	Required/ Optional	Metadata Min	Metadata Max	Metadata Description	Source
Competition_Name	Alphanumeric	Fixed	10	Required	NA	NA	Name of a competition	Competition Event records database
Competition_Award	Numeric	Variable	3	Optional	100	750	Award amount or prize given for a competition	Club's Membership database

## Competition Schedule Entity

Data Item Name	Data Item Type	Fixed/ Variable	Data Item	Required/	Metadata Min	Metadata Max	Metadata Description	Source
----------------	----------------	-----------------	-----------	-----------	--------------	--------------	----------------------	--------

Adult_Swimmer_ID	Numeric	Fixed	4	Required	0001	9999	Unique identification number for a swimmer	Competition registration portal
Competition_Name	Alphanumeric	Fixed	10	Required	NA	NA	Name of a competition	Competition registration portal

### Supplier Entity

Data Item Name	Data Item Type	Fixed/Variable	Max Data Item Length	Required/Optional	Metadata Min	Metadata Max	Metadata Description	Source
Supplier_Name	Char	Variabile	35	Required	NA	NA	Trademark name of the supplier	Supplier catalog or website
Supplier_Primary_Contact	Char	Fixed	11	Required	NA	NA	Primary phone number of the supplier	Supplier catalog or website

### Supplier\_Specialty Entity

Data Item Name	Data Item Type	Fixed/Variable	Max Data Item Length	Required/Optional	Metadata Min	Metadata Max	Metadata Description	Source
Supplier_Specialty	Char	Variabile	35	Required	NA	NA	Type of materials or products supplied by the supplier	Supplier catalog or website
Supplier_Name	Char	Variabile	35	Required	NA	NA	Trademark name of the supplier	Supplier catalog

								ue or website
--	--	--	--	--	--	--	--	---------------

### Bill Entity

Data Item Name	Data Item Type	Fixed/ Variable	Max Data Item Length	Required/ Optional	Metadata Min	Metadata Max	Metadata Description	Source
Bill_Number	Numeric	Fixed	4	Required	0001	9999	Unique identification number of Bill	Club's billing system database
Bill_Amount	Numeric	Fixed	6	Required	000001	999999	Total amount of Bill	Club's billing system database
Supplier_Name	Char	Variable	35	Required	NA	NA	Trademark name of the supplier	Club's billing system database
Employee_ID	Numeric	Fixed	4	Required	0001	9999	Unique identification number for each employee	HR database

### Product Entity

Data Item Name	Data Item Type	Fixed/ Variable	Max Data Item Length	Required/ Optional	Metadata Min	Metadata Max	Metadata Description	Source
Product_Number	Numeric	Fixed	6	Required	1001	999999	Unique identification number of a Product	Supplier's product catalogue or website
Product_name	Char	Variable	50	Required	NA	NA	Trademark name of the product	Supplier's product catalogue or website
Product_Price	Numeric	Fixed	4	Required	100	5000	Fixed Price of a product	Supplier's product catalogue

								e or website
--	--	--	--	--	--	--	--	--------------

### Supply Schedule Entity

Data Item Name	Data Item Type	Fixed/ Variable	Max Data Item Length	Required/ Optional	Metadata Min	Metadata Max	Metadata Description	Source
Supplier_Name	Char	Variable	35	Required	NA	NA	Trademark name of the supplier	Supplier catalog or website
Supply_Date	Date/Time	Fixed	10	Required	NA	NA	Date in which supplier gives the supply	Supply schedule logs or system
Product_Number	Numeric	Fixed	6	Required	1001	999999	Unique identification number of a Product	Supplier's product catalog or website

### Order Line Entity

Data Item Name	Data Item Type	Fixed/ Variable	Max Data Item Length	Required/ Optional	Metadata Min	Metadata Max	Metadata Description	Source
Order_Number	Numeric	Fixed	6	Required	10001	999999	Unique identification of an order	Club's Order tracking system
Product_Number	Numeric	Fixed	6	Required	1001	999999	Unique identification number of a Product	Supplier's product catalog or website
Order_Quantity	Numeric	Fixed	2	Required	1	99	Number of items in the order	Club's Order tracking system

### Order Entity

Data Item Name	Data Item Type	Fixed/ Variable	Max Data Item Length	Required/ Optional	Metadata Min	Metadata Max	Metadata Description	Source
Order_Number	Numeric	Fixed	6	Required	10001	999999	Unique identification of an order	Club's Order tracking system
Order_Date	Date/Time	Fixed	10	Required	NA	NA	Date in which the order was placed	Club's Order tracking system
Order_Quantity	Numeric	Fixed	2	Required	1	99	Number of items in the order	Club's Order tracking system
Product_Number	Numeric	Fixed	6	Required	1001	999999	Unique identification number of a Product	Supplier's product catalog or website
Maintenance_Employee_ID	Numeric	Fixed	4	Required	1	9999	Unique identification number for each employee	HR database

### Developer\_Profile Entity

Data Item Name	Data Item Type	Fixed/ Variable	Max Data Item Length	Required/ Optional	Metadata Min	Metadata Max	Metadata Description	Source
Developer_Name	Char	Variable	50	Required	NA	NA	Name of the developer company that built the pool	Developer company portal or database
Developer_Primary_Contact	Char	Fixed	10	Required	NA	NA	Primary phone number	Developer company

							of the Developer	y portal or database
--	--	--	--	--	--	--	------------------	----------------------

### Developer\_Contract Entity

Data Item Name	Data Item Type	Fixed/ Variable	Max Data Item Length	Required/ Optional	Metadata Min	Metadata Max	Metadata Description	Source
Developer_Name	Char	Variable	50	Required	NA	NA	Name of the developer company that built the pool	Developer company portal or database
Contract_Number	Numeric	Variable	4	Required	0101	9999	Unique number that determines the contract between club and developer for pools	Club's developer contracts system or database
Warranty_Items	Char	Variable	10	Required	NA	NA	Type of items that the warranty covers by the developer	Club's developer contracts system or database
Contract_Warranty (Years)	Numeric	Variable	2	Required	1	15	Duration of the contract in years	Club's developer contracts system or database
Pool_Number	Numeric	Fixed	3	Required	001	500	Unique number of a pool to be identified	Club's Facility database

## **BUSINESS RULES**

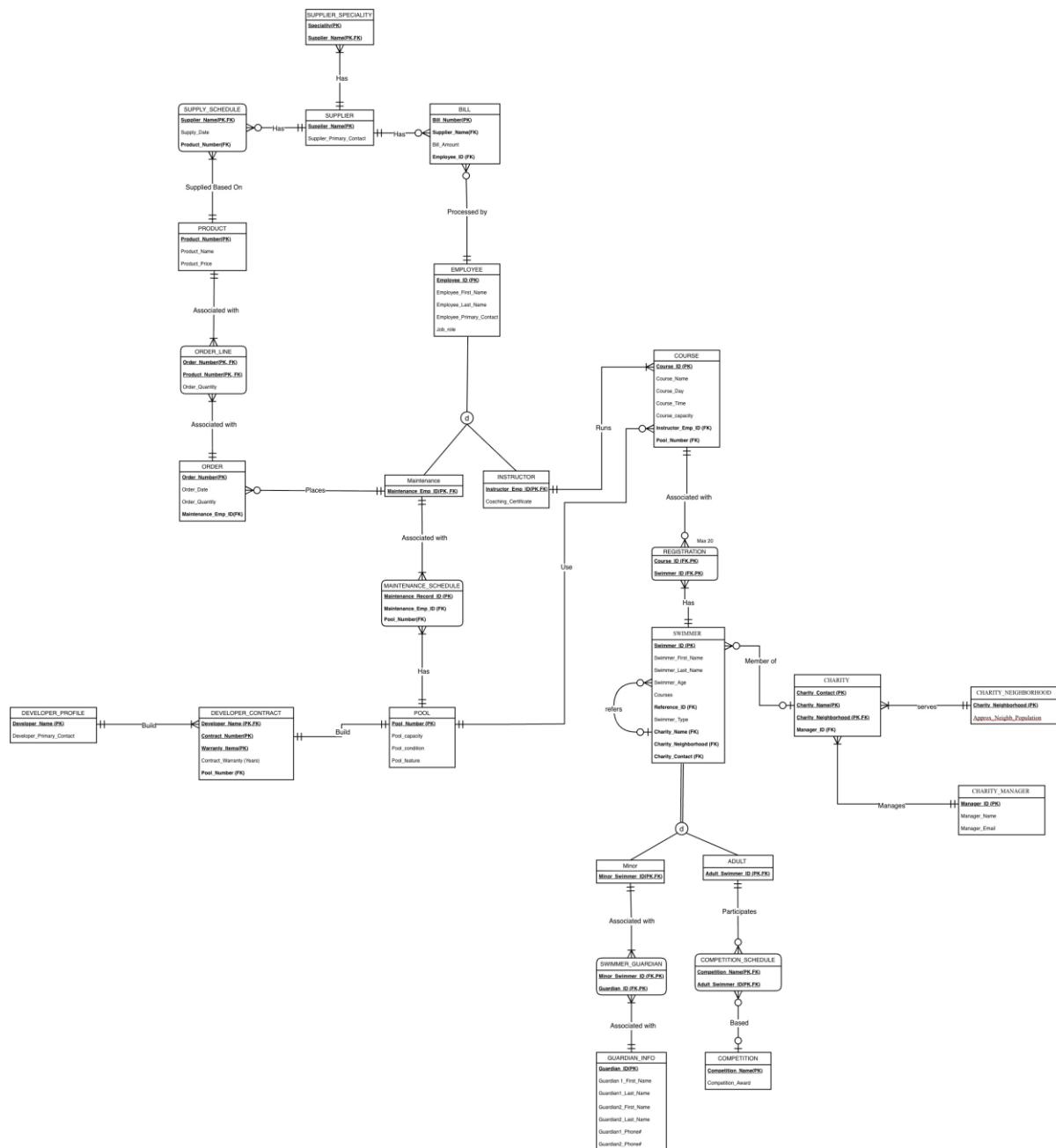
- B01:** Swimmers register to one or more of our swimming classes.
- B02:** New swimmers may be referred by existing members. In this case, both the referrer and the new swimmer will receive a 5% discount.
- B03:** The club may introduce swimmers to certain competitions if they are 18 years of old and above.
- B04:** Swimmers below 18 should be accompanied by previously specified parents/guardians when attending the classes.
- B05:** Each class has its own instructor.
- B06:** Each instructor should not teach more than 20 swimmers in each class.
- B07:** Each class is assigned to one of our ready-to-use swimming pools.
- B08:** Our dedicated maintenance crew handle the cleaning and repairs of each swimming pool.
- B09:** The products needed for cleaning and repairs are specified and directly ordered by the maintenance employees.
- B10:** The requested products are then supplied by either of our supplier partners.
- B11:** Suppliers will send the bill directly to our accountant if they send us any products.
- B12:** The accountant will process the bills accordingly
- B13:** One accountant may manage multiple bills, but each bill is managed by exactly one accountant
- B14:** We give discounts to customers if they are a member of a charity.
- B15:** We give discount only for one membership, so further memberships are not recorded.
- B16:** The discount depends on the approximate population of the neighbourhood that their charity is serving.
- B17:** Each charity has one manager, but one may manage more than a charity.
- B18:** Each charity serves a neighbourhood.
- B19:** There might exist more than one charity in a neighbourhood
- B20:** Each developer company may build multiple pools for us.
- B21:** Each pool has only one developer company.
- B22:** Each pool is built under one particular contract.

## IDENTIFYING PKs and FKs

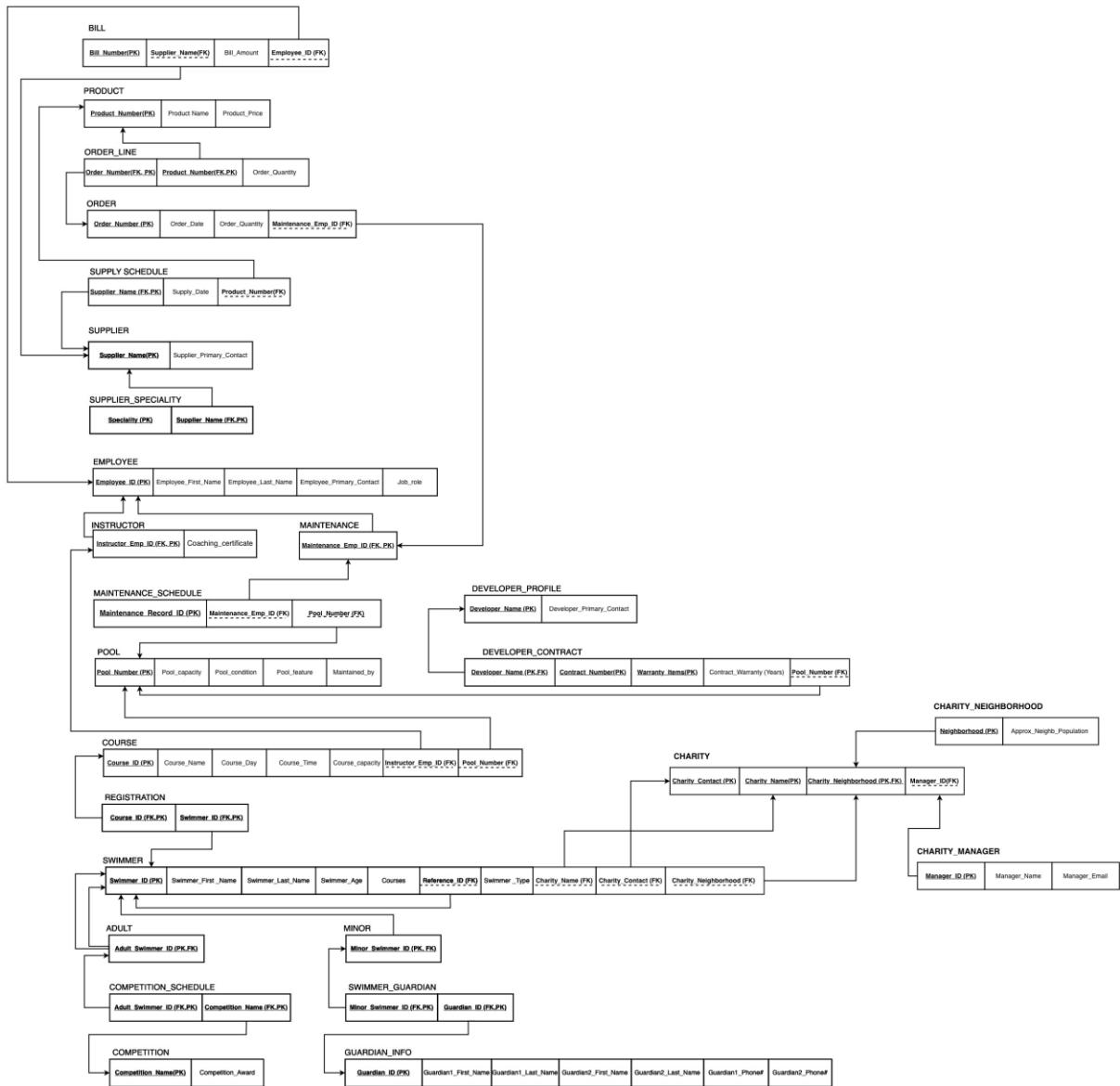
ENTITY	PRIMARY KEY	FOREIGN KEY
Employee	Employee_ID	
Maintenance	Maintenance_Employee_ID	Maintenance_Employee_ID
Instructor	Instructor_Employee_ID	Instructor_Employee_ID
Pool	Pool_Number	
Developer_Profile	Developer_Name	
Developer_Contract	Developer_Name & Contract_Number & Warranty_Items	Developer_Name
Course	Course_ID	Instructor_Employee_ID Pool_Number
Registration	Course_ID & Swimmer_ID	Course_ID & Swimmer_ID
Maintenance_Schedule	Maintenance_Record_ID	Maintenance_Employee_ID , Pool_Number
Swimmer	Swimmer_ID	Reference_ID, Charity_Name, Charity_Contact, Charity_Neighborhood
Minor	Minor_Swimmer_ID	Minor Swimmer_ID
Adult	Adult_Swimmer_ID	Adult_Swimmer_ID
Swimmer_Guardian	Minor_Swimmer_ID & Guardian_ID	Minor_Swimmer_ID & Guardian_ID
Guardian_Info	Guardian_ID	
Competition	Competition_Name	

Competition_Schedule	Competition_Name & Adult_Swimmer_ID	Competition_Name & Adult_Swimmer_ID
Supplier	Supplier_Name	
Supplier_Specialty	Specialty & Supplier_Name	Supplier_Name
Supply_Schedule	Supplier_Name	Supplier_Name & Product_Number
Product	Product_Number	
Order_line	Order_Number & Product_Number	Order_Number & Product_Number
Order	Order_Number	Maintenance_Employee_ID
Bill	Bill_Number	Supplier_Name Employee_ID
Charity	Charity_Name, Charity_Contact, Charity_Neighborhood	Charity_Neighborhood, Manager_ID
Charity_Manager	Manager_ID	
Charity Neighborhood	Neighborhood	

# ENHANCED ENTITY RELATIONSHIP DIAGRAM



# RELATIONAL SCHEMA



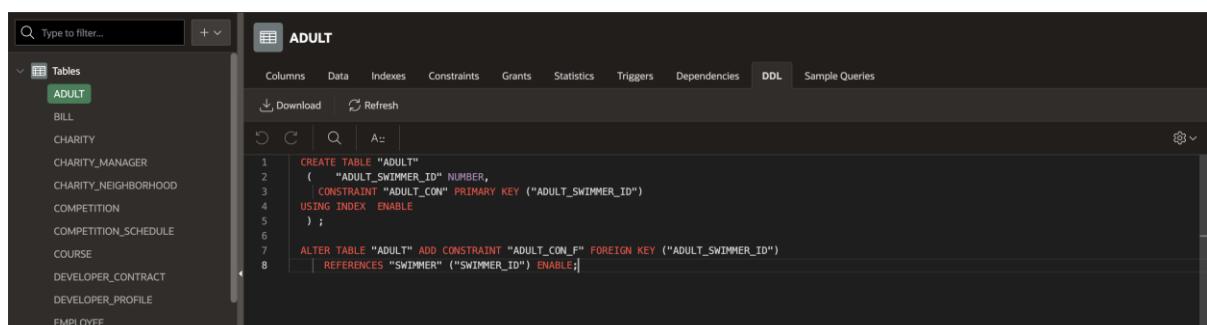
## IDENTIFYING FULL, PARTIAL AND TRANSITIVE DEPENDENCIES

The following tables were only normalized; other tables already satisfy 3NF.

ENTITY	PRIMARY KEY	ATTRIBUTES
<b>Full Dependency</b>		
Developer_Contract	Developer_Name & Contract_Number & Warranty_Items	Contract_Warranty(Years)
Charity	Charity_Name, Charity_Contact, Charity_Neighborhood	Manager_ID
<b>Partial Dependency</b>		
Developer_Profile	Developer_Name	Developer_Primary_Contact
Charity_Neighborhood	Neighborhood	Approx_Neighb_Population
<b>Transitive Dependency</b>		
Charity_Manager	Manager_ID	Manager_Name Manager_Email

## ORACLE SCREENSHOTS:

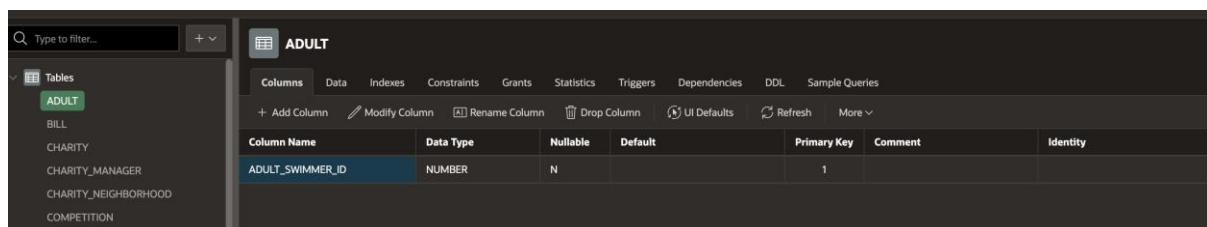
### Table and SQL statements



```

CREATE TABLE "ADULT"
(
    "ADULT_SWIMMER_ID" NUMBER,
    CONSTRAINT "ADULT_CON" PRIMARY KEY ("ADULT_SWIMMER_ID")
    USING INDEX ENABLE
);
ALTER TABLE "ADULT" ADD CONSTRAINT "ADULT_CON_F" FOREIGN KEY ("ADULT_SWIMMER_ID")
    REFERENCES "SWIMMER" ("SWIMMER_ID") ENABLE;

```



Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
ADULT_SWIMMER_ID	NUMBER	N		1		

**BILL**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
BILL_NUMBER	NUMBER	N		1		
BILL_AMOUNT	NUMBER	Y				
SUPPLIER_NAME	VARCHAR2(100 BYTE)	Y				

**BILL**

```

CREATE TABLE "BILL"
(
    "BILL_NUMBER" NUMBER,
    "BILL_AMOUNT" NUMBER,
    "SUPPLIER_NAME" VARCHAR2(100),
    CONSTRAINT "BILL_CON" PRIMARY KEY ("BILL_NUMBER")
    USING INDEX ENABLE
);

```

**CHARITY**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
CHARITY_NAME	VARCHAR2(100 BYTE)	N		1		
CHARITY_NEIGHBORHOOD	VARCHAR2(50 BYTE)	N		2		
MANAGER_ID	NUMBER	N				
CHARITY_CONTACT	NUMBER	N		3		

**CHARITY\_MANAGER**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
MANAGER_ID	NUMBER	N		1		
MANAGER_NAME	VARCHAR2(100 BYTE)	N				
MANAGER_EMAIL	VARCHAR2(255 BYTE)	N				

**CHARITY\_MANAGER**

```

CREATE TABLE "CHARITY_MANAGER"
(
    "MANAGER_ID" NUMBER,
    "MANAGER_NAME" VARCHAR2(100) NOT NULL ENABLE,
    "MANAGER_EMAIL" VARCHAR2(255) NOT NULL ENABLE,
    CONSTRAINT "CHARITY_MANAGER_CON" PRIMARY KEY ("MANAGER_ID")
    USING INDEX ENABLE
);

```

**CHARITY\_NEIGHBORHOOD**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
CHARITY_NEIGHBORHOOD	VARCHAR2(50 BYTE)	N		1		
APPROX_NEIGHB_POPULATION	NUMBER	N				

**CHARITY\_NEIGHBORHOOD**

```

CREATE TABLE "CHARITY_NEIGHBORHOOD"
(
    "CHARITY_NEIGHBORHOOD" VARCHAR2(50),
    "APPROX_NEIGHB_POPULATION" NUMBER NOT NULL ENABLE,
    CONSTRAINT "CHARITY_NEIGHBORHOOD_CON" PRIMARY KEY ("CHARITY_NEIGHBORHOOD")
    USING INDEX ENABLE
);

```

**CHARITY**

```

CREATE TABLE "CHARITY"
(
    "CHARITY_NAME" VARCHAR2(100),
    "CHARITY_NEIGHBORHOOD" VARCHAR2(50),
    "MANAGER_ID" NUMBER NOT NULL ENABLE,
    "CHARITY_CONTACT" NUMBER,
    CONSTRAINT "CHARITY_CON" PRIMARY KEY ("CHARITY_NAME", "CHARITY_NEIGHBORHOOD", "CHARITY_CONTACT")
    USING INDEX ENABLE
);

ALTER TABLE "CHARITY" ADD CONSTRAINT "CHARITY_CON_F" FOREIGN KEY ("CHARITY_NEIGHBORHOOD")
    REFERENCES "CHARITY_NEIGHBORHOOD" ("CHARITY_NEIGHBORHOOD") ENABLE;

```

**COMPETITION**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
COMPETITION_NAME	VARCHAR2(100 BYTE)	N		1		
COMPETITION_AWARD	VARCHAR2(50 BYTE)	Y				

**COMPETITION**

```

CREATE TABLE "COMPETITION"
(
    "COMPETITION_NAME" VARCHAR2(100),
    "COMPETITION_AWARD" VARCHAR2(50),
    CONSTRAINT "COMPETITION_CON" PRIMARY KEY ("COMPETITION_NAME")
    USING INDEX ENABLE
);

```

**COMPETITION\_SCHEDULE**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
ADULT_SWIMMER_ID	NUMBER	N		1		
COMPETITION_NAME	VARCHAR2(100 BYTE)	N		2		

**COMPETITION\_SCHEDULE**

```

1 CREATE TABLE "COMPETITION_SCHEDULE"
2   (
3     "ADULT_SWIMMER_ID" NUMBER,
4     "COMPETITION_NAME" VARCHAR2(100),
5     CONSTRAINT "COMPETITION_SCHEDULE_CON" PRIMARY KEY ("ADULT_SWIMMER_ID", "COMPETITION_NAME")
6     USING INDEX ENABLE
7
8     ALTER TABLE "COMPETITION_SCHEDULE" ADD CONSTRAINT "COMPETITION_SCHEDULE_CON_F" FOREIGN KEY ("ADULT_SWIMMER_ID")
9       REFERENCES "ADULT" ("ADULT_SWIMMER_ID") ENABLE;
10    ALTER TABLE "COMPETITION_SCHEDULE" ADD CONSTRAINT "COMPETITION_SCHEDULE_CON_FF" FOREIGN KEY ("COMPETITION_NAME")
11      REFERENCES "COMPETITION" ("COMPETITION_NAME") ENABLE;

```

**COURSE**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
COURSE_ID	NUMBER	N		1		
COURSE_NAME	VARCHAR2(100 BYTE)	Y				
INSTRUCTOR_EMPLOYEE_ID	NUMBER	Y				
COURSE_DAY	VARCHAR2(50 BYTE)	Y				
COURSE_TIME	VARCHAR2(50 BYTE)	Y				
COURSE_CAPACITY	VARCHAR2(50 BYTE)	Y				
POOL_NUMBER	VARCHAR2(50 BYTE)	Y				

**COURSE**

```

1 CREATE TABLE "COURSE"
2   (
3     "COURSE_ID" NUMBER,
4     "COURSE_NAME" VARCHAR2(100),
5     "INSTRUCTOR_EMPLOYEE_ID" NUMBER,
6     "COURSE_DAY" VARCHAR2(50),
7     "COURSE_TIME" VARCHAR2(50),
8     "COURSE_CAPACITY" VARCHAR2(50),
9     "POOL_NUMBER" VARCHAR2(50),
10    CONSTRAINT "COURSE_CON" PRIMARY KEY ("COURSE_ID")
11    USING INDEX ENABLE
12
13    ALTER TABLE "COURSE" ADD CONSTRAINT "COURSE_CON_F" FOREIGN KEY ("POOL_NUMBER")
14      REFERENCES "POOL" ("POOL_NUMBER") ENABLE;
15    ALTER TABLE "COURSE" ADD CONSTRAINT "COURSE_CON_FF" FOREIGN KEY ("INSTRUCTOR_EMPLOYEE_ID")
16      REFERENCES "INSTRUCTOR" ("INSTRUCTOR_EMPLOYEE_ID") ENABLE;

```

**DEVELOPER\_CONTRACT**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
DEVELOPER_NAME	VARCHAR2(100 BYTE)	N		1		
CONTRACT_NUMBER	VARCHAR2(50 BYTE)	N		2		
CONTRACT_WARRANTY_YEARS	VARCHAR2(50 BYTE)	Y				
WARRANTY_ITEMS	VARCHAR2(50 BYTE)	N		3		
POOL_NUMBER	VARCHAR2(50 BYTE)	Y				

**DEVELOPER\_CONTRACT**

```

CREATE TABLE "DEVELOPER_CONTRACT"
(
    "DEVELOPER_NAME" VARCHAR2(100),
    "CONTRACT_NUMBER" VARCHAR2(50),
    "CONTRACT_WARRANTY_YEARS" VARCHAR2(50),
    "WARRANTY_ITEMS" VARCHAR2(50),
    "POOL_NUMBER" VARCHAR2(50),
    CONSTRAINT "DEVELOPER_CONTRACT_CON" PRIMARY KEY ("DEVELOPER_NAME", "CONTRACT_NUMBER", "WARRANTY_ITEMS")
    USING INDEX ENABLE
);

ALTER TABLE "DEVELOPER_CONTRACT" ADD CONSTRAINT "DEVELOPER_CONTRACT_CON_F" FOREIGN KEY ("DEVELOPER_NAME")
    REFERENCES "DEVELOPER_PROFILE" ("DEVELOPER_NAME") ENABLE;
ALTER TABLE "DEVELOPER_CONTRACT" ADD CONSTRAINT "DEVELOPER_CONTRACT_CON_FF" FOREIGN KEY ("POOL_NUMBER")
    REFERENCES "POOL" ("POOL_NUMBER") ENABLE;

```

**DEVELOPER\_PROFILE**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
DEVELOPER_NAME	VARCHAR2(100 BYTE)	N		1		
DEVELOPER_PRIMARY_CONTACT	NUMBER	Y				

**DEVELOPER\_PROFILE**

```

CREATE TABLE "DEVELOPER_PROFILE"
(
    "DEVELOPER_NAME" VARCHAR2(100),
    "DEVELOPER_PRIMARY_CONTACT" NUMBER,
    CONSTRAINT "DEVELOPER_PROFILE_CON" PRIMARY KEY ("DEVELOPER_NAME")
    USING INDEX ENABLE
);

```

**EMPLOYEE**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
EMPLOYEE_ID	NUMBER	N		1		
EMPLOYEE_FIRST_NAME	VARCHAR2(100 BYTE)	N				
EMPLOYEE_LAST_NAME	VARCHAR2(100 BYTE)	N				
EMPLOYEE_PRIMARY_CONTACT	NUMBER	N				
COACHING_CERTIFICATE	VARCHAR2(50 BYTE)	Y				
JOB_ROLE	VARCHAR2(50 BYTE)	N				

**EMPLOYEE**

```

CREATE TABLE "EMPLOYEE"
(
    "EMPLOYEE_ID" NUMBER NOT NULL ENABLE,
    "EMPLOYEE_FIRST_NAME" VARCHAR2(100) NOT NULL ENABLE,
    "EMPLOYEE_LAST_NAME" VARCHAR2(100) NOT NULL ENABLE,
    "EMPLOYEE_PRIMARY_CONTACT" NUMBER NOT NULL ENABLE,
    "COACHING_CERTIFICATE" VARCHAR2(50),
    "JOB_ROLE" VARCHAR2(50) NOT NULL ENABLE,
    CONSTRAINT "EMPLOYEE_CON" PRIMARY KEY ("EMPLOYEE_ID")
    USING INDEX ENABLE
);

```

**GUARDIAN\_INFO**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
GUARDIAN_ID	NUMBER	N	"WKSP_DHARSHINIUB"."ISEQ\$\$_1"	1		BY DEFAULT
GUARDIAN1_FIRST_NAME	VARCHAR2(100 BYTE)	Y				
GUARDIAN2_LAST_NAME	VARCHAR2(100 BYTE)	Y				
GUARDIAN2_FIRST_NAME	VARCHAR2(100 BYTE)	Y				
GUARDIAN2_LAST_NAME2	VARCHAR2(50 BYTE)	Y				
GUARDIAN1_PHONENUMBER	VARCHAR2(50 BYTE)	Y				
GUARDIAN2_PHONENUMBER	VARCHAR2(50 BYTE)	Y				

**GUARDIAN\_INFO**

```

CREATE TABLE "GUARDIAN_INFO"
(
    "GUARDIAN_ID" NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY MINVALUE 1 MAXVALUE 99999999999999999999999999999999 INCREMENT BY 1 START WITH 1 CACHE 20
    "GUARDIAN1_FIRST_NAME" VARCHAR2(100),
    "GUARDIAN1_LAST_NAME" VARCHAR2(100),
    "GUARDIAN2_FIRST_NAME" VARCHAR2(100),
    "GUARDIAN2_LAST_NAME2" VARCHAR2(50),
    "GUARDIAN1_PHONENUMBER" VARCHAR2(50),
    "GUARDIAN2_PHONENUMBER" VARCHAR2(50),
    PRIMARY KEY ("GUARDIAN_ID")
)
USING INDEX ENABLE

```

**INSTRUCTOR**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
INSTRUCTOR_EMPLOYEE_ID	NUMBER	N		1		
COACHING_CERTIFICATE	VARCHAR2(50 BYTE)	Y				

**INSTRUCTOR**

```

CREATE TABLE "INSTRUCTOR"
(
    "INSTRUCTOR_EMPLOYEE_ID" NUMBER,
    "COACHING_CERTIFICATE" VARCHAR2(50),
    CONSTRAINT "INSTRUCTOR_CON" PRIMARY KEY ("INSTRUCTOR_EMPLOYEE_ID")
)
USING INDEX ENABLE

```

**MAINTENANCE**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
MAINTENANCE_EMPLOYEE_ID	NUMBER	N		1		

Type to filter... +

### Maintenance

Columns Data Indexes Constraints Grants Statistics Triggers Dependencies DDL Sample Queries

Download Refresh

```

1 | CREATE TABLE "MAINTENANCE"
2 | (
3 |   "MAINTENANCE_EMPLOYEE_ID" NUMBER,
4 |   CONSTRAINT "MAINTENANCE_CON" PRIMARY KEY ("MAINTENANCE_EMPLOYEE_ID")
5 |   USING INDEX ENABLE
6 |
7 |   ALTER TABLE "MAINTENANCE" ADD CONSTRAINT "MAINTENANCE_CON_F" FOREIGN KEY ("MAINTENANCE_EMPLOYEE_ID")
8 |     REFERENCES "EMPLOYEE" ("EMPLOYEE_ID") ENABLE;

```

Type to filter... +

### Maintenance\_Schedule

Columns Data Indexes Constraints Grants Statistics Triggers Dependencies DDL Sample Queries

Add Column Modify Column Rename Column Drop Column UI Defaults Refresh More

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
MAINTENANCE_RECORD_ID	NUMBER	N	"WKSP_DHARSHINIUB"."ISEQ\$...	1		BY DEFAULT
POOL_NUMBER	NUMBER	Y				
MAINTENANCE_EMPLOYEE_ID	NUMBER	Y				

Type to filter... +

### Maintenance\_Schedule

Columns Data Indexes Constraints Grants Statistics Triggers Dependencies DDL Sample Queries

Download Refresh

```

1 | CREATE TABLE "MAINTENANCE_SCHEDULE"
2 | (
3 |   "MAINTENANCE_RECORD_ID" NUMBER GENERATED BY DEFAULT ON NULL AS IDENTITY MINVALUE 1 MAXVALUE 99999999999999999999999999999999 INCREMENT BY 1 START WITH 1
4 |   "POOL_NUMBER" NUMBER,
5 |   "MAINTENANCE_EMPLOYEE_ID" NUMBER,
6 |   CONSTRAINT "MAINTENANCE_SCHEDULE_CON" PRIMARY KEY ("MAINTENANCE_RECORD_ID")
7 |   USING INDEX ENABLE
8 |
9 |   ALTER TABLE "MAINTENANCE_SCHEDULE" ADD CONSTRAINT "MAINTENANCE_SCHEDULE_CON" FOREIGN KEY ("MAINTENANCE_EMPLOYEE_ID")
10 |    REFERENCES "MAINTENANCE" ("MAINTENANCE_EMPLOYEE_ID") ENABLE;

```

Type to filter... +

### Minor

Columns Data Indexes Constraints Grants Statistics Triggers Dependencies DDL Sample Queries

Add Column Modify Column Rename Column Drop Column UI Defaults Refresh More

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
MINOR_SWIMMER_ID	NUMBER	N		1		

Type to filter... +

### Minor

Columns Data Indexes Constraints Grants Statistics Triggers Dependencies DDL Sample Queries

Download Refresh

```

1 | CREATE TABLE "MINOR"
2 | (
3 |   "MINOR_SWIMMER_ID" NUMBER,
4 |   CONSTRAINT "MINOR_CON" PRIMARY KEY ("MINOR_SWIMMER_ID")
5 |   USING INDEX ENABLE
6 |
7 |   ALTER TABLE "MINOR" ADD CONSTRAINT "MINOR_CON_F" FOREIGN KEY ("MINOR_SWIMMER_ID")
8 |     REFERENCES "SWIMMER" ("SWIMMER_ID") ENABLE;

```

**ORDER**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
ORDER_NUMBER	NUMBER	N		1		
ORDER_DATE	DATE	Y				
ORDER_QUANTITY	NUMBER	Y				
MAINTENANCE_EMPLOYEE_ID	NUMBER	Y				

```

CREATE TABLE "ORDER"
(
    "ORDER_NUMBER" NUMBER(*,0),
    "ORDER_DATE" DATE,
    "ORDER_QUANTITY" NUMBER(*,0),
    "MAINTENANCE_EMPLOYEE_ID" NUMBER(*,0),
    CONSTRAINT "ORDER_CON" PRIMARY KEY ("ORDER_NUMBER")
    USING INDEX ENABLE
);
ALTER TABLE "ORDER" ADD CONSTRAINT "ORDER_CON_F" FOREIGN KEY ("MAINTENANCE_EMPLOYEE_ID")
    REFERENCES "MAINTENANCE" ("MAINTENANCE_EMPLOYEE_ID") ENABLE;

```

**ORDER\_LINE**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
ORDER_NUMBER	NUMBER	N		1		
PRODUCT_NUMBER	NUMBER	N		2		
ORDER_QUANTITY	NUMBER	Y				

```

CREATE TABLE "ORDER_LINE"
(
    "ORDER_NUMBER" NUMBER,
    "PRODUCT_NUMBER" NUMBER,
    "ORDER_QUANTITY" NUMBER,
    CONSTRAINT "ORDER_LINE_CON" PRIMARY KEY ("ORDER_NUMBER", "PRODUCT_NUMBER")
    USING INDEX ENABLE
);
ALTER TABLE "ORDER_LINE" ADD CONSTRAINT "ORDER_LINE_CON_F" FOREIGN KEY ("PRODUCT_NUMBER")
    REFERENCES "PRODUCT" ("PRODUCT_NUMBER") ENABLE;

```

**POOL**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
POOL_CAPACITY	VARCHAR2(50 BYTE)	Y				
POOL_CONDITION	VARCHAR2(50 BYTE)	Y				
POOL_FEATURE	VARCHAR2(50 BYTE)	Y				
POOL_NUMBER	VARCHAR2(50 BYTE)	N		1		

**POOL**

```

CREATE TABLE "POOL"
(
    "POOL_CAPACITY" VARCHAR2(50),
    "POOL_CONDITION" VARCHAR2(50),
    "POOL_FEATURE" VARCHAR2(50),
    "POOL_NUMBER" VARCHAR2(50),
    CONSTRAINT "POOL_CON" PRIMARY KEY ("POOL_NUMBER")
    USING INDEX ENABLE
);

```

**PRODUCT**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
PRODUCT_NUMBER	NUMBER	N		1		
PRODUCT_NAME	VARCHAR2(100 BYTE)	Y				
PRODUCT_PRICE	NUMBER	Y				

**PRODUCT**

```

CREATE TABLE "PRODUCT"
(
    "PRODUCT_NUMBER" NUMBER,
    "PRODUCT_NAME" VARCHAR2(100),
    "PRODUCT_PRICE" NUMBER,
    CONSTRAINT "PRODUCT_CON" PRIMARY KEY ("PRODUCT_NUMBER")
    USING INDEX ENABLE
);

```

**REGISTRATION**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
COURSE_ID	NUMBER	N		1		
SWIMMER_ID	NUMBER	N		2		

**REGISTRATION**

```

CREATE TABLE "REGISTRATION"
(
    "COURSE_ID" NUMBER(*,0),
    "SWIMMER_ID" NUMBER(*,0),
    PRIMARY KEY ("COURSE_ID", "SWIMMER_ID")
    USING INDEX ENABLE
);

ALTER TABLE "REGISTRATION" ADD FOREIGN KEY ("COURSE_ID")
    REFERENCES "COURSE" ("COURSE_ID") ENABLE;
ALTER TABLE "REGISTRATION" ADD FOREIGN KEY ("SWIMMER_ID")
    REFERENCES "SWIMMER" ("SWIMMER_ID") ENABLE;

```

**SUPPLIER**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
SUPPLIER_NAME	VARCHAR2(100 BYTE)	N		1		
SUPPLIER_PRIMARY_CONTACT	NUMBER	Y				

**SUPPLIER**

```

1 CREATE TABLE "SUPPLIER"
2 (
3     "SUPPLIER_NAME" VARCHAR2(100),
4     "SUPPLIER_PRIMARY_CONTACT" NUMBER,
5     | CONSTRAINT "SUPPLIER_CON" PRIMARY KEY ("SUPPLIER_NAME")
6     USING INDEX ENABLE
7 );

```

**SUPPLIER\_SPECIALITY**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
SUPPLIER_NAME	VARCHAR2(100 BYTE)	N		1		
SUPPLIER_SPECIALTY	VARCHAR2(50 BYTE)	N		2		

**SUPPLIER\_SPECIALITY**

```

1 CREATE TABLE "SUPPLIER_SPECIALITY"
2 (
3     "SUPPLIER_NAME" VARCHAR2(100),
4     "SUPPLIER_SPECIALTY" VARCHAR2(50),
5     | CONSTRAINT "SUPPLIER_SPECIALITY_CON" PRIMARY KEY ("SUPPLIER_NAME", "SUPPLIER_SPECIALTY")
6     USING INDEX ENABLE
7 );
8
9 ALTER TABLE "SUPPLIER_SPECIALITY" ADD CONSTRAINT "SUPPLIER_SPECIALITY_CON_F" FOREIGN KEY ("SUPPLIER_NAME")
10 REFERENCES "SUPPLIER" ("SUPPLIER_NAME") ENABLE;

```

**SUPPLY\_SCHEDULE**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
SUPPLIER_NAME	VARCHAR2(100 BYTE)	N		1		
SUPPLY_DATE	DATE	N		2		
PRODUCT_NUMBER	NUMBER	N		3		

**SUPPLY\_SCHEDULE**

```

1 CREATE TABLE "SUPPLY_SCHEDULE"
2 (
3     "SUPPLIER_NAME" VARCHAR2(100),
4     "SUPPLY_DATE" DATE,
5     "PRODUCT_NUMBER" NUMBER,
6     | CONSTRAINT "SUPPLY_SCHEDULE_CON" PRIMARY KEY ("SUPPLIER_NAME", "SUPPLY_DATE", "PRODUCT_NUMBER")
7     USING INDEX ENABLE
8 );
9
10 ALTER TABLE "SUPPLY_SCHEDULE" ADD CONSTRAINT "SUPPLY_SCHEDULE_CON_F" FOREIGN KEY ("PRODUCT_NUMBER")
11 REFERENCES "PRODUCT" ("PRODUCT_NUMBER") ENABLE;

```

**SWIMMER**

Column Name	Data Type	Nullable	Default	Primary Key	Comment	Identity
SWIMMER_ID	NUMBER	N		1		
SWIMMER_FIRST_NAME	VARCHAR2(100 BYTE)	Y				
SWIMMER_LAST_NAME	VARCHAR2(100 BYTE)	Y				
SWIMMER_AGE	VARCHAR2(50 BYTE)	Y				
REFERENCE_ID	VARCHAR2(50 BYTE)	Y				
SWIMMER_TYPE	VARCHAR2(1 BYTE)	Y				
CHARITY_NAME	VARCHAR2(100 BYTE)	Y				

The screenshots show the creation of three tables:

- SWIMMER** (Table 1):

```

1 CREATE TABLE "SWIMMER"
2 (
3     "SWIMMER_ID" NUMBER,
4     "SWIMMER_FIRST_NAME" VARCHAR2(100),
5     "SWIMMER_LAST_NAME" VARCHAR2(100),
6     "SWIMMER_AGE" VARCHAR2(50),
7     "REFERENCE_ID" VARCHAR2(50),
8     "SWIMMER_TYPE" VARCHAR2(1),
9     "CHARITY_NAME" VARCHAR2(100),
10    CONSTRAINT "SWIMMER_CON" PRIMARY KEY ("SWIMMER_ID")
11    USING INDEX ENABLE
);

```
- SWIMMER\_GUARDIAN** (Table 2):

```

Columns Data Indexes Constraints Grants Statistics Triggers Dependencies DDL Sample Queries
+ Add Column Modify Column Rename Column Drop Column UI Defaults Refresh More
Column Name Data Type Nullable Default Primary Key Comment Identity
MINOR_SWIMMER_ID NUMBER N 1
GUARDIAN_ID NUMBER N 2

```
- SWIMMER\_GUARDIAN** (Table 3):

```

1 CREATE TABLE "SWIMMER_GUARDIAN"
2 (
3     "MINOR_SWIMMER_ID" NUMBER,
4     "GUARDIAN_ID" NUMBER,
5     CONSTRAINT "SWIMMER_GUARDIAN_CON" PRIMARY KEY ("MINOR_SWIMMER_ID", "GUARDIAN_ID")
6     USING INDEX ENABLE
7
8     ALTER TABLE "SWIMMER_GUARDIAN" ADD CONSTRAINT "SWIMMER_GUARDIAN_CON_F" FOREIGN KEY ("GUARDIAN_ID")
9     REFERENCES "GUARDIAN_INFO" ("GUARDIAN_ID") ENABLE;
ALTER TABLE "SWIMMER_GUARDIAN" ADD CONSTRAINT "SWIMMER_GUARDIAN_CON_F2" FOREIGN KEY ("MINOR_SWIMMER_ID")
10    REFERENCES "MINOR" ("MINOR_SWIMMER_ID") ENABLE;
11

```

## Section 2: Query 1

Supplier who has supplied the most varieties of products

SQL Query:

```

Language SQL Rows 10 Clear Command Find Tables Save Run
1 SELECT ss.Supplier_Name,
2        COUNT(DISTINCT ss.Product_Number) AS Variety_Count
3 FROM SUPPLY_SCHEDULE ss
4 GROUP BY ss.Supplier_Name
5 ORDER BY Variety_Count DESC
6 FETCH FIRST 1 ROWS ONLY;

```

Results:

SUPPLIER_NAME	VARIETY_COUNT
Fantastic Everything Co.	4

1 rows returned in 0.00 seconds Download

## Section 2: Query 2

Competition that interests swimmers the most

SQL Query:

```

Language SQL Rows 10 Clear Command Find Tables Save Run
1 SELECT cs.Competition_Name,
2        COUNT(cs.Adult_Swimmer_ID) AS Participants
3 FROM COMPETITION_SCHEDULE cs
4 GROUP BY cs.Competition_Name
5 ORDER BY Participants DESC
6 FETCH FIRST 1 ROWS ONLY;

```

Results:

COMPETITION_NAME	PARTICIPANTS
Regional 1	2

1 rows returned in 0.01 seconds Download

## Section 2: Query 3

Warranty conditions for the pool hosting the max classes

Language SQL Rows 10 Clear Command Find Tables Save Run

```
1 WITH PoolCounts AS (
2     SELECT Pool_Number,
3            COUNT(*) AS Class_Count
4     FROM COURSE
5     GROUP BY Pool_Number
6 ),
7 MaxPool AS (
8     SELECT Pool_Number
9     FROM PoolCounts
10    ORDER BY Class_Count DESC
11    FETCH FIRST 1 ROWS ONLY
12 )
13 SELECT p.Pool_Number,
14        dc.Warranty_Items,
15        dc.Contract_Warranty_Years
16 FROM MaxPool mp
17 JOIN POOL p ON p.Pool_Number = mp.Pool_Number
18 JOIN DEVELOPER_CONTRACT dc ON dc.Pool_Number = p.Pool_Number;
```

Results Explain Describe Saved SQL History

POOL_NUMBER	WARRANTY_ITEMS	CONTRACT_WARRANTY_YEARS
1	Structure	2

1 rows returned in 0.03 seconds [Download](#)

### **Section 3: Advanced Query**

## Top 5 courses with the highest number of enrolled swimmers

```
Language SQL Rows 10 Clear Command Find Tables Save Run

WITH CourseEnrollment AS (
    SELECT Course_ID,
           COUNT(Swimmer_ID) AS Enrolled_Count
    FROM REGISTRATION
   GROUP BY Course_ID
)
SELECT c.Course_ID,
       c.Course_Name,
       COALESCE(ce.Enrolled_Count, 0) AS Enrolled_Count
  FROM COURSE c
 LEFT JOIN CourseEnrollment ce ON c.Course_ID = ce.Course_ID
 ORDER BY Enrolled_Count DESC
FETCH FIRST 5 ROWS ONLY;
14
```

## INTEGRITY CONSTRAINTS

## Domain Integrity

Domain integrity ensures that **every attribute value** falls within its valid and defined domain (e.g., data type, format, allowed range). We reviewed all column definitions in the updated ER diagram and confirmed that attributes follow their expected domains. Examples include:

- **Product\_Price, Bill\_Amount, Contract\_Warranty(years)** → positive numeric values
  - **Supply\_Date, Order\_Date, Maintenance\_Date, Competition\_Date** → valid date formats
  - **Phone, Primary\_Contact, Age** → adhere to expected ranges and formats
  - **Competition\_Award, Job\_Title, Supplier\_SpecialityName** → valid string domains

All attributes in the database adhere to their domain definitions.

## **Referential Integrity**

Rules that maintain consistency between the rows of two related tables, that is a foreign key in one table, must have a reference of a primary key in another table. We've checked foreign key relationships for all entities to ensure every referenced ID exists in the parent table. For example,

- BILL.Employee\_ID → EMPLOYEE.Employee\_ID (PK)
- SUPPLY\_SCHEDULE.Supplier\_Name → SUPPLIER.Supplier\_Name (PK)
- ORDER\_LINE.Product\_Number → PRODUCT.Product\_Number (PK)
- ORDER.Maintenance\_Employee\_ID → EMPLOYEE.Employee\_ID (PK)
- MAINTENANCE\_SCHEDULE.Pool\_Number → POOL.Pool\_Number (PK)
- DEVELOPER\_CONTRACT.Developer\_Name → DEVELOPER\_PROFILE.Developer\_Name (PK)
- POOL.Developer\_Name → DEVELOPER\_PROFILE.Developer\_Name (PK)
- REGISTRATION.Swimmer\_ID → SWIMMER.Swimmer\_ID (PK)
- COURSE.Instructor\_ID → INSTRUCTOR.Instructor\_Employee\_ID (PK)
- COMPETITION\_SCHEDULE.Adult\_Swimmer\_ID / Minor\_Swimmer\_ID → ADULT / MINOR (PK)
- SWIMMER\_GUARDIAN.Guardian\_ID → GUARDIAN\_INFO.Guardian\_ID (PK)

As per our analysis, there were no unmatched foreign keys, concluding that referential integrity is maintained.

## **Entity Integrity**

Every table must have a primary key, and no part of that key can be NULL, duplicate or missing. We've verified that each table had a primary key defined (e.g., EmployeeID, OrderID, etc.) and that no rows contained NULL values, no duplicates or no missing values for that key.

All primary keys are unique and not-null values. Hence, the entity integrity constraint has been satisfied.

## **CONCLUSION**

In conclusion, our project successfully integrates the new entities, updates all prior tasks, and delivers a fully normalized and SQL-implemented database in accordance with the Step 3 rubric. Through identifying functional dependencies and addressing anomalies, all required tables were brought into Third Normal Form, and new tables were created where necessary. The updated ERD and relational schema now accurately reflect the club's operational structure, including the newly added Developer and Charity components and their associated business rules.

The final implementation in Oracle Apex demonstrates correct enforcement of data types, primary keys, foreign keys, and all integrity constraints. The project is logically sound and ready for future system development.