Title: Atlas Corporate Human Resource Data

Objective: Create a Hierarchy between Employee, Company and Services to be used by Employees.

By,

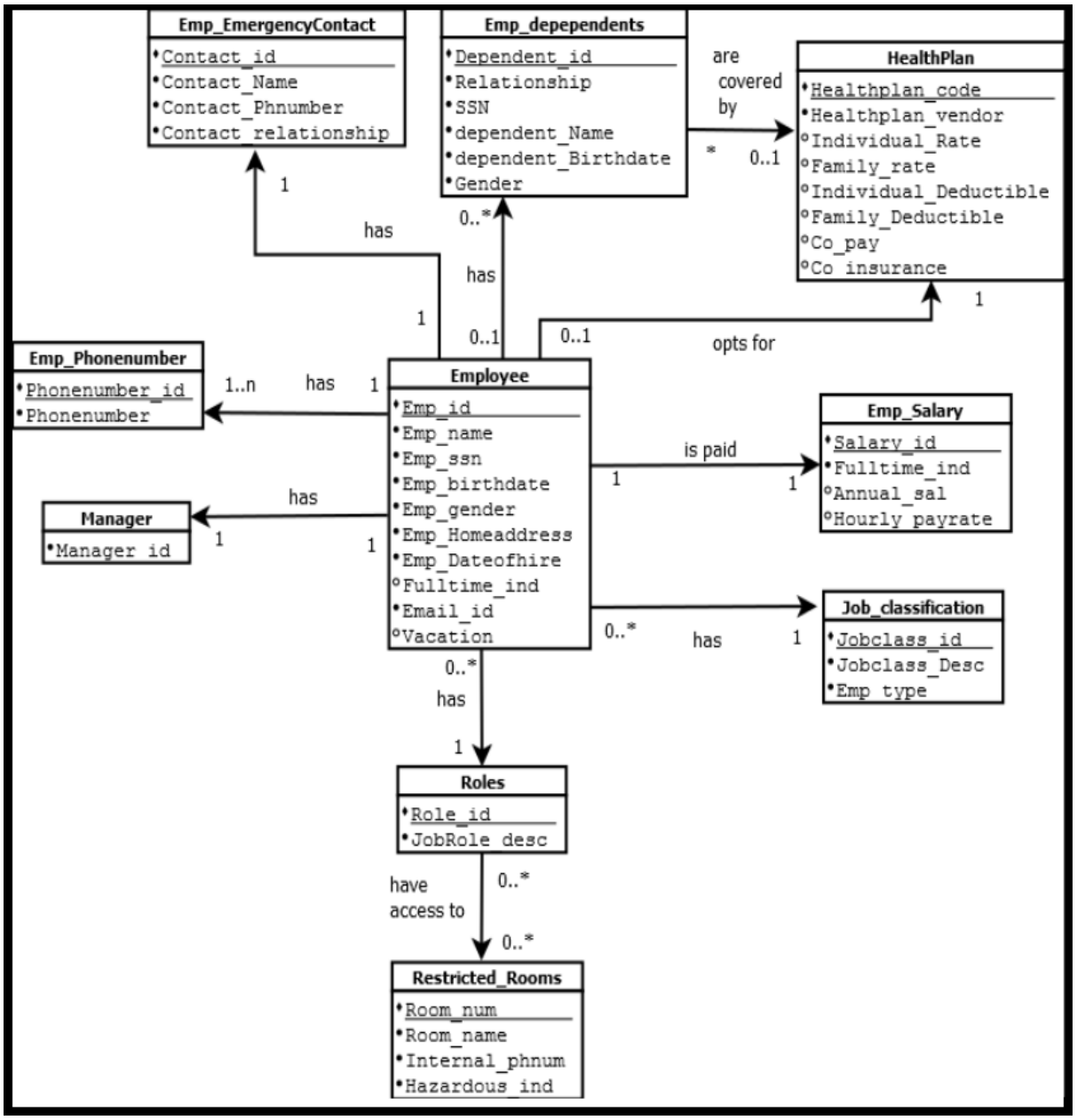
KATUKAM RAJU

Table of Content

1. ER Diagram 3
2. Relationship Database Schema 4
    1. Roles 6
    2. Job_Classification 7
    3. Employee Table 7
    4. Emp_dependents 8
    5. Emp_Phonenumber 8
    6. Emp_EmergencyContact 8
    7. Emp_Hierarchy 9
    8. Emp_Salary 9
    9. HealthPlan 9
    10. Emp_Insurance 10
    11. Insured_dependents 10
    12. Restricted_Rooms 10
    13. Restricted_Access 11
    14. Emp_W4 11
3. Relational Database Implementation 11
    1. Database 11
    2. Domain 11
    3. Table Creation 12
    1. Views 15
4. Table Population 15
5. Database Operational Testing 29

1. ER Diagram

Following is the Entity Relationship diagram for Atlas Corporate HR Data.



As per ER diagram following are the entities

- Employee : Stores personal data of 37 employees , his /her Date of hire, Email id and Vacation earned.
- Emp_Dependents : Employees have Dependents. Employees may have multiple dependents or may not have at all. This entity stores details of Dependents.
- Emp_EmergencyContact: Every employee has one Emergency contact. This entity stores details of that contact.
- Emp_Phonenumber: Employee may have one or more Phonenumber . This entity stores phonenumber for those employees.
- Manager: Every employee except Owner has manager. This entity stores manager id details for all employees.
- Roles: This entity has 4 leadership roles details maintained. Addition to the leadership roles, Roles of other employee is also maintained.
- HealthPlan: This entity stores details of the Health Plan.
- Job Classification: This entity stores details of Job classification in the Atlas Corp. Classification details like Classification Class_id, Exempt or Non exept is maintained.
- Emp_Salary : This entity stores detail of Salary or Hourly rate based on Job classification associated to the employee
- Restricted Rooms: employees have access to Restricted Rooms based on Roles. This entity stores details of the Rooms in the Corporation and Roles who have access on those Rooms.

2. Relationship Database Schema

Following tables gives details on how entities are converted into Relational Schema. Most Relationship from ER Diagram are converted into tables storing the Foreign key from tables.

Sno.	Table Name	Details
1	Roles	This table stores 4 leadership Roles for Atlas Corporation HR Data like the owner, the human resource director, the information systems director, and the operations director. Apart from that it also stores other Roles for other Employees.

2	Job_Classification	This tables stores the information on Job Classification like • Exempt /Non Exempt • Jobclass_id There are no job classifications which are held by both exempt and non-exempt employees. This information is required in Salary table to identify exempt and non-exempt employee and store information like Fulltime/Part-time for Non-exempt employee.
3	Employee	Stores personal data of 37 employees and his /her Date of hire, • it also stores Foreign key to Role id referencing Role, • Job classification id referencing Job Classification • This table also stores Email id of Few employees who have some specific Job class id • It stores calculated value of Vacation in the table.
4	Emp_Dependents	Stores information on employee's dependents. Emp_id is the foreign key referencing the Employee table to identify which dependent is associated to which employee.
5	Emp_Phonenumber	Stores employees Phone number. Since one employee can have multiple phone numbers, it has that many records for an employee.
6	Emp_emergencyContact	Stores Phone number, name of Emergency contact and Relationship to employee.
7	Emp_hierarchy	Every employee has manager so this table stores details of Employee's and manager's Emp_id.
8	Emp_Salary	Stores information on Salary , Jobclass_id for an employee
9	HealthPlan	Stores details of 3 health plans provided by the Company
10	Emp_insurance	Stores information on which plan is opted by which employee and if they have covered their dependents or not.

		Non Exempt who are part-time Employees are not allowed to avail the benefit so there entry is not in the table.
11	Insured_Dependents	Stores details on dependent insured /covered as part of the HealthPlan. Employee in Emp_insurance having Covereddependent_ind indicator 1 has details in this table.
12	Restricted_Rooms	Stores information on the Restricted Room and Room_num available in Company. It also stores information on which Role has access to which Room. Role_id is a foreign key referencing Roles table
13	Restricted_Access	This table stores information on which employee has access to which room. • Employee having a specified Role is having access to the Room. • Supervisor of the employee is also allowed access. So record regarding the supervisor having access to Room is also stored in the table.

Following are the list of view used in the model

Sno	View Name	Details
1	Emp_w4	• This is View to select the number of dependents of the Employee used for W4 tax information • It is calculated as count of dependents for every employee from Emp_dependents table

1. Roles

This table stores 4 leadership Roles for Atlas Corporation HR Data like the owner, the human resource director, the information systems director, and the operations director.

Apart from that it also stores other Roles for other Employees.

Roles		
Role_id	Numeric(4)	Primary key
JobRole_desc	Varchar(40)	Not null

2. Job_Classification

This tables stores the information on Job Classification like

- Exempt /Non Exempt
- Jobclass_id

There are no job classifications which are held by both exempt and non-exempt employees.

This information is required in Salary table to identify exempt and non exempt employee on which job classication is for Non exempt employee.

Job_Classification		
Jobclass_id	Varchar(3)	Primary key
Jobclass_Desc	Varchar(15)	Not null
Emp_type	Varchar(10)	Not null

3. Employee Table

Stores personal data of 37 employees and his /her Date of hire.

Employee		
Emp_id	Numeric(4)	Primary key
Emp_name	Varchar(40)	Not null
Emp_ssn	Varchar(11)	Not null
Emp_birthdate	Date	Not null
Emp_gender	Gender_type	Not null
Emp_Homeaddress	Varchar(100)	Not null
Emp_DateofHire	Date	Not null
Role_id	Varchar(5)	Foreign key to Roles.Role_id

Jobclass_id	Varchar(3)	Foreign key to JobClassification.Jobclass_id
Fulltime_ind	Numeric(1)	
Email_id	Varchar(50)	
Vacation	Numeric(2)	

Gender_type is user defined domain.

Fulltime_ind stores either 0 or 1 (1 -the employee is Full time employee , And 0- Not a full time employee).

4. Emp_dependents

Stores information on employee's dependents. Emp_id is the foreign key referencing the Employee table to identify which dependent is associated to which employee.

Emp_dependents		
Dependent_id	Varchar(4)	Primary key
Emp_id	Numeric(4)	Foreign key
Relationship	Relationship_type	Not null
SSN	Varchar(11)	
Name	Varchar(40)	Not null
Birthdate	Date	Not null
Gender	Gender_type	Not null

Gender_Type and Relationship_type are user defined domain.

5. Emp_Phonenumber

Stores employees Phone number. Since one employee can have multiple phone numbers, this table has that many records for an employee.

Emp_Phonenumber		
Phonenumber_id	Varchar(4)	Primary key
Emp_id	numeric(4)	Foreign key
Phonenumber	Numeric(10)	Not null

6. Emp_EmergencyContact

Stores Phone number, name of Emergency contact and Relationship to employee

Emp_EmergencyContact		
Contact_id	Varchar(6)	Primary Key
Emp_id	Numeric(4)	Foreign key
Contact_name	Varchar(30)	Not null
Contact_phnumber	Numeric(10)	Not null
Contact_relationship	Relationship_type	Not null

7. Emp_Hierarchy

Stores details of Employee and manager's Emp_id.

Emp_hierarchy		
Emp_id	Numeric(4)	Foreign key
Manager_id	Numeric(4)	

8. Emp_Salary

Stores information on Salary , Jobclass_id and Fulltime/Partime indicator for an employee

Emp_Salary		
Salary_id	Varchar(4)	Primary key
Emp_id	Numeric(4)	Foreign key
Fulltime_ind	numeric(1)	
Annual_sal	Numeric(9)	
Hourly_rate	Numeric(4)	

9. HealthPlan

Stores details of 3 health plans provided by the Company

HealthPlan		
Healthplan_code	Varchar(8)	Primary key
Healthplan_vendor	Varchar(10)	Not null
Individual_rate	Numeric(4)	
Family_rate	Numeric(4)	
Individual_deductibles	Numeric(4)	
Family_deductibles	Numeric(4)	
Co_pay	Numeric(2)	
Co_insurance	Numeric(2)	

10.Emp_Insurance

Stores information on which plan is opted by which employee and if they have covered their dependents

Non Exempt- part-time Employees are not allowed to avail the benefit so their entry is not in this table.

Emp_insurance		
Emp_id	Numeric(4)	Foreign key
Healthplan_code	Varchar(8)	Foreign key
Covereddependent_ind	Numeric(1)	Not null

Covereddependent_ind stores either 0 or 1 (1 -the employee has dependents covered in health insurance plan , And 0- dependents not covered).

11. Insured_dependents

Stores details on dependent insured. Employee in Emp_insurance having Covereddependent_ind indicator 1 has details in this table.

Insured_dependents		
Emp_id	Numeric(4)	Foreign key
Dependent_id	Varchar(4)	Foreign key

12. Restricted_Rooms

Stores information on the Restricted Room and Room_num available in Company . It also stores information on which Role has access to the Room.

Role_id is a foreign key referencing Roles table

Insured_dependents		
Room_num	Varchar(4)	Primary key
Room_name	Varchar(30)	
Internal_phnum	Numeric(10)	
Hazardous_ind	Numeric(1)	
Role_id	Varchar(5)	Foreign key

13. Restricted_Access

This table stores information on which employee has access to which room. Employee id having a specified Role is having access to the Room. Supervisor of the employee is also allowed access. So record regarding the supervisor having access to Room is also stored in the table.

Restricted_Access		
Emp_id	Numeric(4)	Foreign key
Room_num	Varchar(4)	Foreign key

14. Emp_W4

This is a view storing information on number of dependents reported by each employees.

3. Relational Database Implementation

1. Database

Below Script will create Database

```
-- Create Database

CREATE DATABASE AtlasCorpHRData;
```

2. Domain

Here, 2 User defined Domains are created

```
-- Create Domain
```

```
CREATE domain Gender_type as CHAR(1)  CHECK (Value in ( 'M', 'F'));

CREATE domain  Relationship_type as CHAR(8) CHECK (Value in ( 'Spouse', 'Children',
'Friend', 'Parent'));
```

3. Table Creation

Following Scripts Create Tables in Database . 4 queries from the Create script has been added here.

```
---Script to create tables on database
-- Create table Roles

create table Roles
(
role_id varchar(5) primary key,
JobRole_desc varchar(40) not null
);


-- Create tableJobclassification

Create table Jobclassification
( jobclass_id varchar(3) ,
  jobclass_Desc varchar(15) not null,
  Emp_typ varchar(10) not null,
  primary  key (jobclass_Id)
  );


-- Create table Employee

Create table Employee
( Emp_id numeric(4) ,
  Emp_name varchar(40) not null,
  Emp_ssn varchar(11) not null,
  Emp_birthdate date not null,
  Emp_gender gender_type ,
  Emp_Homeaddress varchar(100),
  Emp_dateofhire date not null,
  role_id varchar(5) ,
  jobclass_id varchar(3),
  Fulltime_ind numeric(1) ,
  Email_id varchar(50),
  Vacation numeric(2),
  primary key ( Emp_id),
  foreign key (role_id ) references Roles(Role_id),
```

```
  foreign key (jobclass_id) references jobclassification(jobclass_id)  );


-- Create table Emp_Dependents

create table Emp_Dependents
( dependent_id varchar(4),
  emp_id numeric(4) not null,
  Relationship relationship_type not null,
  ssn varchar(11),
  dependent_name varchar(40) not null,
  dependent_birthdate date not null,
  Gender gender_type not null,
  primary key(dependent_id),
  foreign key (emp_id) references Employee( emp_id)
  );

-- Create table Emp_Phonenumber

Create table Emp_Phonenumber
( Phonenumber_id varchar(4),
  Emp_id numeric(4) ,
  Phonenumber numeric(10) ,
  primary key ( Phonenumber_id ),
  foreign key (Emp_id) references Employee(emp_id)
  );


-- Create table Emp_emergencycontact

create table Emp_emergencycontact
( contact_id varchar(6)  ,
  emp_id numeric(4),
  Contact_name varchar(30) not null ,
  Çontact_phonenumber numeric(10) not null,
  Contact_relationship RELATIONSHIP_TYPE not null,
  primary key (contact_id),
  foreign key (emp_id) references Employee(emp_id)
  );



-- Create table Emp_hierarchy

Create table Emp_hierarchy
(Emp_id numeric(4) ,
 Manager_id    numeric(4)  ,
```

```
 primary key (Emp_id),
 foreign key ( Emp_id) references Employee( Emp_id)
 );


-- Create table Emp_Salary

create table Emp_Salary
( Salary_id varchar(4) ,
  Emp_id numeric(4) ,
  Annual_Sal numeric(9),
  hourly_payrate numeric(4),
  primary key (Salary_id),
  foreign key (Emp_id) references Employee (Emp_id)
  );

-- Create table HealthPlan

 Create table HealthPlan
( Healthplan_code varchar(8)  ,
  Healthplan_vendor varchar(10) not null,
  Individual_rate numeric(4),
  Family_rate numeric(4),
  Individual_deductibles numeric(4),
  Family_Deductibles numeric(4),
  Co_pay numeric(2),
  Co_insurance numeric(2),
  primary key (Healthplan_code)
  );


  -- Create table Emp_Insurance

  Create table Emp_insurance
( Emp_id numeric(4) ,
  Healthplan_code varchar(8) ,
  Covereddependents_ind numeric(1) not null,
  primary key (Emp_id),
  foreign key (Emp_id) references Employee( Emp_id),
  foreign key (Healthplan_code) references Healthplan (Healthplan_code)
  );

  -- Create table Insured_Dependents

  Create table Insured_Dependents
( Emp_id numeric(4) ,
  Dependent_id varchar(4) ,
```

```
 primary key (dependent_id),
 foreign key (dependent_id) references Emp_dependents (dependent_id),
 foreign key (Emp_id) references Emp_insurance ( emp_id)
 );

 -- Create table Restricted_Room

 Create table Restricted_Room
( Room_num varchar(4) ,
 Room_name varchar(30) not null,
 Internal_Phnum numeric(10) not null,
 Hazardous_ind numeric(1) not null,
 Role_id varchar(5) ,
 primary key (room_num),
 foreign key (Role_id) references Roles (Role_id)
 );

 -- Create table Restricted_Access

 create table Restricted_Access
(Emp_id numeric(4) ,
 Room_num varchar(4) ,
 primary key (emp_id, Room_num),
 foreign key (Emp_id) references Employee (Emp_id),
 foreign key (Room_num) references Restricted_Room (Room_num)
 );
```

1. Views

Following Script creates Views

```
-- create view Emp_w4

Create view Emp_W4 as
select Emp_id as Emp_id, count(dependent_id) Count_of_dependents from
emp_dependents
 group by emp_id ;
```

4. Table Population

Following Script is used for Table Population

```
-- Populate Roles tables
```

```
insert into Roles (Role_id, JobRole_Desc)
Values
('R0001','Owner'),
('R0002','Human Resource Director'),
('R0003','Information System Director'),
('R0004','Operations Director'),
('R0005','Project Lead'),
('R0006','Team member'),
('R0007','Special HR Role'),
('R0008','Senior Analyst'),
('R0009','Server Room Technician ');

-- Populate Rows in Jobclassification

insert into Jobclassification ( jobclass_id, jobclass_desc, emp_typ)
values
('E1','Executive','Exempt'),
('E2','Managers','Exempt'),
('E3','Leaders','Exempt'),
('E4','Skilled Workers','Exempt'),
('NE1','Executive','Non Exempt'),
('NE2','Managers','Non Exempt'),
('NE3','Leaders','Non Exempt '),
('NE4','Skilled Workers','Non Exempt');


 -- Insert into table Employee for 37 employees
```

Insert into Employee (Emp_id, Emp_name, Emp_ssn, Emp_birthdate, Emp_gender, Emp_Homeaddress, Emp_dateofhire,Role_id, Jobclass_id,Fulltime_ind,Email_id)

Values (1001,'Mark Zukerberg','550-24-3090','12-may-1984','M','290 turnpike road, MA','01-mar-2004','R0001','E1',null,'Owner@atlas.com'),

(1002,'Dhanush Pulyala','599-60-4301','20-jun-1980','M','100 London street, CT','30-mar-2004','R0002','E2',null,'Hrdirector@atlas.com'),

(1003,'Prasad Rao','550-24-3091','12-jan-1984','M','3 Havard Road, MA','10-mar-2004','R0003','E2',null,'InformationDirector@atlas.com'),

(1004,'Bill Gates','599-60-4302','14-dec-1977','M','Beverly Hillls, California','10-mar-2004','R0004','NE2',1,null),

(1005,'Steve Jobs','550-24-3092','14-dec-1958','M','290 turnpike road, MA','10-mar-2004','R0005','NE3',0,null),

(1006,'John Smith','599-60-4303','14-dec-1959','M','100 London street, CT','10-mar-2004','R0005','E3',null,null),

(1007,'Eudardo Saverin','550-24-3093','14-dec-1960','M','3 Havard Road, MA','31-Dec-2004','R0006','E4',null,null),

(1008,'Sean Parker','599-60-4304','14-dec-1961','M','Beverly Hillls, California','31-Dec-2004','R0006','E4',null,null),

(1009,'Chris Hughes','550-24-3094','14-dec-1962','M','290 turnpike road, MA','31-Dec-2004','R0006','NE4',1,null),

(1010,'Andrew McCollum','599-60-4305','14-dec-1963','M','100 London street, CT','31-Dec-2004','R0006','NE4',0,null),

(1011,'Gil Amelio','550-24-3095','14-dec-1964','M','3 Havard Road, MA','31-Dec-2004','R0007','E4',null,null),

(1012,'Warren Buffet','599-60-4306','14-dec-1965','M','Beverly Hillls, California','31-Dec-2004','R0006','E4',null,null),

(1013,'Indira Gandhi ','550-24-3096','14-dec-1966','F','290 turnpike road, MA','31-Dec-2004','R0005','E3',null,null),

(1014,'Hillary Clinton','599-60-4307','14-dec-1967','F','100 London street, CT','31-Dec-2004','R0005','NE3',1,null),

(1015,'Angela Markell','550-24-3097','14-dec-1968','F','3 Havard Road, MA','31-Dec-2004','R0006','NE4',0,null),

(1016,'Ivanka Trump','599-60-4308','14-dec-1969','F','Beverly Hillls, California','31-Dec-2004','R0006','E4',null,null),

(1017,'Chandra Kochhar','550-24-3098','14-dec-1970','F','290 turnpike road, MA','15-mar-2005','R0006','E4',null,null),

(1018,'Vijaylakshmi Iyer','599-60-4309','14-dec-1971','F','100 London street, CT','15-mar-2005','R0006','E4',null,null),

(1019,'Pratibha Patil','550-24-3099','14-dec-1972','F','3 Havard Road, MA','15-mar-2005','R0006','NE4',1,null),

(1020,'Indra Nooyi ','599-60-4310','14-dec-1973','F','Beverly Hillls, California','15-mar-2005','R0008','NE4',0,null),

(1021,'Tim Cook','550-24-3100','14-dec-1974','M','290 turnpike road, MA','15-mar-2005','R0005','E3',null,null),

(1022,'Cameron Winklevoss','599-60-4311','14-dec-1975','M','100 London street, CT','15-mar-2005','R0005','E3',null,null),

(1023,'Tyler Winklevoss','550-24-3101','14-dec-1976','M','3 Havard Road, MA','15-mar-2005','R0005','E3',null,null),

(1024,'Divya Narendra','599-60-4312','14-dec-1977','M','Beverly Hillls, California','15-mar-2005','R0006','NE4',1,null),

(1025,'Steve Woznaik','550-24-3102','14-dec-1978','M','290 turnpike road, MA','15-mar-2005','R0006','NE4',0,null),

(1026,'Ronald Wayne','599-60-4313','14-dec-1979','M','100 London street, CT','20-june-2005','R0006','E4',null,null),

(1027,'Mike Markulla','550-24-3103','14-dec-1980','M','3 Havard Road, MA','20-june-2005','R0006','E4',null,null),

(1028,'John Sculley ','599-60-4314','14-dec-1981','M','Beverly Hillls, California','20-june-2005','R0006','E4',null,null),

(1029,'Michael Spindler','550-24-3104','14-dec-1982','M','290 turnpike road, MA','20-june-2005','R0006','NE4',1,null),

(1030,'Venus Williams','599-60-4315','14-dec-1983','F','100 London street, CT','20-june-2008','R0006','NE4',0,null),

(1031,'Maria Sharapova','550-24-3105','14-dec-1984','F','3 Havard Road, MA','20-june-2009','R0006','E4',null,null),

(1032,'Serena Williams','599-60-4316','14-dec-1985','F','Beverly Hillls, California','20-june-2010','R0006','E4',null,null),

(1033,'Paul Allen','550-24-3106','14-dec-1986','M','290 turnpike road, MA','20-june-2011','R0006','E4',null,null),

(1034,'Steve Ballmer','599-60-4317','14-dec-1987','M','100 London street, CT','20-june-2012','R0006','NE4',1,null),

(1035,'Satya Nadella','550-24-3107','14-dec-1988','M','3 Havard Road, MA','20-june-2013','R0006','NE4',0,null),

(1036,'John Thompson','599-60-4318','14-dec-1989','M','Beverly Hillls, California','20-june-2014','R0009','E4',null,null),

(1037,'Brad Smith','550-24-3108','14-dec-1990','M','290 turnpike road, MA','20-june-2015','R0009','E4',null,null);

-- populate Emp_dependents table

insert into Emp_Dependents (dependent_id, emp_id, Relationship, ssn, dependent_name, dependent_birthdate, Gender)

values

('D001','1006','Spouse','800-34-1234','Karen Smith','19-Nov-1984','F'),

('D002','1006','Children','865-23-3000','Nicole Smith','15-Mar-2000','F'),

('D003','1006','Children','800-34-1235','Kevin Smith','25-dec-2001','M'),

('D004','1001','Spouse','865-23-3001','Pricilla Chan','15-Mar-1983','F'),

('D005','1001','Children','800-34-1236','Jack Zukerberg','12-Apr-2015','M'),

('D006','1004','Spouse','865-23-3002','Melinda Gates','27-Oct-1970','F'),

('D007','1005','Children','800-34-1237','Jennifer Hogan','12-Apr-2015','F'),

('D008','1006','Children','865-23-3003','Kathy Ford','12-Apr-2015','F'),

('D009','1007','Spouse','800-34-1238','Rachel Saverin','16-Aug-1984','F'),

('D010','1008','Children','865-23-3004','George Clooney','01-May-2003','M'),

('D011','1009','Spouse','800-34-1239','Diandra Anderson','27-nov-1975','F'),

('D012','1010','Children','865-23-3005','Jenna Carlson','12-Apr-2001','M'),

('D013','1011','Children','800-34-1240','Mike Watson','01-May-2002','M'),

('D014','1012','Children','865-23-3006','Michael Beaton','30-Apr-2003','M'),

('D015','1013','Children','800-34-1241','Shannon Harris','12-mar-2004','F'),

('D016','1014','Children','865-23-3007','Karen Clinton','12-Apr-2005','F'),

('D017','1015','Children','800-34-1242','Donald Lewis','12-Apr-2001','M'),

('D018','1016','Children','865-23-3008','Jeff Goldblum','01-May-2002','M'),

('D019','1017','Children','800-34-1243','Eric Roberts','12-Apr-2003','M'),

('D020','1018','Children','865-23-3009','Pranav Kher','10-mar-2004','M'),

('D021','1019','Children','800-34-1244','Rachel Patil','10-Apr-2005','F'),

('D022','1020','Children','865-23-3010','Samuel Jackson','10-Apr-2001','M'),

('D023','1021','Children','800-34-1245','Jim Morgan','01-May-2002','M'),

('D024','1022','Children','865-23-3011','Alexander Bell','10-Apr-2003','M'),

('D025','1023','Children','800-34-1246','Jessica Shannon','13-mar-2004','F'),

('D026','1024','Children','865-23-3012','Brandon Wright','13-Apr-2005','M'),

('D027','1025','Children','800-34-1247','Matthew Roberts','13-Apr-2001','M'),

('D028','1026','Children','865-23-3013','Joseph Thomas','05-May-2002','M');

-- Populate Emp_Phonenumber

Insert into Emp_Phonenumber (Phonenumber_id, Emp_id, Phonenumber)

values

('P001',1001,9100034590),

('P002',1002,9101034653),

('P003',1003,9102034716),

('P004',1003,9103034779),

('P005',1003,9104034842),

('P006',1004,9105034905),

('P007',1005,9106034968),

('P008',1006,9107035031),

('P009',1007,9108035094),

('P010',1008,9109035157),

('P011',1009,9110035220),

('P012',1010,9111035283),

('P013',1011,9112035346),

('P014',1012,9113035409),

('P015',1013,9114035472),

('P016',1014,9115035535),

('P017',1015,9116035598),

('P018',1016,9117035661),

('P019',1017,9118035724),

('P020',1018,9119035787),

('P021',1019,9120035850),

('P022',1020,9121035913),

('P023',1021,9122035976),

('P024',1022,9123036039),

('P025',1023,9124036102),

('P026',1024,9125036165),

('P027',1025,9126036228),

('P028',1026,9127036291),

('P029',1027,9128036354),

('P030',1028,9129036417),

('P031',1029,9130036480),

('P032',1030,9131036543),

('P033',1031,9132036606),

('P034',1032,9133036669),

('P035',1033,9134036732),

('P036',1034,9135036795),

('P037',1035,9136036858),

('P038',1036,9137036921),

('P039',1037,9138036984);

-- Populate Emp_Emergencycontact

```
Insert into Emp_emergencycontact
values
('E00101',1006,'Michael Smith',8898999992,'Parent'),
('E00102',1005,'Pricilla Chan',9001002301,'Spouse'),
('E00103',1004,'Melinda Gates',9103004610,'Spouse'),
('E00104',1005,'Jennifer Hogan',9205006919,'Children'),
('E00105',1007,'Rachel Saverin',9307009228,'Children'),
('E00106',1008,'George Clooney',9409011537,'Children'),
('E00107',1009,'Diandra Anderson',9511013846,'Children'),
('E00108',1010,'Jenna Carlson',9613016155,'Children'),
('E00109',1011,'Mike Watson',9715018464,'Children'),
('E00110',1012,'Michael Beaton',9817020773,'Children'),
('E00111',1013,'Rajeev Dave',1449057717,'Friend'),
('E00112',1014,'Melinda Gates',9103004610,'Friend'),
('E00113',1015,'Donald Lewis',1123027700,'Children'),
('E00114',1016,'Rajeev Dave',9449057717,'Friend'),
('E00115',1017,'Eric Roberts',9449057718,'Children'),
('E00116',1018,'Pranav Kher',9449057719,'Children'),
('E00117',1019,'Rachel Patil',9449057720,'Children'),
('E00118',1020,'Samuel Jackson',9449057721,'Children'),
('E00119',1021,'Jim Morgan',9449057722,'Children'),
('E00120',1022,'Alexander Bell',9449057723,'Children'),
('E00121',1023,'Jessica Shannon',9449057724,'Children'),
('E00122',1024,'Brandon Wright',9449057725,'Children'),
('E00123',1025,'Matthew Roberts',9449057726,'Children'),
('E00124',1026,'Joseph Thomas',9449057727,'Children'),
('E00125',1003,'Sachin Kumar',9449057728,'Friend'),
('E00126',1002,'Rajeev Dave',9449057729,'Friend'),
('E00127',1027,'Matthew Roberts',9449057730,'Friend'),
```

('E00128',1028,'Jessica Shannon',9449057731,'Friend'),

('E00129',1029,'Brandon Wright',9449057732,'Friend'),

('E00130',1030,'Matthew Roberts',9449057733,'Friend'),

('E00131',1031,'Joseph Thomas',9449057734,'Friend'),

('E00132',1032,'Sachin Kumar',9449057735,'Friend'),

('E00133',1033,'Matthew Roberts',9449057736,'Friend'),

('E00134',1034,'Matthew Roberts',9449057737,'Friend'),

('E00135',1035,'Matthew Roberts',9449057738,'Friend'),

('E00136',1036,'Matthew Roberts',9449057739,'Friend'),

('E00137',1037,'Matthew Roberts',9449057740,'Friend');

-- Populate Emp_hierarchy

Insert into Emp_hierarchy

values

(1001,null),

(1002,1001),

(1003,1001),

(1004,1001),

(1005,1002),

(1006,1002),

(1007,1005),

(1008,1005),

(1009,1005),

(1010,1006),

(1011,1006),

(1012,1006),

(1013,1003),

(1014,1003),

(1015,1013),

(1016,1013),

(1017,1013),

(1018,1014),

(1019,1014),

(1020,1014),

(1021,1004),

(1022,1004),

(1023,1004),

(1024,1021),

(1025,1021),

(1026,1021),

(1027,1021),

(1028,1022),

(1029,1022),

(1030,1022),

(1031,1022),

(1032,1022),

(1033,1023),

(1034,1023),

(1035,1023),

(1036,1023),

(1037,1023);

-- Populate Emp_Salary table

Insert into Emp_Salary (Salary_id, Emp_id, Annual_sal, hourly_payrate)

```
 values('S001',1001,5000000,null),
('S002',1002,4000000,null),
('S003',1003,4000000,null),
('S004',1004,null,200),
('S005',1005,null,150),
('S006',1006,3000000,null),
('S007',1007,250000,null),
('S008',1008,300000,null),
('S009',1009,null,125),
('S010',1010,null,125),
('S011',1011,400000,null),
('S012',1012,250000,null),
('S013',1013,3000000,null),
('S014',1014,null,150),
('S015',1015,null,125),
('S016',1016,250000,null),
('S017',1017,250000,null),
('S018',1018,550000,null),
('S019',1019,null,120),
('S020',1020,null,125),
('S021',1021,3000000,null),
('S022',1022,3000000,null),
('S023',1023,3000000,null),
('S024',1024,null,120),
('S025',1025,null,110),
('S026',1026,250000,null),
('S027',1027,250000,null),
('S028',1028,600000,null),
('S029',1029,null,125),
```

('S030',1030,null,125),

('S031',1031,250000,null),

('S032',1032,250000,null),

('S033',1033,880000,null),

('S034',1034,null,125),

('S035',1035,null,130),

('S036',1036,250000,null),

('S037',1037,250000,null);

-- Populate HealthPlan table

insert into HealthPlan(Healthplan_code, Healthplan_vendor, Individual_rate, Family_Rate, Individual_Deductibles,Family_Deductibles, Co_pay, Co_insurance)

values

('Acme HMO','Acme',500,1000,1000,1500,20,10),

('BCBS HMO','BCBS',480,900,1200,2000,30,15),

('UHC HMO','UHC',450,800,1200,3000,30,20);

-- Populate Emp_insurance table

insert into Emp_insurance (Emp_id, Healthplan_code,Covereddependents_ind)

values

(1001,'Acme HMO',1),

(1002,'BCBS HMO',0),

(1003,'UHC HMO',0),

(1004,'Acme HMO',1),

(1006,'Acme HMO',1),

(1007,'BCBS HMO',0),

(1008,'UHC HMO',0),

(1009,'BCBS HMO',0),

```
(1011,'Acme HMO',0),
(1012,'BCBS HMO',0),
(1013,'UHC HMO',0),
(1014,'UHC HMO',1),
(1016,'Acme HMO',0),
(1017,'BCBS HMO',0),
(1018,'UHC HMO',0),
(1019,'Acme HMO',1),
(1021,'Acme HMO',0),
(1022,'BCBS HMO',0),
(1023,'UHC HMO',0),
(1024,'Acme HMO',1),
(1026,'Acme HMO',0),
(1027,'BCBS HMO',0),
(1028,'UHC HMO',0),
(1029,'Acme HMO',1),
(1031,'Acme HMO',0),
(1032,'BCBS HMO',0),
(1033,'UHC HMO',0),
(1034,'Acme HMO',0),
(1036,'BCBS HMO',0),
(1037,'UHC HMO',0);

-- Populate Insured_Dependents
insert into Insured_Dependents ( Emp_id, Dependent_id)
values
(1001,'D004'),
(1001,'D005'),
(1004,'D006'),
```

```
(1006,'D001'),
(1006,'D002'),
(1006,'D003'),
(1006,'D008'),
(1014,'D016'),
(1019,'D019'),
(1024,'D026'),
(1029,'D028');

-- Populate Restricted_Room table
 insert into Restricted_Room ( Room_num , Room_name,  Internal_Phnum,  Hazardous_ind , Role_id )
 values
('R001','Operations Room',3433339000,1,'R0008'),
('R002','Server Room',3433332300,0,'R0009'),
('R003','HR Confidential data Room',3433331101,0,'R0007');

-- Populate Restricted_Access table
 insert into Restricted_Access (emp_id, Room_num)
 values
(1001,'R001'),
(1001,'R002'),
(1001,'R003'),
(1002,'R003'),
(1003,'R001'),
(1004,'R002'),
(1006,'R003'),
(1011,'R003'),
```

```
(1014,'R001'),
(1020,'R001'),
(1023,'R002'),
(1036,'R002'),
(1037,'R002');

--update column vacation
Update Employee
set vacation = ( current_date -emp_dateofhire) /(365*3) +2

--end
```

5. Database Operational Testing

Following are the various Script that will be used for Testing . Few of the queries used are listed below based on Scenarios.

```
-- script for finding who is supervisor
select a.emp_name Employee_name, c.emp_name Manager_name, r.JobRole_desc
from employee a inner join emp_hierarchy b on a.emp_id = b.Emp_id
        inner join employee c on b.Manager_id = c.emp_id
                inner join roles r on a.Role_id= r.role_id
 order by c.emp_id


---- What is the name of the Atlas Corporation owner?
 Select a.Emp_name
 from Employee a inner join Emp_hierarchy b on  a.emp_id = b.Emp_id
```

```
inner join Roles c on a.Role_id = c.Role_id
                    where C.JobRole_desc ='Owner'
```

--- What are the names of employees who directly reports to the Atlas Corporation human resources director?

```
Select d.Emp_name from Roles a inner join employee b on a.role_id = b.Role_id  and a.JobRole_desc  ='Human Resource Director'
        inner join emp_hierarchy c on c.Manager_id = b.Emp_id
                inner join Employee d on c.Emp_id = d.Emp_id
```

--What are the names of employees who have access to restricted room 'Server Room' ?

```
Select c.Emp_name From Restricted_Room  a
   inner join Restricted_Access b on a.Room_num = b.Room_num and  a.Room_name ='Server Room'
      inner join Employee c on c.Emp_id = b.Emp_id
```

--How many employees have chosen the Acme HMO health plan?

```
Select count(1) from Emp_Insurance a where Healthplan_code ='Acme HMO'
```

--How many dependents does each employee have?

```
select a.Emp_name as Emp_name, count(dependent_id) Count_of_dependents from Employee a
```

```
        inner join Emp_Dependents b on a.Emp_id =b.Emp_id
            Group by a.Emp_name
```

-- What is the name of employee John Smith's emergency contact?

```
Select a.Contact_name from Emp_EmergencyContact a
    inner join Employee b on a.Emp_id = b.emp_id
        where b.Emp_name ='John Smith'
```

-- What is employee John Smith's job description?

```
select a.Emp_name, b.Jobrole_desc from Employee a  inner join Roles b on a.Role_id
= b.Role_id
                        and a.Emp_name ='John Smith'
```

-- What are the restricted rooms to which John Smith has access (if any)?

```
Select c.Room_name from Restricted_Access a inner join employee b on a.emp_id
=b.emp_id and b.Emp_name ='John Smith'
        inner join Restricted_Room c on a.Room_num = c.Room_num
```

-- Which of the health insurance plans has the most employees enrolled?

```
select Healthplan_code, count(emp_id)
from Emp_Insurance Group by HealthPlan_code
having count(emp_id)
 in
 (select max(a.Emp_count) from
```

```
(Select Healthplan_code, count(emp_id) Emp_count  from Emp_Insurance
Group by HealthPlan_code) a
)
```

--List all dependents of male employee whose salary is >300000

```
Select b.Emp_id,b.emp_name,  a.Dependent_name from emp_dependents a inner join Employee b on a.Emp_id = b.Emp_id and b.Emp_gender ='M'
    inner join Emp_salary c on b.emp_id = c.Emp_id and c.Annual_Sal >300000
        order by b.emp_id
```

-- Name of male Employee having authorised access to Server Room

```
Select Emp_name from Employee a inner join Restricted_Access b on a.Emp_id = b.Emp_id and  a.Emp_gender ='M'
                    inner join Restricted_Room c on b.Room_num = c.Room_num and c.Room_name ='Server Room'
```

-- Employees born after 1/1/1985

```
select Emp_name, Emp_birthdate from Employee where Emp_birthdate > cast('01/01/1985 ' as date);
```

-- list of employess who work under manager Cameroon Winklevoss after 2010

```
select c.Emp_name from Employee a inner join Emp_hierarchy b on a.Emp_id =
b.Manager_id and a.Emp_name ='Cameron Winklevoss'
                    inner join Employee c on b.Emp_id = c.Emp_id
                                                and c.Emp_Dateofhire >
cast('12/31/2009 ' as date)
```

```
-- Youngest male employee
select Emp_name from Employee where  Emp_gender='M' and  (current_date -
Emp_birthdate)
in (
select  min(current_date - Emp_birthdate)  from Employee where Emp_gender ='M')
```
