April 19, 2018

S307 Sp2018: Database Management

Kelley School of business

Indiana University, Bloomington

DreamCorps

World Peach Tech

Zi Chai, Yimei Tong, Huaqiu Zheng, Hengyi Hu

**Table of Contents**

[1 Introduction 2](#_Toc511583072)

[2 Design of the Database 2](#_Toc511583073)

[2.1 Conceptual Design 2](#_Toc511583074)

[2.2 Logical Design 4](#_Toc511583075)

[2.3 Physical Design 4](#_Toc511583076)

[3 Implementation of the Database 9](#_Toc511583101)

[4 User Interface 23](#_Toc511583102)

[4.1 Design 23](#_Toc511583103)

[4.2 Implementation 23](#_Toc511583104)

[5 Conclusion 23](#_Toc511583105)

# 1 Introduction

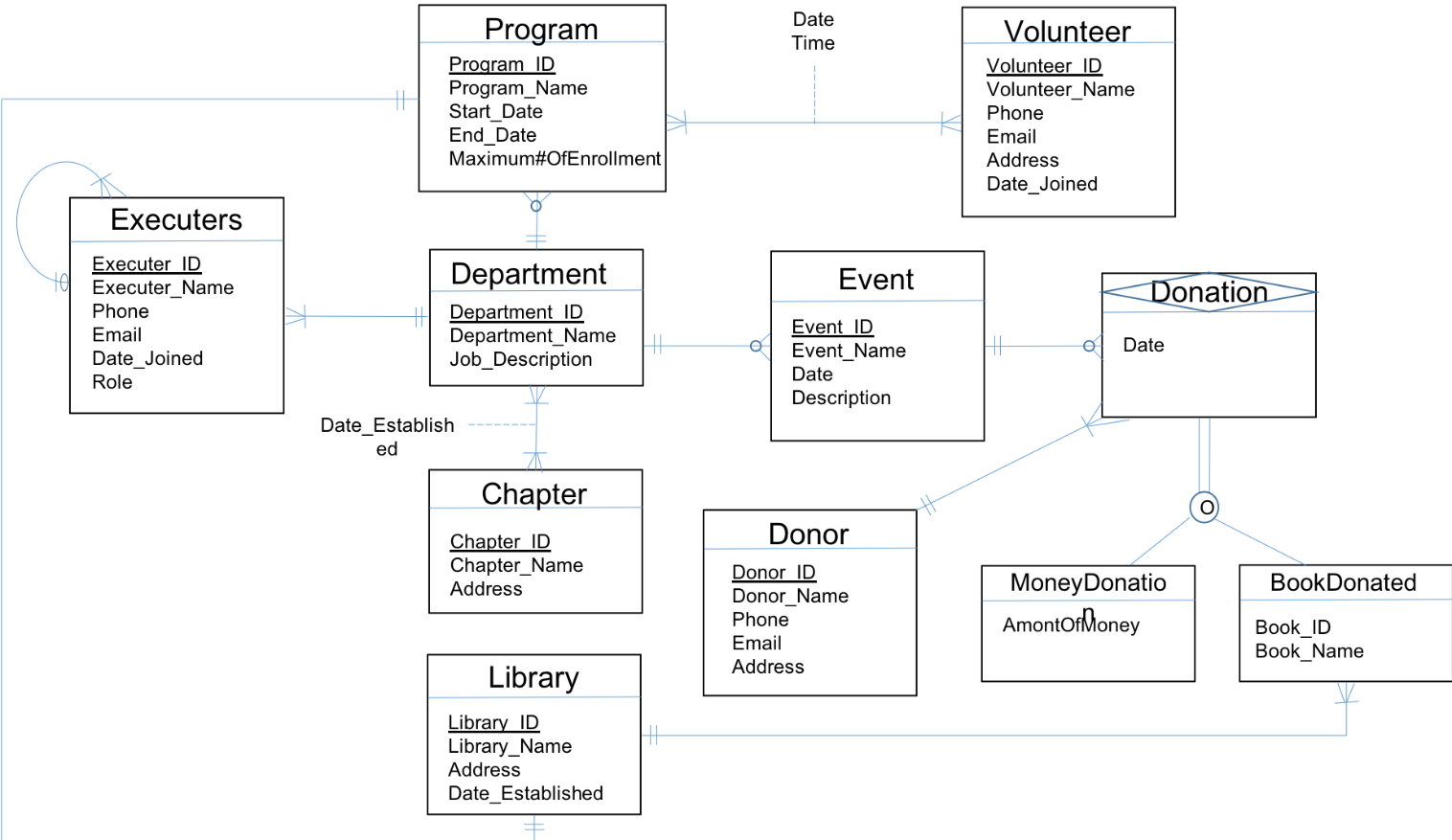
In this project, our team’s goal is to design and implement a database for Dream Corps International which is a nonprofit organization. It aims to improve the environment for learning and development of children in rural China and encourages them to realize their full potential, unfettered by the lack of resources and the economic conditions that they are born into. It also involves privileged youth in projects that cultivate their sense of social responsibility.

We designed an entity relationship diagram that maps out operation processes of the Dream Corps. Our ER diagram demonstrates how each department organizes events to collect donations from donors. The donation includes money and books which will be used in building libraries in rural areas in China. In order to do that, Executers lead departments in Dream Corp. and recruit volunteers for further development. We used PowerPoint to build this diagram. After we completed our ER diagram, we move on to logical design of our system which reveals data in the form of tables. It can be easily understood and used even by those unfamiliar with the underlying theory.

# 2 Design of the Database

## 2.1 Conceptual Design

The entity relationship model (ER Diagram) graphically represents all entities and their relationships to each other based on cardinality in the database. In our model, Dream Corps. has set its chapters all over the universities and different departments which hold different programs and events. Programs are aimed to recruit volunteers while events are aimed to raise donation. We collect donations from donors and also record their information. If a donor donates a book, it will go to Dream Corps library which will deliver to students who need help. If a donor donates cash, all the money combined will go to Dream Corp. headquarter to help schools innovate their facilities.

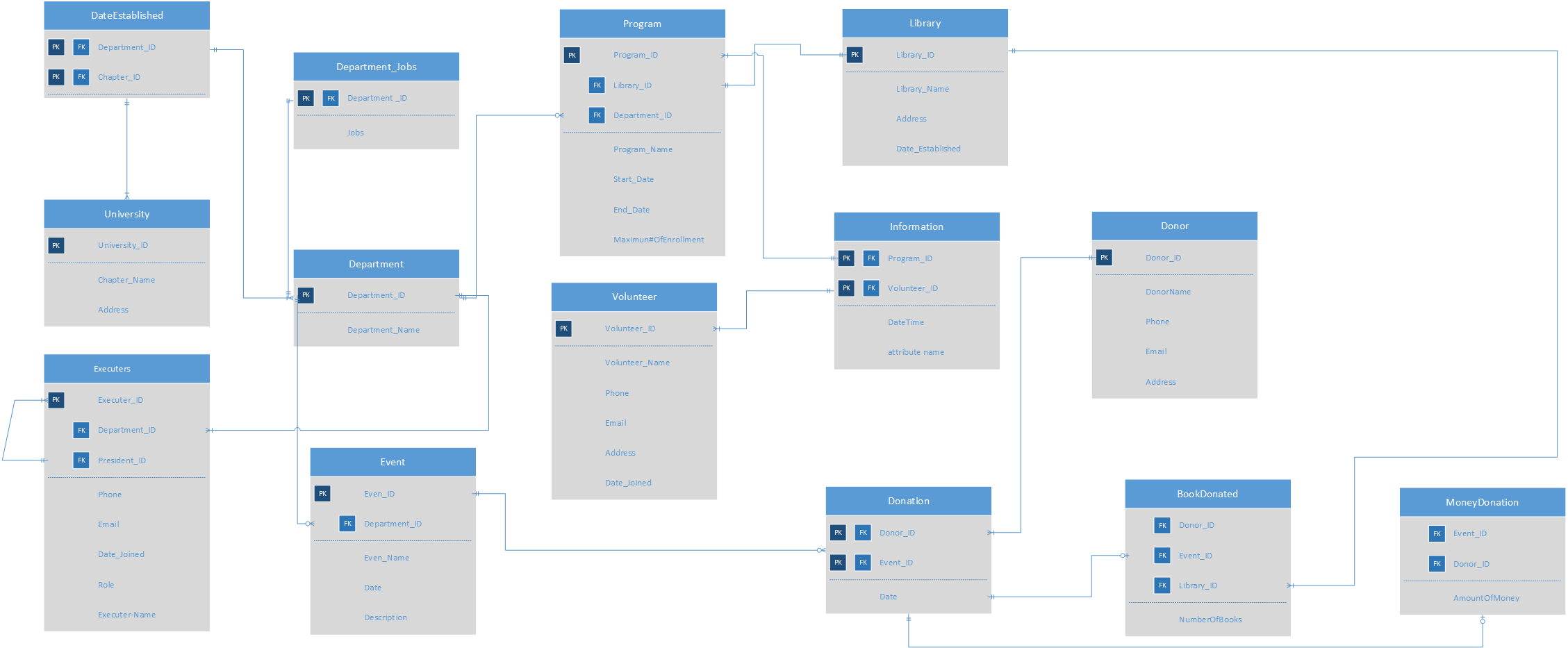


Assumptions:

1. One chapter can have one and more departments.
2. Each department can have one or more executers, and one executer belongs to only one department.
3. Each executer can supervise one or more members, each member can be supervised by only one executer.
4. Each department can hold zero or more programs, but one program can be held by only one department.
5. A program can have one or more volunteers, and individual volunteers can attend one or more programs.
6. Each department can hold zero or move events, but one event can be held by only one department.
7. One event can raise zero or more donation, donation can be raised only through events. Donation can be either money or books, or both.
8. A donor can make one or more donations, and donations can only made by donors.
9. Each library can have one or more books, but one book can only go to one library.

## 2.2 Logical Design

Description: The relationship model here is pretty straight forward. All we did was setting 'id's are primary keys to differentiate an entity to the other and create attribute(s) as foreign keys to connect with other entities. However, one trick thing we did was that we create a few entities that are not shown on our ERD because they are created just for associative entity purpose, such as Information, DateEstablished. We also made a subtype entity for MoneyDonation and BookDonation which were subtyped with Donor. However, we cannot directly do that because some donors may only donate money not the book or the other way around. So we built a Donation entity to connect not only MoneyDonation and BookDonation but also show this supertype/subtype relationship with Donor.



## 2.3 Physical Design

*Data Dictionary:*

In the data dictionary, our team concluded the contents, data type, valid value if null is allowed, and description of data in our database. Data dictionary provides necessary information for our team to create our logical design and write the SQL code.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Data Type | Valid Value | NULL Allowed | Description |
| Department\_id | Character | N/A | No | Department identification number |
| Department Name | Character | N/A | No |  |
| Job | Character | Volunteer | No | Tasks assigned |

## Table 1: Data Dictionary for DEPARTMENT Entity Type

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Data Type | Valid Value | NULL Allowed | Description |
| Program\_id | Number | N/A | No | Program identification number |
| Program\_Name | Character | N/A | No |  |
| Start\_Date | Time | mm-dd-yy | No | When will a program starts |
| End\_Date | Time | mm-dd-yy | No | When will a program ends |
| Maximun#ofEnrollment | Number | 1-20 | No | How many volunteers needed |

## Table 2: Data Dictionary for PROGRAM Entity Type

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Data Type | Valid Value | NULL Allowed | Description |
| Library\_ID | Character | N/A | No | Library identification number |
| Library\_Name | Character |  | No | Name of the Library |
| Address | Character |  | No | Address of a library is |
| Date\_Established | Date | mm-dd-yy | No | The construction date of a library |

## Table 3: Data Dictionary for LIBRARY Entity Type

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Data Type | Valid Value | NULL Allowed | Description |
| University\_id | Character | N/A | No | University identification number |
| University\_Name | Character | N/A | No | Name of University |
| Address | Character |  | No | Address of an University |

## Table 4: Data Dictionary for UNIVERSITY Entity Type

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Data Type | Valid Value | Null Allowed | Description |
| Volunteer\_ID | Number | N/A | No | Volunteer identification number |
| Volunteer\_Name | Character | N/A | No | Volunteer's name |
| Phone | Character | ###-###-#### | Yes | Volunteer's phone number |
| Email | Character |  | No | Volunteer's Email address |
| Address | Character |  | Yes | Volunteer's current addreess |
| Date\_Joined | Date | mm-dd-yy | No | When a volunteer starts working for Dream Corps. |

## Table 5: Data Dictionary for VOLUNTEER Entity Type

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Data Type | Valid Value | Null Allowed | Description |
| Event\_ID | Character | ########## | No | Event identification number |
| Event\_Name | Character | N/A | No | Name for an event |
| Date | Date | mm-dd-yy | Yes | Date for an event |
| Description | Character | N/A | No | What an event is about |

## Table 6: Data Dictionary for EVENT Entity Type

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Data Type | Valid Value | Null Allowed | Description |
| Executer\_ID | Character | N/A | No | Executer identification number |
| Executer\_Name | Character | N/A | No | An executer's name |
| Phone | Character | ###-###-#### | No | An executer's phnone number |
| Email | Character |  | No | An executer's Email address |
| Date\_Joined | Date | mm-dd-yy | No | Date when an executer joined Dream Corp. |
| Role | Character | N/A | No | The jobs an executer has |

## Table 7: Data Dictionary for EXECUTER Entity Type

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Attribute | Data Type | Valid Value | Null Allowed | Description |
| Donor\_ID | Character | N/A | No | Donor identification number |
| Donor\_Name | Character | N/A | Yes | Name for a donor |
| DPhone | Number | ###-###-#### | Yes | Phone number for donors |
| DEmail | Character |  | Yes | Email address for donors |
| DAddress | Mixed |  | Yes | Current address for donors |
| MoneyDonated | Number | N/A | No | Amount of money a donor donates |
| BookDonated | Number | N/A | No | Number of books a donor donates |

## Table 8: Data Dictionary for DONOREntity Type

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Attribute* | Data Type | Valid Value | Null Allowed | *Description* |
| Department\_ID | Character | N/A | No | Department identification number |
| University\_ID | Character | N/A | No | University identification number |

## Table 9: Data Dictionary for DATEESTABLISHED Entity Type

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Attribute* | Data Type | Valid Value | Null Allowed | *Description* |
| Program\_ID | Character | N/A | No | Program identification number |
| Volunteer\_ID | Character | N/A | No | Volunteer identification number |
| DateTime | Date | N/A | No | Date |

## Table 10: Data Dictionary for INFORMATION Entity Type

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Attribute* | Data Type | Valid Value | Null Allowed | *Description* |
| Donor\_ID | Character | N/A | No | Donor identification number |
| Library\_ID | Character | N/A | No | Library identification number |
| Event\_ID | Character | N/A | No | Event identification number |
| NumberofBooks | Number | N/A | No | Number of books |
| BDonatedate | Date | N/A | Yes | Book Donated date |

## Table 11: Data Dictionary for BOOKDONATEDEntity Type

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Attribute* | Data Type | Valid Value | Null Allowed | *Description* |
| Donor\_ID | Character | N/A | No | Donor identification number |
| Event\_ID | Character | N/A | No | Event identification number |
| Amountofmoney | Number | N/A | No | Amount of money donated |
| Mdonatedate | Date | N/A | Yes | Money donated date |

## Table 12: Data Dictionary for MONEYDONATED Entity Type

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Attribute* | Data Type | Valid Value | Null Allowed | *Description* |
| Donor\_ID | Character | N/A | No | Donor identification number |
| Event\_ID | Character | N/A | No | Event identification number |
| Date | Date | N/A | Yes | Date of donation |

## Table 13: Data Dictionary for DONATION Entity Type

# 3 Implementation of the Database

Department Table Null? Type

Department\_id No Char(10)

Department\_name No Varchar2(20)

Job\_description No Varchar2(500)

Program Table Null? Type

Program\_ID No Char(10)

Program\_Name No Varchar2(50)

Pstart\_Date No Date(7)

Pend\_Date No Date(7)

Pmaxenroll No Number(22)

Library\_ID No Char(10)

Department\_ID No Char(10)

Library Table Null? Type

Library\_ID No Char(10)

Library\_Name No Varchar2(30)

Laddress Yes Varchar2(50)

Date\_Established Yes Date(7)

University Table Null? Type

University\_ID No Char(10)

University\_Name No Varchar2(50)

Uaddress Yes Varchar2(50)

Volunteer Table Null? Type

Volunteer\_ID No Char(10)

Volunteer\_Name No Varchar2(22)

Vphone Yes Char(10)

Vemail No Varchar2(50)

Vaddress Yes Varchar2(50)

Date\_Joined No Date(7)

Event Table Null? Type

Event\_ID No Char(10)

Event\_Name No Varchar2(50)

Edate Yes Date(7)

Edescription No Varchar2(200)

Department\_ID Yes Char(10)

Executers Table Null? Type

Executer\_ID No Char(10)

Executer\_Name No Varchar2(20)

Phone No Char(20)

Email No Varchar2(25)

Department\_ID No Char(10)

President\_ID No Char(10)

Date\_Joined No Varchar2(15)

Role No Varchar2(15)

Donor Table Null? Type

Donor\_ID No Char(10)

Donor\_Name Yes Varchar2(22)

Dphone Yes Char(20)

Demail No Varchar2(50)

Daddress Yes Varchar2(50)

Type Yes Varchar2(20)

Bookdonated Null? Type

Donor\_ID No Char(10)

Library\_ID No Char(10)

Event\_ID No Char(10)

Numberofbooks No Number(22)

BDonatedate Yes Date(7)

Moneydonated Null? Type

Donor\_ID No Char(10)

Event\_ID No Char(10)

Amountofmoney No Number(22)

MDonatedate Yes Date(7)

Dateestablished Null? Type

Department\_ID No Char(10)

University\_ID No Char(10)

Information Null? Type

Volunteer\_ID No Char(10)

Program\_ID No Char(10)

**Scripts:**

create table Department

(Department\_ID char(10) not null,

Department\_name varchar(20) not null,

Job\_description varchar(500) not null,

constraint pk\_Department PRIMARY KEY(Department\_ID)

);

Create Table University(

University\_ID char(10) primary key not null,

University\_Name varchar2(50) not null,

UAddress varchar(50)

);

Create Table Library(

Library\_ID char(10) primary key not null,

Library\_Name varchar2(30) not null,

LAddress varchar(50),

Date\_Established date

);

create table Program

(Program\_ID char(10) not null,

Program\_Name varchar(50) not null,

PStart\_Date date not null,

PEnd\_Date date not null,

PMaxEnroll number(20) not null,

Library\_ID char(10) not null,

Department\_ID char(10) not null,

constraint pk\_Program PRIMARY KEY(Program\_ID),

Constraint fk\_ProgramLibrary foreign key (Library\_ID)

References Library(Library\_ID) on delete cascade,

Constraint fk\_ProgramDepartment foreign key (Department\_ID)

References Department(Department\_ID) on delete cascade

);

Create Table Volunteer(

Volunteer\_ID char(10) primary key not null,

Volunteer\_FName varchar2(30) not null,

Volunteer\_LName varchar2(30) not null,

VPhone char(20),

VEmail varchar2(50) not null,

VAddress varchar2(100),

Date\_Joined date not null

);

create table Event

(Event\_ID char(10) not null,

Event\_Name varchar(50) not null,

EDate Date not null,

EDescription varchar(200) not null,

Department\_ID char(10),

constraint pk\_Event primary key(Event\_ID),

Constraint fk\_DepartmentEvent foreign key (Department\_ID)

References Department(Department\_ID) on delete cascade

);

Create Table Donor(

Donor\_ID char(10) primary key not null,

Donor\_FName varchar2(30),

Donor\_LName varchar2(30),

DPhone varchar2(20),

DEmail varchar(50) not null,

DAddress varchar(50),

Type varchar(20) check (Type in ('Money Donor','Book Donor','Both'))

);

Create table DateEstablished(

Department\_ID char(10) not null references Department (Department\_ID),

University\_ID char(10) not null references University (University\_ID),

Date\_Established Date,

Primary key(Department\_ID, University\_ID)

);

Create table BookDonated(

Donor\_ID char(10) not null references Donor (Donor\_ID),

Library\_ID char(10) not null references Library (Library\_ID),

Event\_ID char(10) not null references Event (Event\_ID),

NumberOfBooks number(10) not null,

BDonateDate Date

);

Create table MoneyDonated(

Donor\_ID char(10) not null references Donor (Donor\_ID),

Event\_ID char(10) not null references Event (Event\_ID),

AmountOfMoney number(10) not null,

MDonateDate Date

);

create table Executers

(

Executer\_ID char(10)

constraint pk\_Executers primary key,

Executer\_Fname varchar2(20)

constraint Executers\_Executer\_Fname not null,

Executer\_Lname varchar2(20)

constraint Executers\_Executer\_Lname not null,

Phone varchar2(20)

constraint Executers\_Phone not null,

Email varchar2(25)

constraint Excuters\_Email not null,

Department\_ID char(10) not null,

President\_ID char(10),

Date\_Joined date not null,

Role varchar(15)not null

check(Role in ('director','VP', 'president')),

Constraint fk\_DepartmentExecuters foreign key (Department\_ID)

References Department(Department\_ID) on delete cascade,

Constraint fk\_President foreign key (President\_ID)

References Executers(Executer\_ID) on delete cascade

);

create table Information

(

Volunteer\_ID char(10) not null,

Program\_ID char (10) not null,

constraint pk\_vpid primary key (Volunteer\_ID, Program\_ID),

constraint fk\_volunteer foreign key (Volunteer\_ID)

references Volunteer(Volunteer\_ID) on delete cascade,

constraint fk\_program foreign key (Program\_ID)

references Program(Program\_ID) on delete cascade

);

insert into Department (Department\_ID, Department\_name,Job\_description)

Values(0000000001, 'Activity','Hold events' );

insert into Department (Department\_ID, Department\_name,Job\_description)

Values(0000000002, 'Internal Management', 'Communication with members');

insert into Department (Department\_ID, Department\_name,Job\_description)

Values(0000000003, 'Finance', 'Record cash inflow and outflow');

insert into Department (Department\_ID, Department\_name,Job\_description)

Values(0000000004, 'Marketing', 'Create posters');

insert into Department (Department\_ID, Department\_name,Job\_description)

Values(0000000005,'Public Relation','Marketing promotion');

Insert into University(University\_ID, University\_Name, UAddress)

values(1000000000,'Indiana University Bloomington','107 S Indiana Ave, Bloomington, IN 47405');

Insert into University(University\_ID, University\_Name, UAddress)

values(2000000000,'University of Texas at Austin','Austin, TX 78712');

Insert into University(University\_ID, University\_Name, UAddress)

values(3000000000,'Michigan State University',' 220 Trowbridge Rd, East Lansing, MI 48824');

Insert into University(University\_ID, University\_Name, UAddress)

values(4000000000,'Carnegie Mellon University','5000 Forbes Ave, Pittsburgh, PA 15213');

Insert into University(University\_ID, University\_Name, UAddress)

values(5000000000,'University of Chicago','5801 S Ellis Ave, Chicago, IL 60637');

Insert into Library(Library\_ID, Library\_Name, LAddress, Date\_Established)

Values(1100000000,'ABC','155 S, Greenwood, IN 46142',to\_date('5-20-98','MM-DD-YY'));

Insert into Library(Library\_ID, Library\_Name, LAddress, Date\_Established)

Values(2200000000,'BCD',' 1251 US Highway, Greenwood, IN 4614',to\_date('5-20-98','MM-DD-YY'));

Insert into Library(Library\_ID, Library\_Name, LAddress, Date\_Established)

Values(3300000000,'CDE','2687 E, Plainfield, IN 46168',to\_date('6-24-98','MM-DD-YY'));

Insert into Library(Library\_ID, Library\_Name, LAddress, Date\_Established)

Values(4400000000,'DEF','10791 E, Avon, IN 46123',to\_date('6-19-04','MM-DD-YY'));

Insert into Library(Library\_ID, Library\_Name, LAddress, Date\_Established)

Values(5500000000,'EFG','6231N, Indianapolis, IN 46220',to\_date('6-20-11','MM-DD-YY'));

Insert into Library(Library\_ID, Library\_Name, LAddress, Date\_Established)

Values(6600000000,'Qinghua','6231N, Indianapolis, IN 46220',to\_date('6-20-11','MM-DD-YY'));

Insert into Program(Program\_ID, Program\_Name, PStart\_Date, PEnd\_Date, PMaxEnroll, Library\_ID, Department\_ID)

Values(1010000000,'Little 5', to\_date('5-21-15','MM-DD-YY'), to\_date('9-1-16','MM-DD-YY'),15,1100000000,0000000001);

Insert into Program(Program\_ID, Program\_Name, PStart\_Date, PEnd\_Date, PMaxEnroll, Library\_ID, Department\_ID)

Values(1020000000,'Little 4', to\_date('5-21-16','MM-DD-YY'), to\_date('9-1-17','MM-DD-YY'),15,2200000000,0000000002);

Insert into Program(Program\_ID, Program\_Name, PStart\_Date, PEnd\_Date, PMaxEnroll, Library\_ID, Department\_ID)

Values(1030000000,'Little 3', to\_date('5-21-10','MM-DD-YY'), to\_date('9-1-11','MM-DD-YY'),15,3300000000,0000000003);

Insert into Program(Program\_ID, Program\_Name, PStart\_Date, PEnd\_Date, PMaxEnroll, Library\_ID, Department\_ID)

Values(1040000000,'Little 2', to\_date('5-21-09','MM-DD-YY'), to\_date('9-1-10','MM-DD-YY'),15,4400000000,0000000004);

Insert into Program(Program\_ID, Program\_Name, PStart\_Date, PEnd\_Date, PMaxEnroll, Library\_ID, Department\_ID)

Values(1050000000,'Little 1', to\_date('5-21-05','MM-DD-YY'), to\_date('9-1-06','MM-DD-YY'),15,5500000000,0000000005);

insert into Volunteer (Volunteer\_ID,Volunteer\_FName,Volunteer\_LName,VPhone,VEmail,VAddress,Date\_Joined)

values (7398214390,'Makala', 'Johnson','829-792-7912','majohla@sina.com','32 Volunteer Revenue,LA,California',to\_date('5-21-05','MM-DD-YY'));

insert into Volunteer (Volunteer\_ID,Volunteer\_FName,Volunteer\_LName,VPhone,VEmail,VAddress,Date\_Joined)

values (5901958159,'Bill', 'Rite','532-902-0328','bijoritaa@gmail.com','Luck Way 532,Portland,Oregon',to\_date('5-21-05','MM-DD-YY'));

insert into Volunteer (Volunteer\_ID,Volunteer\_FName,Volunteer\_LName,VPhone,VEmail,VAddress,Date\_Joined)

values (8092412568,'Abigial', 'Gryffth','118-120-529','bigmouth@iu.edu','Made Revenue 031, Minniapolis,Minnesoda',to\_date('5-21-05','MM-DD-YY'));

insert into Volunteer (Volunteer\_ID,Volunteer\_FName,Volunteer\_LName,VPhone,VEmail,VAddress,Date\_Joined)

values (7583901275,'Blake', 'Harden','192-509-0950','bladen@iu.edu','5812 Convenanter DR. Bloomington, Indiana',to\_date('5-30-05','MM-DD-YY'));

insert into Volunteer (Volunteer\_ID,Volunteer\_FName,Volunteer\_LName,VPhone,VEmail,VAddress,Date\_Joined)

values (8590410239,'Justin', 'Funny','853-983-0925','whysoserious@joker.com','Bruce Wayne Castle,Gotham',to\_date('7-21-05','MM-DD-YY'));

insert into Event (Event\_ID,Event\_Name,EDate,EDescription,Department\_ID)

values (7642801572,'Chase Your Dream',to\_date('5-21-06','MM-DD-YY'),'Chase your dream while help other students chasing theirs',0000000001);

insert into Event (Event\_ID,Event\_Name,EDate,EDescription,Department\_ID)

values (7895310341,'Read More Books',to\_date('5-11-05','MM-DD-YY'),'Donate your books to those who cannot afford', 0000000002);

insert into Event (Event\_ID,Event\_Name,EDate,EDescription,Department\_ID)

values (5638910254,'Give me your money',to\_date('6-21-05','MM-DD-YY'),'This is not a robery. Actually it is',0000000003);

insert into Event (Event\_ID,Event\_Name,EDate,EDescription,Department\_ID)

values (8509431579,'Jane is the Best Teacher',to\_date('5-10-15','MM-DD-YY'),'Jane is the Best teacher. Say yes she will give an A', 0000000004);

insert into Event (Event\_ID,Event\_Name,EDate,EDescription,Department\_ID)

values (7530912808,'Hello World',to\_date('5-21-16','MM-DD-YY'),'Help students to go out and see the world',0000000005);

insert into Donor(Donor\_ID,Donor\_FName,Donor\_LName,DPhone,DEmail,DAddress,Type)

values (0920127916,'Victor', 'Oladipo','362-837-9327','vicoladi@gamil.com','Indianapolis','Money Donor');

insert into Donor(Donor\_ID,Donor\_FName,Donor\_LName,DPhone,DEmail,DAddress,Type)

values (0125829572,'Darron', 'James','812-333-3333','dajmejr@hotmail.com','Indianapolis','Book Donor');

insert into Donor(Donor\_ID,Donor\_FName,Donor\_LName,DPhone,DEmail,DAddress,Type)

values (9140074142,'Lucy', 'Liu','812-666-6666','821008518@qq.com','LA','Money Donor');

insert into Donor(Donor\_ID,Donor\_FName,Donor\_LName,DPhone,DEmail,DAddress,Type)

values (2803758915,'Jane', 'Tan','812-339-6434','janetan@iu.edu','CA','Book Donor');

insert into Donor(Donor\_ID,Donor\_FName,Donor\_LName,DPhone,DEmail,DAddress,Type)

values (5215750125,'Emma', 'Watson','812-345-5678','emwat52@gamil.com','Bloomington','Money Donor');

insert into Donor(Donor\_ID,Donor\_FName,Donor\_LName,DPhone,DEmail,DAddress,Type)

values (5245650125,'Jiawei', 'Li','812-955-5632','jiawli@iu.edu','Bloomington','Both');

insert into Donor(Donor\_ID,Donor\_FName,Donor\_LName,DPhone,DEmail,DAddress,Type)

values (5290850158,'Tom', 'Simpson','568-323-3548','tomsimp@gamil.com','Sebastopol','Both');

insert into MoneyDonated (Donor\_ID,Event\_ID,AmountOfMoney,MDonateDate)

Values (0920127916,7642801572,100000,to\_date('5-21-16','MM-DD-YY'));

insert into MoneyDonated (Donor\_ID,Event\_ID,AmountOfMoney,MDonateDate)

values (9140074142,5638910254,49,to\_date('5-21-15','MM-DD-YY'));

insert into MoneyDonated (Donor\_ID,Event\_ID,AmountOfMoney,MDonateDate)

values (2803758915,8509431579,200,to\_date('5-21-05','MM-DD-YY'));

insert into MoneyDonated (Donor\_ID,Event\_ID,AmountOfMoney,MDonateDate)

values (5290850158,8509431579,500,to\_date('5-15-05','MM-DD-YY'));

insert into MoneyDonated (Donor\_ID,Event\_ID,AmountOfMoney,MDonateDate)

values (5245650125,8509431579,875,to\_date('5-16-05','MM-DD-YY'));

insert into BookDonated (Donor\_ID,Library\_ID,Event\_ID,NumberOfBooks)

values (0125829572,1100000000,7895310341,5);

insert into BookDonated (Donor\_ID,Library\_ID,Event\_ID,NumberOfBooks)

values (5215750125,5500000000,7530912808,4);

insert into BookDonated (Donor\_ID,Library\_ID,Event\_ID,NumberOfBooks)

values (5215750125,6600000000,7530912808,4);

insert into Executers (Executer\_ID,Executer\_Fname,Executer\_Lname,Phone,Email,Department\_ID,President\_ID,Date\_Joined,Role)

values (7642009851,'Pierce', 'Ding','812-349-1020','congding@hotmail.com',0000000004,null,to\_date('9-15-16','MM-DD-YY'),'president');

insert into Executers (Executer\_ID,Executer\_Fname,Executer\_Lname,Phone,Email,Department\_ID,President\_ID,Date\_Joined,Role)

values (8509419011,'Yimei', 'Tong','135-790-256','yimei@iu.edu',0000000001,7642009851,to\_date('3-15-15','MM-DD-YY'),'director');

insert into Executers (Executer\_ID,Executer\_Fname,Executer\_Lname,Phone,Email,Department\_ID,President\_ID,Date\_Joined,Role)

values (0238750094,'Zi', 'Chai','581-923-9305','zichai@hotmail.com',0000000002,7642009851,to\_date('9-15-16','MM-DD-YY'),'VP');

insert into Executers (Executer\_ID,Executer\_Fname,Executer\_Lname,Phone,Email,Department\_ID,President\_ID,Date\_Joined,Role)

values (6749185093,'Amy', 'Zheng','812-349-8931','bmyzheng@gmail.com',0000000003,7642009851,to\_date('9-15-16','MM-DD-YY'),'VP');

insert into Executers (Executer\_ID,Executer\_Fname,Executer\_Lname,Phone,Email,Department\_ID,President\_ID,Date\_Joined,Role)

values (7589103100, 'Huaqiu', 'Zheng','091-053-9835','huaqiuzheng@gmail.com',0000000005,7642009851,to\_date('9-15-16','MM-DD-YY'),'director');

insert into Information (Volunteer\_ID,Program\_ID)

values (7398214390, 1010000000);

insert into Information (Volunteer\_ID,Program\_ID)

values (8590410239, 1010000000);

insert into Information (Volunteer\_ID,Program\_ID)

values (7583901275, 1020000000);

insert into Information (Volunteer\_ID,Program\_ID)

values (7583901275, 1010000000);

insert into Information (Volunteer\_ID,Program\_ID)

values (5901958159, 1030000000);

insert into Information (Volunteer\_ID,Program\_ID)

values (5901958159, 1010000000);

insert into Information (Volunteer\_ID,Program\_ID)

values (5901958159, 1020000000);

insert into DateEstablished (Department\_ID, University\_ID, Date\_Established)

values (1,1000000000,to\_date('5-21-16','MM-DD-YY'));

insert into DateEstablished (Department\_ID, University\_ID, Date\_Established)

values (2,1000000000, to\_date('5-21-16','MM-DD-YY'));

insert into DateEstablished (Department\_ID, University\_ID, Date\_Established)

values (3,1000000000, to\_date('5-21-16','MM-DD-YY'));

insert into DateEstablished (Department\_ID, University\_ID, Date\_Established)

values (4,1000000000, to\_date('5-21-16','MM-DD-YY'));

insert into DateEstablished (Department\_ID, University\_ID, Date\_Established)

values (1,4000000000, to\_date('4-1-05','MM-DD-YY'));

insert into DateEstablished (Department\_ID, University\_ID, Date\_Established)

values (1,3000000000, to\_date('4-10-05','MM-DD-YY'));

insert into DateEstablished (Department\_ID, University\_ID, Date\_Established)

values (2,2000000000, to\_date('6-14-13','MM-DD-YY'));

insert into DateEstablished (Department\_ID, University\_ID, Date\_Established)

values (1,2000000000, to\_date('6-14-17','MM-DD-YY'));

**SQL Queries:**

**Query 1. Show contact information of donors who have donated more than 5 books (Multiple Join, Subquery, Aggregation)**

Select Donor\_ID, Donor\_FName, Donor\_LName, DEmail

from Donor

where Donor\_ID in

(Select Donor\_ID

from BookDonated

Group by Donor\_ID

Having Sum(NumberOfBooks) >= 5);

## Query 1: Big Book Donors

|  |  |  |  |
| --- | --- | --- | --- |
| **DONOR\_ID** | **DONOR\_FNAME** | **DONOR\_LNAME** | **DEMAIL** |
| 125829572 | Darron | James | dajmejr@hotmail.com |
| 5215750125 | Emma | Watson | emwat52@gamil.com |

**Query 2. Show information of libraries which were built in 1998 (Multiple Join, Subquery, Date, is null, REGEXP)**

select l.Library\_ID, Library\_Name, Sum(NumberofBooks), LAddress, Date\_Established

From Library l, BookDonated bd

Where Date\_Established is not null AND l.Library\_ID in

(Select Library\_ID

from Library

where to\_date(Date\_Established, 'DD-MON-YY') like '%98')

Group by l.Library\_ID, Library\_Name, LAddress, Date\_Established;

## Query 2: Libraries Built in 1998

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **LIBRARY\_ID** | **LIBRARY\_NAME** | **SUM(NUMBEROFBOOKS)** | **LADDRESS** | **DATE\_ESTABLISHED** |
| 2200000000 | BCD | 30 | 1251 US Highway, Greenwood, IN 4614 | 20-MAY-98 |
| 1100000000 | ABC | 30 | 155 S, Greenwood, IN 46142 | 20-MAY-98 |
| 3300000000 | CDE | 30 | 2687 E, Plainfield, IN 46168 | 24-JUN-98 |

**Query 3. Show the event name and event id of events which has raised money for more than $500 (Aggregation, Subquery, Group By, Having)**

select Event\_Name, Event\_ID

from Event

where Event\_ID in (select Event\_ID from MoneyDonated Group by Event\_ID Having sum(AmountOfMoney) > 500);

## Query 3: Big Events

|  |  |
| --- | --- |
| **EVENT\_NAME** | **EVENT\_ID** |
| Chase Your Dream | 7642801572 |
| Jane is the Best Teacher | 8509431579 |

**Query 4. How many Money does DreamCorps rise in each event. (Aggregation, Group By)**

select Event\_ID, sum(AmountOfMoney), MDonateDate

From MoneyDonated

Group by Event\_ID, MDonateDate;

## Query 4: MoneyDonated - Event

|  |  |  |
| --- | --- | --- |
| **EVENT\_ID** | **SUM(AMOUNTOFMONEY)** | **MDONATEDATE** |
| 5638910254 | 49 | 21-MAY-15 |
| 7642801572 | 100000 | 21-MAY-16 |
| 8509431579 | 500 | 15-MAY-05 |
| 8509431579 | 875 | 16-MAY-05 |
| 8509431579 | 200 | 21-MAY-05 |

**Query 5. Show how many donors is for each donor type (Group By with Having, Is Null, Order By)**

Select Type, Count(Donor\_ID)

From Donor

Group by Type

Having Type is not null

Order by Type;

## Query 5: Donor Types

|  |  |
| --- | --- |
| **TYPE** | **COUNT(DONOR\_ID)** |
| Book Donor | 2 |
| Both | 2 |
| Money Donor | 3 |

**Query 6. Show Donor ID and total amount of money that the donors from Bloomington has contributed more than $100. (Multiple Join, REGEXP)**

select d.Donor\_ID, Donor\_FName, Donor\_LName, DEmail, DPhone, AmountOfMoney

from Donor d, MoneyDonated m

where DPhone like '812%' AND d.Donor\_ID = m.Donor\_ID AND AmountOfMoney >= 100;

## Query 6: Bloomington Big Money Donors

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **DONOR\_ID** | **DONOR\_FNAME** | **DONOR\_LNAME** | **DEMAIL** | **DPHONE** | **AMOUNTOFMONEY** |
| 2803758915 | Jane | Tan | janetan@iu.edu | 812-339-6434 | 200 |
| 5245650125 | Jiawei | Li | jiawli@iu.edu | 812-955-5632 | 875 |

**Query 7. Show programs id and name of program which has more than 2 volunteer.(Multiple Join, Group By, Aggregation)**

Select i.Program\_ID, Program\_Name, Count(Volunteer\_ID)

From Information i, Program p

where i.Program\_ID = p.Program\_ID

Group by i.Program\_ID, Program\_Name

Having count(Volunteer\_ID) >= 2

Order by Program\_Name;

## Query 7: Big Programs

|  |  |  |
| --- | --- | --- |
| **PROGRAM\_ID** | **PROGRAM\_NAME** | **COUNT(VOLUNTEER\_ID)** |
| 1020000000 | Little 4 | 2 |
| 1010000000 | Little 5 | 4 |

**Query 8. How many books are there in Qinghua Library. (Multiple Join, REGEXP)**

Select Library\_Name, NumberOfBooks

From Library l, BookDonated b

Where l.Library\_ID = b.Library\_ID AND l.Library\_Name = 'Qinghua';

## Query 8: Qinghua Library

|  |  |
| --- | --- |
| **LIBRARY\_NAME** | **NUMBEROFBOOKS** |
| Qinghua | 4 |

**Query 9. Show Executer ID and Executer name of all executers in finance department. (Multiple Join, In/Not In, REGEXP)**

Select Executer\_ID, Executer\_FName, Executer\_LName, Role

From Executers

Where Department\_ID in (select Department\_ID from Department where Department\_name = 'Finance');

## Query 9: Finance Executors

|  |  |  |  |
| --- | --- | --- | --- |
| **EXECUTER\_ID** | **EXECUTER\_FNAME** | **EXECUTER\_LNAME** | **ROLE** |
| 6749185093 | Amy | Zheng | VP |

**Query 10. Show how many departments are set up in each university(Multiple Join, Group By, Aggregation, )**

select de.University\_ID, University\_Name, Count(de.Department\_ID)

from DateEstablished de, University u

where de.University\_ID = u.University\_ID

Group by de.University\_ID, University\_Name

Order by de.University\_ID;

## Query 10: Department Number

|  |  |  |
| --- | --- | --- |
| **UNIVERSITY\_ID** | **UNIVERSITY\_NAME** | **COUNT(DE.DEPARTMENT\_ID)** |
| 1000000000 | Indiana University Bloomington | 4 |
| 2000000000 | University of Texas at Austin | 2 |
| 3000000000 | Michigan State University | 1 |
| 4000000000 | Carnegie Mellon University | 1 |

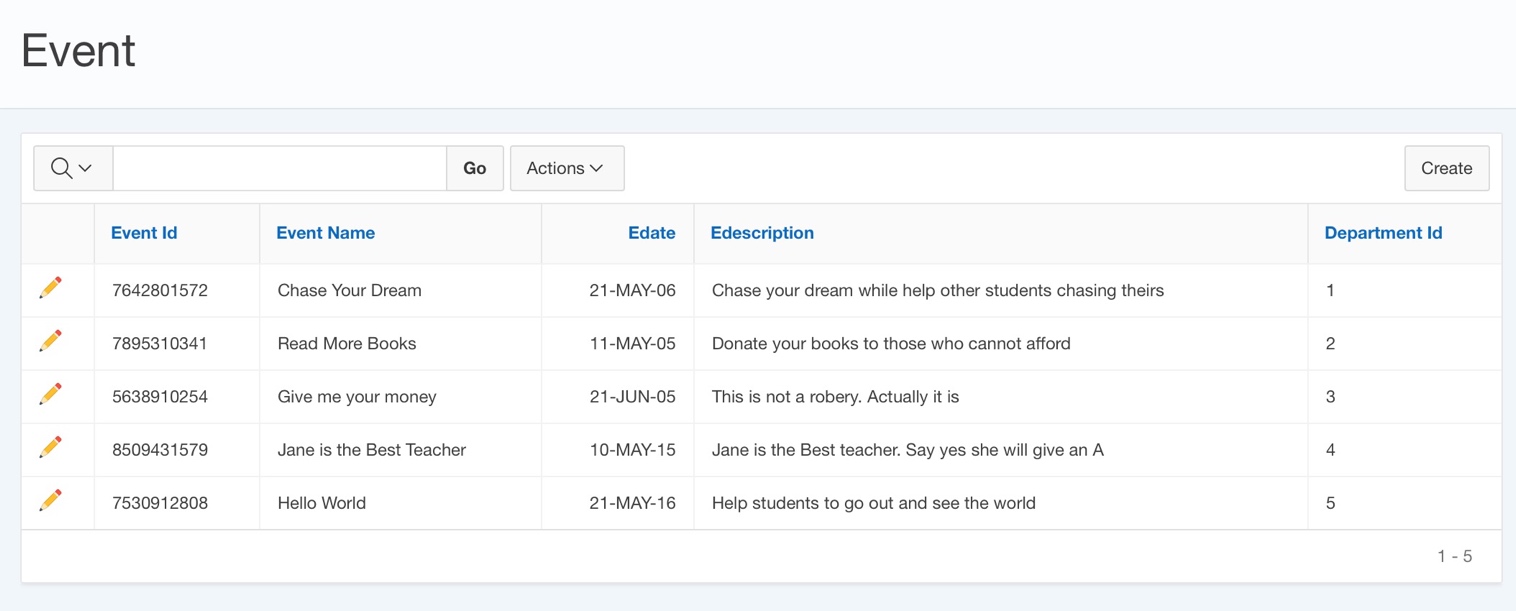
# 4 User Interface

The user interface is built for recording and updating the information for Dream Corps organization, which make easier for the executors and finance managers to keep track of the operating management for every university.

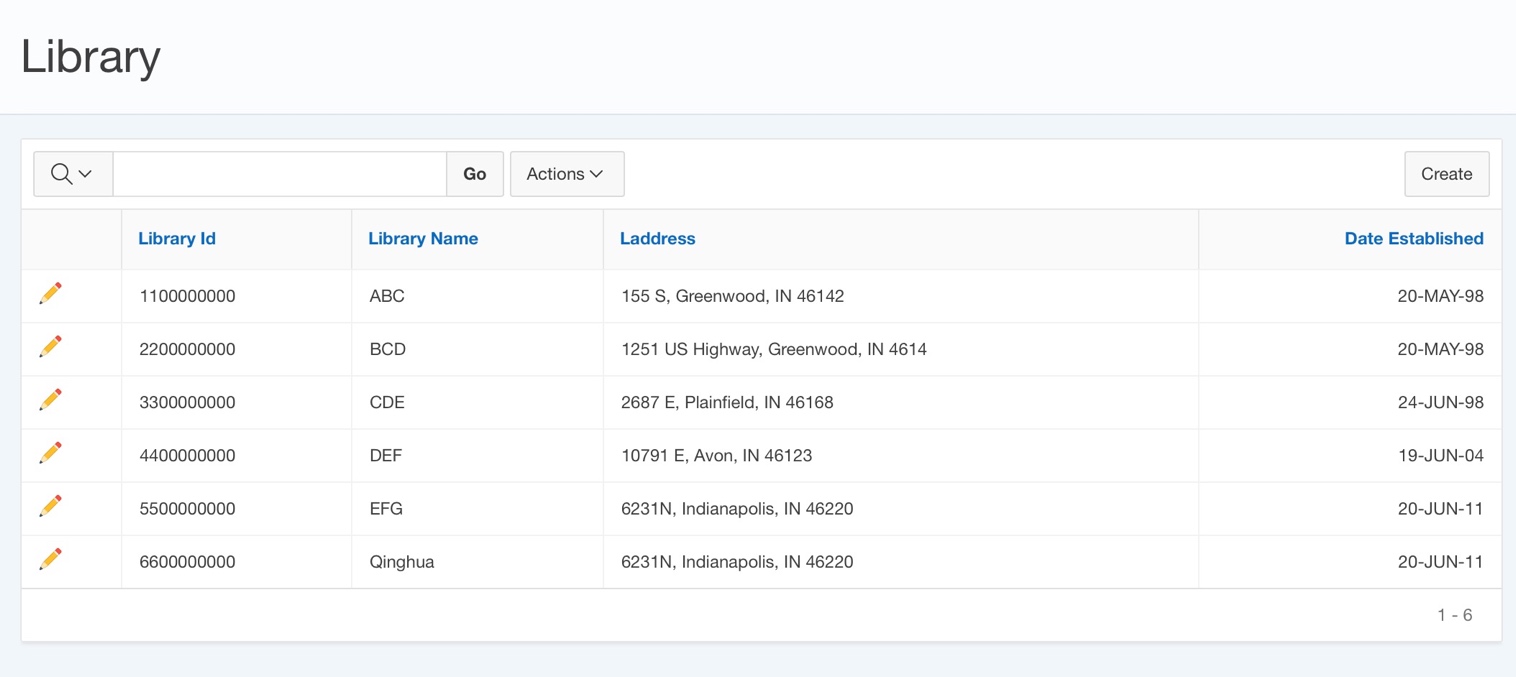
## 4.1 Design

Our targeted users are executors and finance managers. By creating this interface for our end users, they can easily record and manage the whole database. Also, this system can help them to solve managerial questions. Executors can use this interface to input or look up board members’ information, finance managers can use this interface to record the book donation and money donation and so on.

## 4.2 Implementation

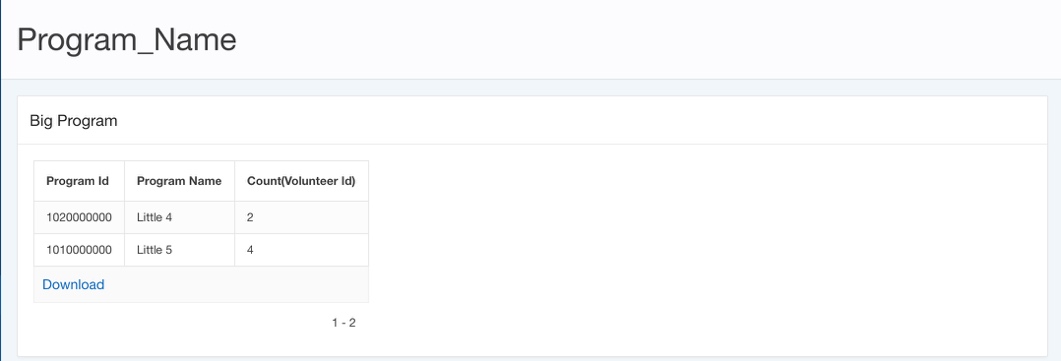
Basic Reports:

The executors can view the event list and the detail of each event through viewing this table.



This table lists all the libraries that donated by Dream Crops., it can help executors make better donation decisions when they consider to build new libraries

* Advanced Report:



Select i.Program\_ID, Program\_Name, Count(Volunteer\_ID)

From Information i, Program p

where i.Program\_ID=p.Program\_ID

Group by i.Program\_ID, Program\_Name

Having count(Volunteer\_ID) >= 2

Order by Program\_Name;

This report tells the users what are the programs IDs and program name that have more than 2 volunteers. First we select program ID, program name and counted volunteer ID from Information and program table, which we use program id to link these two table, and then we grouped by program ID and Program name to show the final results.

# 5 Conclusion

For this project, our team had designed and developed a database for Dream Corp. from conceptual stage, to logical design, to physical design, finally, to implementation. Dream Corps. is a non-profitable organization that raise donations (both money and books) over the world, mainly in mainland China and U.S. for students in rural area in China who cannot afford their tuitions or don’t even have opportunities to go to schools. In this case, the corporation really want to know in details about its programs and donations. With proper data management, it will have better understanding of its current operations and how effective its programs are to raise money/books. Most importantly, board members of the Dream Corp. really want to know that if the money they raised had been placed in the places they want to place. We hope with our designs, the corporation will not only have better corporative level management but also better university level management with their programs, events, volunteers as well as donations.

We did make plenty of mistakes when we were working on this project. Luckily, we were able to identify our problems and took proper actions to address the issues. When we were drawing the relationship, we couldn’t find a good way to connect books in the library and the book donations. We originally create an entity for Book which is related to Library and we connected Library to Donations. However, this is quite confusing. Besides, managing books in Library is beyond Dream Corp.’s responsibility. Thus, in our final version of ERD, we decided to take off Book entity. The even harder part was creating interfaces. We had to really think about what interfaces would be helpful to refine our system. We created interfaces about departments, money/book donations. But after some severe discussions, we decided to re-design our interfaces because those interfaces were not necessarily the most important information that the board members want to know. Therefore, we re-designed our interfaces based on location, library, programs and volunteer information etc. Then we have included most crucial information about the daily operation and the board members can find any other information by just simply change the condition settings.

By working on this project, we have truly benefited a lot from it. Conceptually, now we understand the aspects we need to consider when managing an organization’s data and we now know where to start when dealing with this kind of database management problem. Practically, this project definitely helped our coding skills and allowed us to have better understanding of SQL language. Unlike class practices, which all have the right answers, in this project, we had to design our own codes and give us the answer we want. So we had to really push ourselves with it and tried every possible code we learned to give us a desired outcome. Overall, we all enjoyed this project and the class very much.