

CPSC 304 Project Cover Page

Milestone #: 2

Date: October 22, 2025

Group Number: 34

Name	Student Number	CS Alias (Userid)	Preferred E-mail Address
Pouyan Forouzandeh	29206802	c5b9z	Fz.pouyan@gmail.com
Hossein Toutounchi	32372716	x6u50	Hossein.tutunchi@gmail.com
Muhtadi Elmardi	55816979	a9k9j	Muhtadi@student.ubc.ca

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

Project: CommunityCare

Project Description:

The domain of this project covers the management and logistics of non-profit organizations and charities. Our database focuses on tracking donations, campaigns, projects, and beneficiaries. Along with information about donors and donation receipts, the project addresses the relationships between volunteers who participate in campaigns or events, office workers and supervisors who manage projects, and how donations fund projects that benefit individuals or organizations in need.

Database Specifications:

This database combines most of the data necessary to manage a charity or non-profit organization. Some sample use cases for this database are:

- Administrators can track campaigns and the projects they fund.
- Issue and reissue donation receipts for different purposes like reissuing, donation split or donations allocated to multiple campaigns.
- Monitor volunteer participation, including assignment to events and shifts.
- Manage office staff and supervisors.
- Track beneficiaries and link them to the projects or campaigns that help them.
- Allocate donations to multiple projects or campaigns as needed.
- Generate reports and analytics to show donation distribution, volunteer involvement, and project impact.

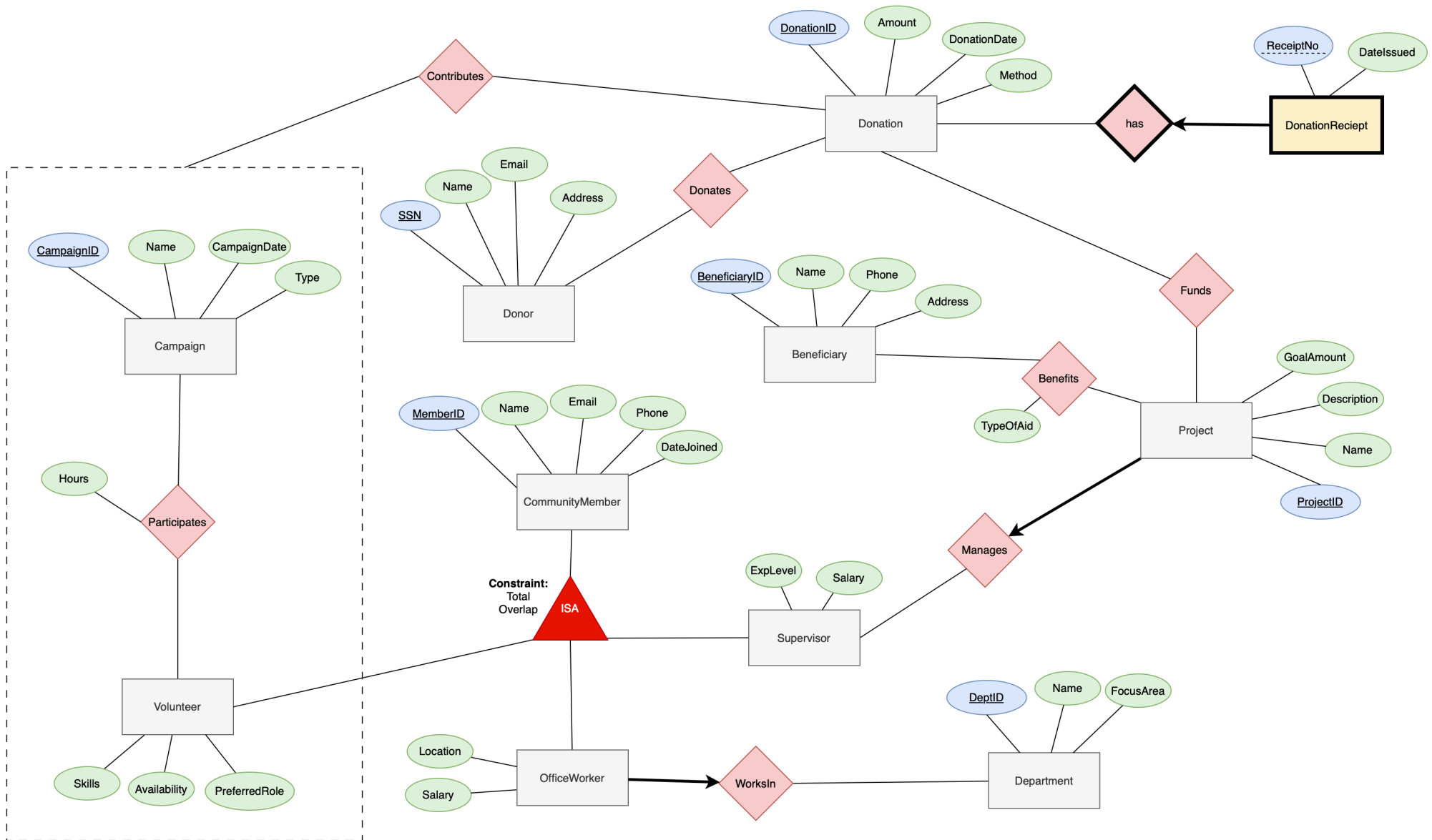
AI Tools Acknowledgment

No, we have not used any AI tools for this assignment.

Note: Some attribute names were modified to avoid conflicts with Oracle reserved keywords. Specifically:

Attributes originally named Date were renamed to DonationDate, CampaignDate, for clarity.

The attribute Level was renamed to ExpLevel because LEVEL is a reserved keyword in Oracle SQL.



Relational Schema

1. Donation(DonationID, Amount, DonationDate, Method)
2. Donor(SSN, Name, Email, Address)
3. DonationReceipt(DonationID, ReceiptNo, DateIssued)
4. Donates(DonationID, SSN)
5. Contributes(DonationID, CampaignID, MemberID)
6. Project (ProjectID, SupervisorMemberID, Name, Description, GoalAmount)
7. Beneficiary(BeneficiaryID, Name, Phone, Address)
8. Funds(ProjectID, DonationID)
9. Benefits(ProjectID, BeneficiaryID, TypeOfAid)
10. CommunityMember(MemberID, Name, Email, Phone, DateJoined)
11. Volunteer(MemberID, Skills, Availability, PreferredRole)
12. OfficeWorker(MemberID, DeptID, Location, Salary)
13. Supervisor(MemberID, ExpLevel, Salary)
14. Department(DeptID, Name, FocusArea)
15. Campaign(CampaignID, Name, CampaignDate, Type)
16. Participates(MemberID, CampaignID, hours)

1. Donation(

DonationID: NUMBER,
Amount: NUMBER,
DonationDate: DATE,
Method: VARCHAR2(30)

)

Primary Key: DonationID

Candidate Key: —

Foreign Key: —

Constraints: Method NOT NULL

2. Donor(

SSN: NUMBER,
Name: VARCHAR2(100),
Email: VARCHAR2(255),
Address: VARCHAR2(200)

)

Primary Key: SSN

Candidate Key: Email

Constraints: Email UNIQUE

3. DonationReceipt(

DonationID: NUMBER,
ReceiptNo: NUMBER,

DateIssued: DATE

)

Primary Key: (DonationID, ReceiptNo)

Foreign Key: DonationID REFERENCES Donation(DonationID)

Constraints: ON DELETE CASCADE

4. Donates(

DonationID: NUMBER,

SSN: NUMBER

)

Primary Key: (DonationID, SSN)

Foreign Keys:

- DonationID REFERENCES Donation(DonationID)
- SSN REFERENCES Donor(SSN)

5. Contributes(

DonationID: NUMBER,

CampaignID: NUMBER,

MemberID: NUMBER

)

Primary Key: (DonationID, CampaignID, MemberID)

Foreign Keys:

- DonationID REFERENCES Donation(DonationID)
- CampaignID REFERENCES Campaign(CampaignID)
- MemberID REFERENCES CommunityMember(MemberID)

6. Project(

ProjectID: NUMBER,

Name: VARCHAR2(100),

Description: VARCHAR2(500),

GoalAmount: NUMBER,

SupervisorMemberID: NUMBER

)

Primary Key: ProjectID

Foreign Key: SupervisorMemberID REFERENCES Supervisor(MemberID)

Constraints: ON DELETE CASCADE, ON UPDATE CASCADE

7. Beneficiary(

BeneficiaryID: NUMBER,

Name: VARCHAR2(20),

Phone: NUMBER,

Address: VARCHAR2(30)

)

Primary Key: BeneficiaryID

8. Funds(

ProjectID: NUMBER,

DonationID: NUMBER

)

Primary Key: (ProjectID, DonationID)

Foreign Keys:

- ProjectID REFERENCES Project(ProjectID)
- DonationID REFERENCES Donation(DonationID)

9. Benefits(

ProjectID: NUMBER,
BeneficiaryID: NUMBER,
TypeOfAid: VARCHAR2(30)

)

Primary Key: (ProjectID, BeneficiaryID)

Foreign Keys:

- ProjectID REFERENCES Project(ProjectID)
- BeneficiaryID REFERENCES Beneficiary(BeneficiaryID)

10. CommunityMember(

MemberID: NUMBER,
Name: VARCHAR2(100),
Email: VARCHAR2(255),
Phone: VARCHAR2(30),
DateJoined: DATE

)

Primary Key: MemberID

Candidate Key: Email

Constraints: Email UNIQUE

11. Volunteer(

MemberID: NUMBER,
Skills: VARCHAR2(100),
Availability: VARCHAR2(100),
PreferredRole: VARCHAR2(100)

)

Primary Key: MemberID

Foreign Key: MemberID REFERENCES CommunityMember(MemberID)

Constraints:

- ON DELETE CASCADE
- ON UPDATE CASCADE

12. OfficeWorker(

MemberID: NUMBER,
Location: VARCHAR2(100),
Salary: NUMERIC,
DeptID: NUMBER

)

Primary Key: MemberID

Foreign Keys:

- MemberID REFERENCES CommunityMember(MemberID)
- DeptID REFERENCES Department(DeptID)

Constraints:

- ON DELETE CASCADE (MemberID)

- ON UPDATE CASCADE (MemberID)
- ON DELETE SET NULL (DeptID)
- ON UPDATE CASCADE (DeptID)

13. Supervisor(

MemberID: NUMBER,
ExpLevel: VARCHAR2(30),
Salary: NUMBER

)

Primary Key: MemberID

Foreign Key: MemberID REFERENCES CommunityMember(MemberID)

Constraints:

- ON DELETE CASCADE
- ON UPDATE CASCADE

14. Department(

DeptID: NUMBER,
Name: VARCHAR2(100),
FocusArea: VARCHAR2(100)

)

Primary Key: DeptID

Candidate Key: Name

15. Campaign(

CampaignID: NUMBER,
Name: VARCHAR2(100),
CampaignDate: DATE,
Type: VARCHAR2(50)

)

Primary Key: CampaignID

16. Participates(

MemberID: NUMBER,
CampaignID: NUMBER,
Hours: NUMERIC

)

Primary Key: (MemberID, CampaignID)

Foreign Keys:

- MemberID REFERENCES Volunteer(MemberID)
- CampaignID REFERENCES Campaign(CampaignID)

Constraints:

- ON DELETE CASCADE
- ON UPDATE CASCADE

ISA Hierarchy

Superclass: CommunityMember(MemberID, Name, Email, Phone, DateJoined)

Subclasses:

- Volunteer(MemberID, Skills, Availability, PreferredRole)

- OfficeWorker(MemberID, Location, Salary, DeptID)
- Supervisor(MemberID, ExpLevel, Salary)

Weak Entity

DonationReceipt is a weak entity dependent on Donation.

- Partial key: ReceiptNo
- Owner key: DonationID
- Identifying relationship: HasReceipt
- Constraint: ON DELETE CASCADE

Functional Dependencies

1. Donation

- DonationID → Amount, DonationDate, Method

2. Donor

- SSN → Name, Email, Address
- Email → SSN, Name, Address

3. DonationReceipt

- (DonationID, ReceiptNo) → DateIssued
- DonationID → possibly multiple ReceiptNo values (weak entity dependency)

4. Donates

- (DonationID, SSN) → (no non-key attributes)

5. Contributes

- (DonationID, CampaignID, MemberID) → (pure relationship table)

6. Project

- ProjectID → Name, Description, GoalAmount, SupervisorMemberID
- SupervisorMemberID → may supervise multiple projects

7. Beneficiary

- BeneficiaryID → Name, Phone, Address

8. Funds

- (ProjectID, DonationID) → (pure M:N relationship, no dependent attributes)

9. Benefits

- (BeneficiaryID, ProjectID) → TypeOfAid

10. CommunityMember

- MemberID → Name, Email, Phone, DateJoined
- Email → MemberID, Name, Phone, DateJoined

11. Volunteer

- MemberID → Skills, Availability, PreferredRole

12. OfficeWorker

- MemberID → DeptID, Location, Salary
- DeptID → DeptInfo/Focus, Location, Salary

13. Supervisor

- MemberID → ExpLevel, Salary

14. Department

- DeptID → Name, FocusArea
- Name → FocusArea

15. Campaign

- CampaignID → Name, CampaignDate, Type

16. Manages

- ProjectID → MemberID (No table indicated in Project)

ISA Hierarchy Dependencies

- CommunityMember → Volunteer / OfficeWorker / Supervisor:
 - MemberID uniquely identifies all subclass attributes.
 - MemberID → Skills, Availability, PreferredRole (if Volunteer)

- MemberID → Location, Salary, DeptID (if OfficeWorker)
- MemberID → ExpLevel, Salary (if in Supervisor)

Weak Entity Dependency

- DonationReceipt
 - DonationID → identifies owner Donation
 - (DonationID, ReceiptNo) → DateIssued
 - ReceiptNo is partial key it identifies receipts only within one Donation

Normalization:

1. **Donation(DonationID, Amount, DonationDate, Method)**

- Candidate Keys: {DonationID}
- FDs: DonationID → Amount, DonationDate, Method
- **Assessment:** LHS is a superkey ⇒ BCNF.

2. **Donor(SSN, Name, Email, Address)**

- Candidate Keys: {SSN}, {Email} (Email is declared UNIQUE)
- FDs:
 - SSN → Name, Email, Address
 - Email → SSN, Name, Address
- **Assessment:** In each FD, LHS is a superkey (SSN or Email) ⇒ BCNF.

3. **DonationReceipt(DonationID, ReceiptNo, DateIssued)**

- Candidate Keys: {(DonationID, ReceiptNo)}
- FDs: (DonationID, ReceiptNo) → DateIssued
- **Assessment:** LHS is the key and weak entity identified by owner + partial key ⇒ BCNF.

4. **Donates(DonationID, SSN)**

- Candidate Keys: {(DonationID, SSN)}
- FDs: none non-trivial (pure associative)
- **Assessment:** No non-trivial FD with non-key LHS ⇒ BCNF.

5. **Contributes(DonationID, CampaignID, MemberID)**

- Candidate Keys: {(DonationID, CampaignID, MemberID)}
- FDs: none non-trivial (pure associative)
- **Assessment:** BCNF.

6. **Project(ProjectID, Name, Description, GoalAmount, SupervisorMemberID)**

- Candidate Keys: {ProjectID}
- FDs: ProjectID → Name, Description, GoalAmount, SupervisorMemberID
- **Assessment:** LHS is a superkey ⇒ BCNF.

7. Beneficiary(BeneficiaryID, Name, Phone, Address)

- Candidate Keys: {BeneficiaryID, Phone}
- FDs: BeneficiaryID → Name, Phone, Address
- **Assessment:** LHS is a superkey ⇒ BCNF

8. Funds(ProjectID, DonationID)

- Candidate Keys: {ProjectID, DonationID}
- FDs: none non-trivial (pure associative)
- **Assessment:** No non-trivial FD with non-key LHS ⇒ BCNF.

9. Benefits(BeneficiaryID, TypeOfAid, ProjectID)

- Candidate Keys: {ProjectID, BeneficiaryID}
- FDs: (BeneficiaryID, ProjectID) → TypeOfAid
- **Assessment:** LHS is a superkey ⇒ BCNF

10. Community Member(MemberID,Name,Email,Phone,DateJoined)

- Candidate Keys: {MemberID}
- MemberID → Name, Email, Phone, DateJoined
- Email → MemberID, Name, Phone, DateJoined
- **Assessment:** LHS is a superkey ⇒ BCNF

11. Volunteer(MemberID,Skills,Availability,PreferredRoles)

- Candidate Keys: {MemberID}
- MemberID → Skills, Availability, PreferredRole
- **Assessment:** LHS is a superkey ⇒ BCNF

12. OfficeWorker(MemberID,Location,Salary)

- Candidate Keys: {MemberID,DeptID}
- MemberID → DeptID, Location, Salary
- **Assessment:** LHS is a superkey ⇒ BCNF

13. Supervisor(MemberID, ExpLevel, Salary)

- Candidate Keys: {MemberID} (and MemberID FK to OfficeWorker/CommunityMember)
- FDs: MemberID → ExpLevel, Salary

- **Assessment:** LHS is a superkey \Rightarrow BCNF.

14. Department(DeptID, Name, FocusArea)

- Candidate Keys: {DeptID};
- FDs:
 - DeptID \rightarrow Name, FocusArea
 - Name \rightarrow FocusArea
- **Assessment:** Name \rightarrow FocusArea violates BCNF because Name is not a superkey.
- **Decomposition:**
 - DepartmentInfo(DeptID, Name)
 - DepartmentFocus(Name, FocusArea)

15. Campaign(CampaignID, Name, CampaignDate, Type)

- Candidate Keys: {CampaignID}
 - CampaignID \rightarrow Name, CampaignDate, Type
- Assessment:** LHS is a superkey \Rightarrow BCNF

16. Participates(CampaignID, MemberID, Hours)

- Candidate Keys: {MemberID, CampaignID}
 - (MemberID, CampaignID) \rightarrow Hours
- Assessment:** LHS is a superkey \Rightarrow BCNF

SQL DDL Statements

CREATE TABLE

```
Donation (  
    DonationID NUMBER PRIMARY KEY,  
    Amount NUMBER,  
    DonationDate DATE,  
    Method VARCHAR2 (30) NOT NULL  
);
```

CREATE TABLE

```
Donor (  
    SSN NUMBER PRIMARY KEY,  
    Name VARCHAR2 (100),  
    Email VARCHAR2 (255) UNIQUE,  
    Address VARCHAR2 (200)  
);
```

CREATE TABLE

```
CommunityMember (  
    MemberID NUMBER PRIMARY KEY,  
    Name VARCHAR2 (100),  
    Email VARCHAR2 (255) UNIQUE,  
    Phone VARCHAR2 (30),  
    DateJoined DATE  
);
```

CREATE TABLE

```
Beneficiary (  
    BeneficiaryID NUMBER PRIMARY KEY,  
    Name VARCHAR2 (20),  
    Phone VARCHAR2 (10),  
    Address VARCHAR2 (30)  
);
```

CREATE TABLE

```
Campaign (  
    CampaignID NUMBER PRIMARY KEY,  
    Name VARCHAR2 (100),  
    CampaignDate DATE,  
    Type VARCHAR2 (50)  
);
```

CREATE TABLE

```
DepartmentInfo (  
    DeptID NUMBER PRIMARY KEY,  
    Name VARCHAR2 (100) NOT NULL UNIQUE  
);
```

-- Needs ON UPDATE CASCADE but oracle does not support this

```
CREATE TABLE
  DepartmentFocus (
    Name VARCHAR2 (100) PRIMARY KEY,
    FocusArea VARCHAR2 (100) NOT NULL,
    FOREIGN KEY (Name) REFERENCES DepartmentInfo (Name) ON DELETE CASCADE
  );
```

```
CREATE TABLE
  Supervisor (
    MemberID NUMBER PRIMARY KEY,
    ExpLevel VARCHAR2 (30),
    Salary NUMBER,
    FOREIGN KEY (MemberID) REFERENCES CommunityMember (MemberID) ON DELETE
  CASCADE
  );
```

```
CREATE TABLE
  Volunteer (
    MemberID NUMBER PRIMARY KEY,
    Skills VARCHAR2 (100),
    Availability VARCHAR2 (100),
    PreferredRole VARCHAR2 (100),
    FOREIGN KEY (MemberID) REFERENCES CommunityMember (MemberID) ON DELETE
  CASCADE
  );
```

```
CREATE TABLE
  Project (
    ProjectID NUMBER PRIMARY KEY,
    Name VARCHAR2 (100),
    Description VARCHAR2 (500),
    GoalAmount NUMBER,
    SupervisorMemberID NUMBER NOT NULL,
    FOREIGN KEY (SupervisorMemberID) REFERENCES Supervisor (MemberID) ON DELETE
  CASCADE
  );
```

-- Needs ON UPDATE CASCADE but oracle does not support this

```
CREATE TABLE
  OfficeWorker (
    MemberID NUMBER PRIMARY KEY,
    Location VARCHAR2 (100),
    Salary NUMBER,
    DeptID NUMBER NOT NULL,
    FOREIGN KEY (MemberID) REFERENCES CommunityMember (MemberID) ON DELETE
  CASCADE,
    FOREIGN KEY (DeptID) REFERENCES DepartmentInfo (DeptID) ON DELETE SET NULL
  );
```

-- Needs ON UPDATE CASCADE but oracle does not support this

CREATE TABLE

```
DonationReceipt (  
    ReceiptNo NUMBER,  
    DonationID NUMBER,  
    DateIssued DATE,  
    PRIMARY KEY (DonationID, ReceiptNo),  
    FOREIGN KEY (DonationID) REFERENCES Donation (DonationID) ON DELETE CASCADE  
);
```

CREATE TABLE

```
Donates (  
    DonationID NUMBER,  
    SSN NUMBER,  
    PRIMARY KEY (DonationID, SSN),  
    FOREIGN KEY (DonationID) REFERENCES Donation (DonationID) ON DELETE CASCADE,  
    FOREIGN KEY (SSN) REFERENCES Donor (SSN) ON DELETE CASCADE  
);
```

CREATE TABLE

```
Funds (  
    ProjectID NUMBER,  
    DonationID NUMBER,  
    PRIMARY KEY (ProjectID, DonationID),  
    FOREIGN KEY (ProjectID) REFERENCES Project (ProjectID) ON DELETE CASCADE,  
    FOREIGN KEY (DonationID) REFERENCES Donation (DonationID) ON DELETE CASCADE  
);
```

CREATE TABLE

```
Benefits (  
    BeneficiaryID NUMBER,  
    ProjectID NUMBER,  
    TypeOfAid VARCHAR2 (30),  
    PRIMARY KEY (ProjectID, BeneficiaryID),  
    FOREIGN KEY (ProjectID) REFERENCES Project (ProjectID) ON DELETE CASCADE,  
    FOREIGN KEY (BeneficiaryID) REFERENCES Beneficiary (BeneficiaryID) ON DELETE  
CASCADE  
);
```

CREATE TABLE

```
Contributes (  
    DonationID NUMBER,  
    CampaignID NUMBER,  
    MemberID NUMBER,  
    PRIMARY KEY (DonationID, CampaignID, MemberID),  
    FOREIGN KEY (DonationID) REFERENCES Donation (DonationID) ON DELETE CASCADE,  
    FOREIGN KEY (CampaignID) REFERENCES Campaign (CampaignID) ON DELETE CASCADE,  
    FOREIGN KEY (MemberID) REFERENCES CommunityMember (MemberID) ON DELETE  
CASCADE  
);
```

CREATE TABLE

Participates (

MemberID NUMBER,

CampaignID NUMBER,

Hours NUMBER,

PRIMARY KEY (MemberID, CampaignID),

FOREIGN KEY (MemberID) REFERENCES Volunteer (MemberID) ON DELETE CASCADE,

FOREIGN KEY (CampaignID) REFERENCES Campaign (CampaignID) ON DELETE CASCADE

);

Insert Statements

-- Donation

INSERT INTO Donation VALUES (101, 200, '2025-01-15', 'Credit Card');

INSERT INTO Donation VALUES (102, 500, '2025-02-10', 'PayPal');

INSERT INTO Donation VALUES (103, 150, '2025-03-05', 'Cash');

INSERT INTO Donation VALUES (104, 800, '2025-04-20', 'Bank Transfer');

INSERT INTO Donation VALUES (105, 300, '2025-05-12', 'Credit Card');

INSERT INTO Donation VALUES (106, 1000, '2025-06-08', 'Cheque');

-- Donor

INSERT INTO Donor VALUES (1111, 'Jane Doe', 'jane.doe@gmail.com', '23 King St, Toronto');

INSERT INTO Donor VALUES (2222, 'Brian Chen', 'brian.chen@gmail.com', '90 Queen Rd, Ottawa');

INSERT INTO Donor VALUES (3333, 'Ryan Zf', 'ryan.zf@email.com', '12 Maple Ave, Vancouver');

INSERT INTO Donor VALUES (4444, 'David Miller', 'david.miller@email.com', '55 Elm St, Montreal');

INSERT INTO Donor VALUES (5555, 'Eva Jackson', 'eva.jackson@email.com', '78 Pine Dr, Calgary');

-- CommunityMember (used supervisors numbers)

INSERT INTO CommunityMember VALUES(401, 'Ava Green', 'ava.green@ngo.org', '604-555-1101', '2023-01-15');

INSERT INTO CommunityMember VALUES(402, 'Ben Carter', 'ben.carter@ngo.org', '604-555-1102', '2023-03-10');

INSERT INTO CommunityMember VALUES(403, 'Cara Gomez', 'cara.gomez@ngo.org', '604-555-1103', '2023-06-21');

INSERT INTO CommunityMember VALUES(404, 'Derek Shaw', 'derek.shaw@ngo.org', '604-555-1104', '2024-02-05');

INSERT INTO CommunityMember VALUES(405, 'Ella Watts', 'ella.watts@ngo.org', '604-555-1105', '2024-09-12');

INSERT INTO CommunityMember VALUES(406, 'Finn Patel', 'finn.patel@ngo.org', '604-555-1106', '2025-01-09');

-- Beneficiary

INSERT INTO Beneficiary VALUES(1001, 'David', 6041245789, '25 Kent road Vancouver');

INSERT INTO Beneficiary VALUES(1002, 'Mamad', 7781452768, '122 Lonsdale North Vancouver');

INSERT INTO Beneficiary VALUES(1003, 'Hasan', 2345896347, '535 Shaw Ave Coquitlam');

INSERT INTO Beneficiary VALUES(1004, 'John', 6048897514, '122 Bluemountain Vancouver');

INSERT INTO Beneficiary VALUES(1005, 'Alex', 7784977896, '3033 East Town road Burnaby');

-- Campaign

INSERT INTO Campaign VALUES (3001, 'Winter Food Drive', '2025-01-10', 'Food');

INSERT INTO Campaign VALUES (3002, 'Back-to-School Kits', '2025-08-01', 'Education');

INSERT INTO Campaign VALUES (3003, 'Community Garden', '2025-04-15', 'Environment');

INSERT INTO Campaign VALUES (3004, 'Mobile Clinic', '2025-05-22', 'Health');

INSERT INTO Campaign VALUES (3005, 'Shelter Renovation', '2025-03-05', 'Housing');

-- DepartmentInfo

INSERT INTO DepartmentInfo VALUES (501, 'Community Services');

INSERT INTO DepartmentInfo VALUES (502, 'Finance');

INSERT INTO DepartmentInfo VALUES (503, 'Health and Safety');

INSERT INTO DepartmentInfo VALUES (504, 'Education');

INSERT INTO DepartmentInfo VALUES (505, 'Environmental Support');

-- DepartmentFocus

INSERT INTO DepartmentFocus VALUES ('Community Services', 'Poverty Relief');

INSERT INTO DepartmentFocus VALUES ('Finance', 'Fund Management');

INSERT INTO DepartmentFocus VALUES ('Health and Safety', 'Medical Assistance');

INSERT INTO DepartmentFocus VALUES ('Education', 'Youth Literacy Programs');

INSERT INTO DepartmentFocus VALUES ('Environmental Support', 'Clean Energy');

-- Supervisor

INSERT INTO Supervisor VALUES (401, 'Senior', 85000);

INSERT INTO Supervisor VALUES (402, 'Senior', 90000);

INSERT INTO Supervisor VALUES (403, 'Intermediate', 70000);

INSERT INTO Supervisor VALUES (404, 'Junior', 60000);

INSERT INTO Supervisor VALUES (405, 'Intermediate', 75000);

INSERT INTO Supervisor VALUES (406, 'Junior', 58000);

-- Volunteer

INSERT INTO Volunteer VALUES (401, 'First aid; distribution', 'Weekends', 'Field helper');

INSERT INTO Volunteer VALUES (402, 'Driving; logistics', 'Evenings', 'Driver');

INSERT INTO Volunteer VALUES (404, 'Event setup', 'Flexible', 'Setup crew');

INSERT INTO Volunteer VALUES (405, 'Tutoring', 'Weekdays', 'Mentor');

INSERT INTO Volunteer VALUES (406, 'Food prep', 'Weekends', 'Kitchen support');

-- Project

INSERT INTO Project VALUES (301, 'Food Drive', 'Collect and distribute food for low-income families', 10000, 401);

INSERT INTO Project VALUES (302, 'Clean Water', 'Install water purification systems in rural areas', 20000, 402);

INSERT INTO Project VALUES (303, 'School Supplies', 'Provide educational kits for children', 12000, 403);

INSERT INTO Project VALUES (304, 'Shelter Renovation', 'Repair shelters for homeless families', 25000, 404);

INSERT INTO Project VALUES (305, 'Healthcare Outreach', 'Medical support for remote communities', 18000, 405);

-- OfficeWorker

```
INSERT INTO OfficeWorker VALUES (402, 'HQ 3F', 72000, 501);  
INSERT INTO OfficeWorker VALUES (403, 'Finance Wing', 68000, 502);  
INSERT INTO OfficeWorker VALUES (405, 'Clinic Trailer', 70000, 503);  
INSERT INTO OfficeWorker VALUES (406, 'Warehouse', 61000, 505);  
INSERT INTO OfficeWorker VALUES (404, 'HQ 2F', 60000, 504);
```

-- DonationReceipt

```
INSERT INTO DonationReceipt VALUES (9001, 101, '2025-01-16');  
INSERT INTO DonationReceipt VALUES (9002, 102, '2025-02-11');  
INSERT INTO DonationReceipt VALUES (9003, 103, '2025-03-06');  
INSERT INTO DonationReceipt VALUES (9004, 104, '2025-04-21');  
INSERT INTO DonationReceipt VALUES (9005, 105, '2025-05-13');  
INSERT INTO DonationReceipt VALUES (9006, 106, '2025-06-09');
```

-- Donates

```
INSERT INTO Donates VALUES (101, 1111);  
INSERT INTO Donates VALUES (102, 2222);  
INSERT INTO Donates VALUES (103, 3333);  
INSERT INTO Donates VALUES (104, 4444);  
INSERT INTO Donates VALUES (105, 5555);  
INSERT INTO Donates VALUES (106, 1111);
```

-- Funds

```
INSERT INTO Funds VALUES(301, 103);  
INSERT INTO Funds VALUES(301,102);  
INSERT INTO Funds VALUES(302,104);  
INSERT INTO Funds VALUES(303,102);  
INSERT INTO Funds VALUES(304,101);
```

-- Benefits

```
INSERT INTO Benefits VALUES (1001, 301, 'Cash Payment');  
INSERT INTO Benefits VALUES (1002, 302, 'Medical Supplies');  
INSERT INTO Benefits VALUES (1003, 303, 'Scholarship');  
INSERT INTO Benefits VALUES (1004, 304, 'Shelter Support');  
INSERT INTO Benefits VALUES (1005, 305, 'Groceries');
```

-- Contributes

```
INSERT INTO Contributes VALUES (101, 3001, 401);  
INSERT INTO Contributes VALUES (102, 3002, 402);  
INSERT INTO Contributes VALUES (103, 3003, 403);  
INSERT INTO Contributes VALUES (104, 3004, 404);  
INSERT INTO Contributes VALUES (105, 3005, 405);  
INSERT INTO Contributes VALUES (106, 3001, 406);
```

-- Participates

```
INSERT INTO Participates VALUES (401, 3001, 12);  
INSERT INTO Participates VALUES (402, 3001, 8);  
INSERT INTO Participates VALUES (404, 3002, 15);  
INSERT INTO Participates VALUES (405, 3004, 10);  
INSERT INTO Participates VALUES (406, 3005, 9);
```