A red and white flag

Description automatically generated with medium confidence

Onondaga Yacht Club

Membership Database

Table of Contents

[Part One 3](#_Toc82620563)

[Summary 3](#_Toc82620564)

[Stakeholders 3](#_Toc82620565)

[Business Rules 3](#_Toc82620566)

[Data Questions 3](#_Toc82620567)

[Conceptual Model 4](#_Toc82620568)

[Logical Model 4](#_Toc82620569)

[Part 2 5](#_Toc82620570)

[Data Definition Language – Creating Tables and Constraints 5](#_Toc82620571)

[Data Manipulation Language 6](#_Toc82620572)

[Adding Data using INSERT Statements 6](#_Toc82620573)

[Querying Data Using SELECT Statements 6](#_Toc82620574)

[Programming Objects 7](#_Toc82620575)

[User Interface 7](#_Toc82620576)

[Reflection 9](#_Toc82620577)

# Part One

## Summary

The Onondaga Yacht Club (OYC) was founded in 1883 and since that time they have tracked member information either on paper or on spreadsheets. To ease the level of effort and increase security and integrity of maintaining membership info, the OYC Officers would like to build a database to house the membership info.

Note that while OYC exists, all organizational particulars, including business rules and club member information mentioned in this document are done so for academic purposes expressly as an example of the final project deliverable.

## Stakeholders

* Existing OYC Members
  + People who are currently active members of the club
* New OYC Members
  + People who are applying for membership to the club
* OYC Officers
  + A selection of existing members who perform club-related administrative functions

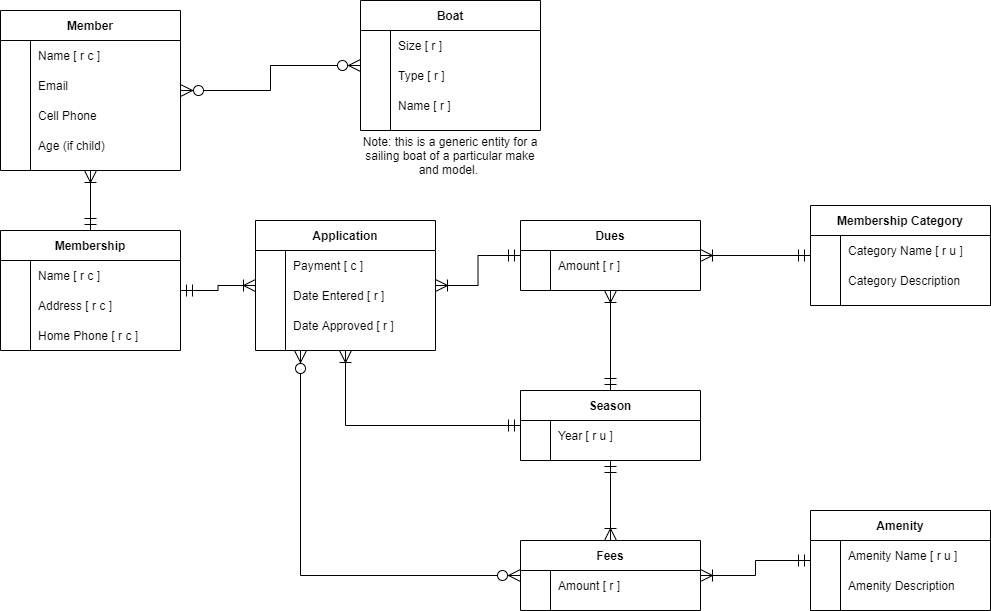
## Business Rules

* Each membership has a membership category: Single, Single Family, Family, or Associate. Any married members must join as family. Single family is one adult plus children. Associate members include extraorganizational affiliations as well as students.
* A membership category has a dues and capital amount, which may change each year at the discretion of the OYC Officers.
* As part of a membership, an applicant can elect to reserve certain club amenities. These also have required dues at the discretion of the officers.
* Each member can have zero or more boats to list with the club. It is not required to own a boat to be a member.
* Other business rules will be defined in later iterations of the database.

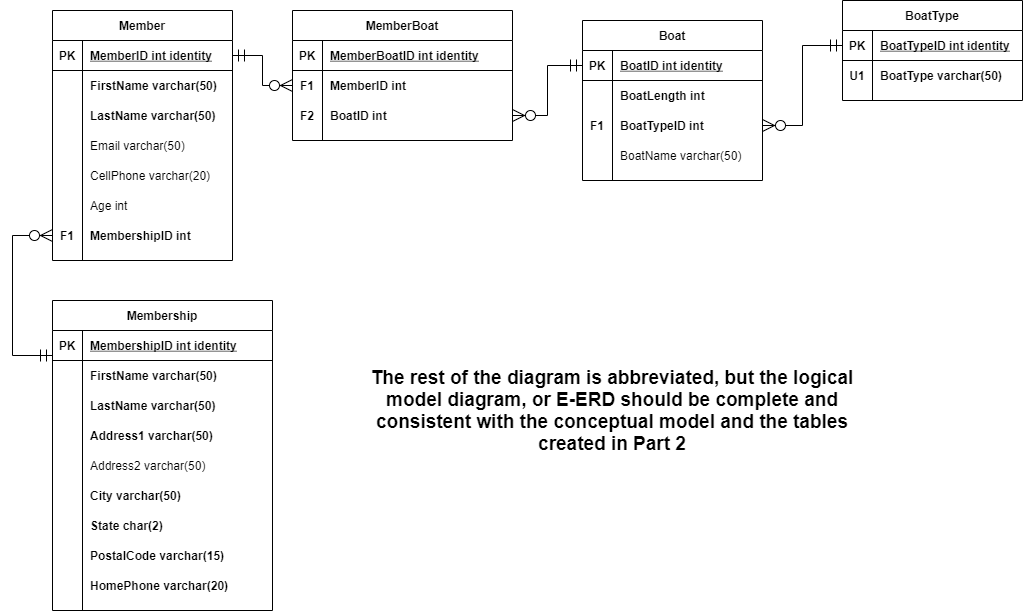
## Data Questions

* Who are our members and what is their contact information?
* How much was collected in dues for this membership year?
* Which members are in the “Hunter” club? This is a list of members who have applied and paid the dues for use of the Hunter 240
* Are there members who did not renew their membership and, of so, who are they?

## Conceptual Model



## Logical Model



# Part 2

## Data Definition Language – Creating Tables and Constraints

-- DDL

/\*

Abbreviated to only include tables shown in the logical model diagram. It's okay

if your database doesn't include all tables, so long as you have a workable database

that answers 3-5 data questions

\*/

-- Cleaning up the database for initialization by conditionally dropping tables

DROP TABLE IF EXISTS MemberBoat

DROP TABLE IF EXISTS Member

DROP TABLE IF EXISTS Membership

DROP TABLE IF EXISTS Boat

DROP TABLE IF EXISTS BoatType

-- Creating the tables and constraints

CREATE TABLE BoatType (

BoatTypeID int identity primary key

, BoatType varchar(50) not null unique

)

CREATE TABLE Boat (

BoatID int identity primary key

, BoatLength int not null

, BoatTypeID int not null foreign key references BoatType(BoatTypeID)

, BoatName varchar(50)

)

CREATE TABLE Membership (

MembershipID int identity primary key

, FirstName varchar(50) not null

, LastName varchar(50) not null

, Address1 varchar(50) not null

, Address2 varchar(50)

, City varchar(50) not null

, StateAbbrev char(2) not null default 'NY' -- Changed name to avoid conflict.

, PostalCode varchar(15) not null

, HomePhone varchar(20)

)

CREATE TABLE Member (

MemberID int identity primary key

, FirstName varchar(50) not null

, LastName varchar(50) not null

, Email varchar(50)

, CellPhone varchar(20)

, Age int -- Not required for adults

, MembershipID int not null foreign key references Membership(MembershipID)

)

CREATE TABLE MemberBoat (

MemberBoatID int identity primary key

, MemberID int not null foreign key references Member(MemberID)

, BoatID int not null foreign key references Boat(BoatID)

)

## Data Manipulation Language

### Adding Data using INSERT Statements

INSERT INTO BoatType (BoatType) VALUES

('Sailboat')

, ('Powerboat')

, ('Kayak')

INSERT INTO Boat (BoatLength, BoatName, BoatTypeID) VALUES

(14, 'Catalina Capri', (SELECT BoatTypeID FROM BoatType WHERE BoatType = 'Sailboat'))

, (24, 'Morgan', (SELECT BoatTypeID FROM BoatType WHERE BoatType = 'Sailboat'))

, (24, 'Hunter', (SELECT BoatTypeID FROM BoatType WHERE BoatType = 'Sailboat'))

, (18, 'Flying Scot', (SELECT BoatTypeID FROM BoatType WHERE BoatType = 'Sailboat'))

insert into Membership (FirstName, LastName, Address1, City, StateAbbrev, PostalCode, HomePhone) values ('Melina', 'MacNish', '55744 Delaware Park', 'Jamaica', 'NY', '11447', '917-405-1841')

insert into Membership (FirstName, LastName, Address1, City, StateAbbrev, PostalCode, HomePhone) values ('Susannah', 'Senten', '55 Summer Ridge Avenue', 'Buffalo', 'NY', '14263', '716-618-3240')

insert into Membership (FirstName, LastName, Address1, City, StateAbbrev, PostalCode, HomePhone) values ('Annemarie', 'Eyers', '312 Waxwing Junction', 'Brooklyn', 'NY', '11254', '212-615-2756')

insert into Membership (FirstName, LastName, Address1, City, StateAbbrev, PostalCode, HomePhone) values ('Noble', 'Water', '4 La Follette Way', 'White Plains', 'NY', '10606', '914-620-9695')

insert into Membership (FirstName, LastName, Address1, City, StateAbbrev, PostalCode, HomePhone) values ('Gustavus', 'Lomasney', '07 5th Lane', 'Port Washington', 'NY', '11054', '516-214-4193')

insert into Membership (FirstName, LastName, Address1, City, StateAbbrev, PostalCode, HomePhone) values ('Myrtia', 'Donaher', '9 Maywood Place', 'Jamaica', 'NY', '11480', '718-624-3406')

insert into Membership (FirstName, LastName, Address1, City, StateAbbrev, PostalCode, HomePhone) values ('Patti', 'Pacey', '2 Garrison Trail', 'Mount Vernon', 'NY', '10557', '914-291-4807')

insert into Membership (FirstName, LastName, Address1, City, StateAbbrev, PostalCode, HomePhone) values ('Nadiya', 'Allmark', '1166 Barnett Terrace', 'Port Washington', 'NY', '11054', '516-827-2632')

insert into Membership (FirstName, LastName, Address1, City, StateAbbrev, PostalCode, HomePhone) values ('Row', 'Fitchell', '91970 Butterfield Park', 'Flushing', 'NY', '11388', '347-768-6798')

insert into Membership (FirstName, LastName, Address1, City, StateAbbrev, PostalCode, HomePhone) values ('Albrecht', 'Coolican', '441 Scott Point', 'Brooklyn', 'NY', '11231', '917-302-0376')

insert into Membership (FirstName, LastName, Address1, City, StateAbbrev, PostalCode, HomePhone) values ('Fayre', 'Pulhoster', '035 Russell Plaza', 'Buffalo', 'NY', '14215', '716-364-4595')

-- etc. Include inserts for each of your tables.

### Querying Data Using SELECT Statements

Note: We’re also satisfying the Programming Objects requirements by creating views at the same time

CREATE OR ALTER VIEW MembershipCount AS

SELECT COUNT(MembershipID) as NumberOfMemberships FROM Membership

GO

SELECT \* FROM MembershipCount

-- Results:

Graphical user interface

Description automatically generated

-- In which cities do our members live and how many from each city?

CREATE OR ALTER VIEW CityReport AS

SELECT City, StateAbbrev, COUNT(MembershipID) as MembershipCount

FROM Membership

GROUP BY City, StateAbbrev

GO

SELECT \* FROM CityReport ORDER BY MembershipCount DESC, StateAbbrev, City

GO

--Results:

Table

Description automatically generated

Continue this for the remainder of your select statements.

## Programming Objects

If you have any stored procedures or functions, paste their CREATE code here.

## User Interface

(3 – 5 artifacts of an external application using your database. I use Access here, but you may use Access, Excel, Python, R, Tableau, or whatever you like – including combinations of each. Screenshots are fine if there’s evidence of your data in the screenshot)

Membership Maintenance ReportGraphical user interface, application

Description automatically generated

Boat Maintenance Form

Graphical user interface

Description automatically generated

Member Boat Report

Graphical user interface, text, application, email

Description automatically generated

*\* except make it meaningful*

## Reflection

The next time you go through the process of creating a database, what will you do differently now that you have been through the whole process?