**Database Design Term Project**

Insurance Company

**Deivid Mafra Student ID: C0752173**

**Sherwayne Palmer Student ID: C0738019**

**Diego Perez .Student ID: C0746800**

**Manpreet Singh Sandhu Student ID: C0747450**

**Ramanpreet kaur Student ID: C0748669**

**Paramjeet kaur Student ID: C0747429**

Contents

[PART 1 3](#_Toc5160879)

[Company Overview 3](#_Toc5160880)

[Invoice 4](#_Toc5160881)

[PART 2 5](#_Toc5160882)

[Conceptual Design 5](#_Toc5160883)

[PART 3 AND 4 6](#_Toc5160884)

[Logical Model & Physical Model 6](#_Toc5160885)

[PART 5 9](#_Toc5160886)

[Business Rules/Database Constraints 9](#_Toc5160887)

[Implementation of Physical Model 9](#_Toc5160888)

[Database Constructions 9](#_Toc5160889)

[Insert / Populating Tables 16](#_Toc5160890)

[Constraints 16](#_Toc5160891)

# PART 1

Company Overview

Our company, Growing Together is an insurance company specialized in health insurance around the world. Our products are designed to fit from the basic needs to the most complex scenario.

Growing together team has members with more than 30 years of experience accumulating knowledge and ready to help you in each part of your life.

Our products characteristics:

Benefits

* Exclusions
* Limits
* Exclusions
* Available
* General\_condition

Invoice

| 2019-02-20 |  | Invoice No. 1234 |
| --- | --- | --- |
| **To**  John W. Lennon Abbey Road Liverpool, NW8, UK  **Ship To**  Same as recipient  **Instructions**  Delivery Instructions |  | |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Policy #** | **Quantity** | **Description** | | **Price** | | 2019021900001 | 1 | Health Insurance | | CAD5,000.00 | |  |  |  | |  | |  |  |  | |  | |  |  |  | |  | |  |  |  | |  | |  |  |  | |  | | Subtotal | CAD5,000.00 | | Sales Tax | CAD650.00 | | Shipping & Handling | CAD0.00 | | Total Due | CAD5,000.00 | |

Due upon receipt

Thank you for your business!

# PART 2

Conceptual Design

# PART 3 AND 4

Logical Model & Physical Model

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DEPARTMENT** | | | | |
| **Column** | **Key type** | **Data Type** | **Length** | **Optionality** |
| **department\_id** | PK | INTEGER |  | NOT NULL |
| department\_name |  | VARCHAR | 256 | NOT NULL |
| usr\_create |  | VARCHAR | 3 | NOT NULL |
| date\_create |  | DATE |  | NOT NULL |
| usr\_update |  | VARCHAR | 3 | NOT NULL |
| date\_update |  | DATE |  | NOT NULL |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CATEGORY** | | | | |
| **Column** | **Key type** | **Data Type** | **Length** | **Optionality** |
| **category\_id** | PK | VARCHAR | 3 | NOT NULL |
| category\_name |  | VARCHAR | 256 | NOT NULL |
| monthly\_price |  | NUMERIC | 6 | NOT NULL |
| coverage |  | DECIMAL |  | NOT NULL |
| term |  | INTEGER |  | NOT NULL |
| usr\_create |  | VARCHAR | 3 | NOT NULL |
| date\_create |  | DATE |  | NOT NULL |
| usr\_update |  | VARCHAR | 3 | NOT NULL |
| date\_update |  | DATE |  | NOT NULL |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SPONSOR** | | | | |
| **Column** | **Key type** | **Data Type** | **Length** | **Optionality** |
| **sponsor\_id** | PK | INTEGER |  | NOT NULL |
| company\_name |  | VARCHAR | 256 | NOT NULL |
| email |  | VARCHAR | 256 |  |
| phone |  | INTEGER |  | NOT NULL |
| sales\_rep |  | VARCHAR | 256 |  |
| address |  | VARCHAR | 256 | NOT NULL |
| usr\_create |  | VARCHAR | 3 | NOT NULL |
| date\_create |  | DATE |  | NOT NULL |
| usr\_update |  | VARCHAR | 3 | NOT NULL |
| date\_update |  | DATE |  | NOT NULL |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **COUNTRY** | | | | |
| **Column** | **Key type** | **Data Type** | **Length** | **Optionality** |
| **country\_id** | PK | INTEGER |  | NOT NULL |
| country\_name |  | VARCHAR | 256 | NOT NULL |
| usr\_create |  | VARCHAR | 3 | NOT NULL |
| date\_create |  | DATE |  | NOT NULL |
| usr\_update |  | VARCHAR | 3 | NOT NULL |
| date\_update |  | DATE |  | NOT NULL |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PROVINCE** | | | | |
| **Column** | **Key type** | **Data Type** | **Length** | **Optionality** |
| **province\_id** | PK | INTEGER |  | NOT NULL |
| province\_name |  | VARCHAR | 256 | NOT NULL |
| *country\_id* | FK | INTEGER |  | NOT NULL |
| usr\_create |  | VARCHAR | 3 | NOT NULL |
| date\_create |  | DATE |  | NOT NULL |
| usr\_update |  | VARCHAR | 3 | NOT NULL |
| date\_update |  | DATE |  | NOT NULL |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CITY** | | | | |
| **Column** | **Key type** | **Data Type** | **Length** | **Optionality** |
| **city\_id** | PK | INTEGER |  | NOT NULL |
| city\_name |  | VARCHAR | 256 | NOT NULL |
| *province\_id* | FK | INTEGER |  | NOT NULL |
| usr\_create |  | VARCHAR | 3 | NOT NULL |
| date\_create |  | DATE |  | NOT NULL |
| usr\_update |  | VARCHAR | 3 | NOT NULL |
| date\_update |  | DATE |  | NOT NULL |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PRODUCT** | | | | |
| **Column** | **Key type** | **Data Type** | **Length** | **Optionality** |
| **product\_id** | PK | INTEGER |  | NOT NULL |
| name |  | VARCHAR | 256 | NOT NULL |
| description |  | VARCHAR | 8000 | NOT NULL |
| definition |  | VARCHAR | 8000 | NOT NULL |
| limits |  | VARCHAR | 8000 | NOT NULL |
| exclusions |  | VARCHAR | 8000 | NOT NULL |
| general\_condition |  | VARCHAR | 8000 | NOT NULL |
| indenization\_procedure |  | VARCHAR | 8000 | NOT NULL |
| data\_protection\_advice |  | VARCHAR | 8000 | NOT NULL |
| emergency\_procedure |  | VARCHAR | 8000 | NOT NULL |
| available |  | BIT |  | NOT NULL |
| usr\_create |  | VARCAHR | 3 | NOT NULL |
| date\_create |  | DATE |  | NOT NULL |
| usr\_update |  | VARCHAR | 3 | NOT NULL |
| date\_update |  | DATE |  | NOT NULL |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STAFF** | | | | |
| **Column** | **Key type** | **Data Type** | **Length** | **Optionality** |
| **staff\_id** | PK | INTEGER |  | NOT NULL |
| date\_of\_birth |  | DATE |  | NOT NULL |
| last\_name |  | VARCHAR | 256 | NOT NULL |
| first\_name |  | VARCAHR | 256 | NOT NULL |
| phone |  | NUMERIC | 10 | NOT NULL |
| email |  | VARCAHR | 256 |  |
| address |  | VARCHAR | 256 | NOT NULL |
| zipcode |  | VARCHAR | 7 | NOT NULL |
| isemployee |  | BIT |  | NOT NULL |
| hire\_date |  | DATE |  | NOT NULL |
| *department\_id* | FK | INTEGER |  | NOT NULL |
| usr\_create |  | VARCHAR | 3 | NOT NULL |
| date\_create |  | DATE |  | NOT NULL |
| usr\_update |  | VARCHAR | 3 | NOT NULL |
| date\_update |  | DATE |  | NOT NULL |
| job\_ended |  | DATE |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CUSTOMER** | | | | |
| **Column** | **Key type** | **Data Type** | **Length** | **Optionality** |
| **customer\_id** | PK | INTEGER |  | NOT NULL |
| last\_name |  | VARCHAR | 256 | NOT NULL |
| first\_name |  | VARCHAR | 256 | NOT NULL |
| phone |  | INTEGER |  | NOT NULL |
| email |  | VARCHAR | 256 |  |
| address |  | VARCHAR | 256 | NOT NULL |
| zipcode |  | INTEGER |  | NOT NULL |
| date\_of\_birth |  | DATE |  | NOT NULL |
| *city\_id* | FK | INTEGER |  | NOT NULL |
| usr\_create |  | VARCHAR | 3 | NOT NULL |
| date\_create |  | DATE |  | NOT NULL |
| usr\_update |  | VARCHAR | 3 | NOT NULL |
| date\_update |  | DATE |  | NOT NULL |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **POLICY** | | | | |
| **Column** | **Key type** | **Data Type** | **Length** | **Optionality** |
| **policy\_id** | PK | INTEGER |  | NOT NULL |
| issue\_date |  | DATE |  | NOT NULL |
| expire\_date |  | DATE |  | NOT NULL |
| *previous\_policy* | FK | INTEGER |  |  |
| *staff\_id* | FK | INTEGER |  | NOT NULL |
| *customer\_id* | FK | INTEGER |  | NOT NULL |
| *product\_id* | FK | INTEGER |  | NOT NULL |
| *category\_id* | FK | VARCHAR | 3 | NOT NULL |
| *cause\_death\_id* | FK | INTEGER |  |  |
| usr\_create |  | VARCHAR | 3 | NOT NULL |
| date\_create |  | DATE |  | NOT NULL |
| usr\_update |  | VARCHAR | 3 | NOT NULL |
| date\_update |  | DATE |  | NOT NULL |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CAUSE\_OF\_DEATH** | | | | |
| **Column** | **Key type** | **Data Type** | **Length** | **Optionality** |
| **cause\_death\_id** | PK | INTEGER |  | NOT NULL |
| cause\_death\_description |  | VARCHAR | 8000 | NOT NULL |
| usr\_create |  | VARCHAR | 3 | NOT NULL |
| date\_create |  | DATE |  | NOT NULL |
| usr\_update |  | VARCHAR |  | NOT NULL |
| date\_update |  | DATE |  | NOT NULL |
| payable |  | BIT |  | NOT NULL |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **BENEFECIARY** | | | | |
| **Column** | **Key type** | **Data Type** | **Length** | **Optionality** |
| **beneficiary\_id** | PK | INTEGER |  | NOT NULL |
| beneficiary\_last\_name |  | VARCHAR | 256 | NOT NULL |
| beneficiary\_name |  | VARCHAR | 256 | NOT NULL |
| beneficiary\_middle\_name |  | VARCHAR | 256 |  |
| beneficiary\_brith\_day |  | DATE |  | NOT NULL |
| beneficiary\_relation |  | VARCHAR | 100 | NOT NULL |
| beneficiary\_percentage |  | NUMERIC | 4 | NOT NULL |
| *policy\_id* | FK | INTEGER |  | NOT NULL |
| usr\_create |  | VARCHAR | 3 | NOT NULL |
| date\_create |  | DATE |  | NOT NULL |
| usr\_update |  | VARCHAR | 3 | NOT NULL |
| date\_update |  | DATE |  | NOT NULL |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CATEGORY\_BENEFIT** | | | | |
| **Column** | **Key type** | **Data Type** | **Length** | **Optionality** |
| **category\_ben\_id** | PK | INTEGER |  | NOT NULL |
| *category\_id* | FK | VARCHAR | 3 | NOT NULL |
| *benefit\_id* | FK | INTEGER |  | NOT NULL |
| usr\_create |  | VARCHAR | 3 | NOT NULL |
| date\_create |  | DATE |  | NOT NULL |
| usr\_update |  | VARCHAR | 3 | NOT NULL |
| date\_update |  | DATE |  | NOT NULL |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **BENEFIT** | | | | |
| **Column** | **Key type** | **Data Type** | **Length** | **Optionality** |
| **benefit\_id** | PK | INTEGER |  | NOT NULL |
| benefit\_name |  | VARCHAR | 250 | NOT NULL |
| benefit\_description |  | VARCHAR | 3200 | NOT NULL |
| usr\_create |  | VARCHAR | 3 | NOT NULL |
| date\_create |  | DATE |  | NOT NULL |
| usr\_update |  | VARCHAR | 3 | NOT NULL |
| date\_update |  | DATE |  | NOT NULL |

# PART 5

Business Rules/Database Constraints

It’s presented in the Implementation of Physical Model section

Implementation of Physical Model

### Database Constructions

/\* This line to create the database.

create database INSURANCE\_COMPANY \*/

/\* This line to erase the database.

Drop database INSURANCE\_COMPANY \*/

USE INSURANCE\_COMPANY;

-- TABLE DEPARTMENT

CREATE TABLE DEPARTMENT

( -- The departmet\_id could be a varchar idetifier

department\_id INTEGER NOT NULL IDENTITY(1,1),

department\_name VARCHAR(256) NOT NULL,

usr\_create VARCHAR(3) NOT NULL,

date\_create DATE NOT NULL DEFAULT GETDATE(),

usr\_update VARCHAR(3) NOT NULL,

date\_update DATE NOT NULL DEFAULT GETDATE()

);

-- ADD PRIMARY KEY

ALTER TABLE DEPARTMENT

ADD CONSTRAINT department\_department\_id\_pk

PRIMARY KEY( department\_id );

-- ADD DATE\_UPDATE CONSTRAINT - IT COULD NOT BE BEFORE DATE\_CREATE

ALTER TABLE DEPARTMENT

ADD CONSTRAINT chk\_date\_update

CHECK (date\_update >= date\_create)

-- TABLE CATEGORY

CREATE TABLE CATEGORY

( category\_id VARCHAR(3) NOT NULL,

category\_name VARCHAR(256) NOT NULL,

monthly\_price NUMERIC(6,2) NOT NULL,

coverage DECIMAL(5,2) NOT NULL,

term INTEGER NOT NULL,

usr\_create VARCHAR(3) NOT NULL,

date\_create DATE NOT NULL DEFAULT GETDATE(),

usr\_update VARCHAR(3) NOT NULL,

date\_update DATE NOT NULL DEFAULT GETDATE()

);

ALTER TABLE CATEGORY

ADD CONSTRAINT category\_category\_id\_pk

PRIMARY KEY( category\_id );

-- TABLE SPONSOR

CREATE TABLE SPONSOR

( sponsor\_id INTEGER NOT NULL IDENTITY(1,1),

company\_name VARCHAR(256) NOT NULL,

email VARCHAR(256),

phone INTEGER NOT NULL,

sales\_rep VARCHAR(256),

address VARCHAR(256) NOT NULL,

usr\_create VARCHAR(3) NOT NULL,

date\_create DATE NOT NULL DEFAULT GETDATE(),

usr\_update VARCHAR(3) NOT NULL,

date\_update DATE NOT NULL DEFAULT GETDATE()

);

ALTER TABLE SPONSOR

ADD CONSTRAINT sponsor\_sponsor\_id\_pk

PRIMARY KEY( sponsor\_id );

-- TABLE COUNTRY

CREATE TABLE COUNTRY

( -- the country\_id could be a varchar column too.

country\_id INTEGER NOT NULL IDENTITY(1,1),

country\_name VARCHAR(256) NOT NULL,

usr\_create VARCHAR(3) NOT NULL,

date\_create DATE NOT NULL DEFAULT GETDATE(),

usr\_update VARCHAR(3) NOT NULL,

date\_update DATE NOT NULL DEFAULT GETDATE()

);

ALTER TABLE COUNTRY

ADD CONSTRAINT country\_country\_id\_pk

PRIMARY KEY( country\_id );

-- TABLE PROVINCE

CREATE TABLE PROVINCE

( --The province\_id could be a varchar too.

province\_id INTEGER NOT NULL IDENTITY(1,1),

province\_name VARCHAR(256) NOT NULL,

country\_id INTEGER NOT NULL,

usr\_create VARCHAR(3) NOT NULL,

date\_create DATE NOT NULL DEFAULT GETDATE(),

usr\_update VARCHAR(3) NOT NULL,

date\_update DATE NOT NULL DEFAULT GETDATE()

);

ALTER TABLE PROVINCE

ADD CONSTRAINT province\_province\_id\_pk

PRIMARY KEY( province\_id );

ALTER TABLE PROVINCE

ADD CONSTRAINT province\_country\_id\_fk

FOREIGN KEY( country\_id )

REFERENCES country( country\_id );

--TABLE CITY

CREATE TABLE CITY

( city\_id INTEGER NOT NULL IDENTITY(1,1),

city\_name VARCHAR(256) NOT NULL,

province\_id INTEGER NOT NULL,

usr\_create VARCHAR(3) NOT NULL,

date\_create DATE NOT NULL DEFAULT GETDATE(),

usr\_update VARCHAR(3) NOT NULL,

date\_update DATE NOT NULL DEFAULT GETDATE()

);

ALTER TABLE CITY

ADD CONSTRAINT city\_city\_id\_pk

PRIMARY KEY( city\_id );

ALTER TABLE CITY

ADD CONSTRAINT city\_province\_id\_fk

FOREIGN KEY( province\_id )

REFERENCES province( province\_id );

--TABLE PRODUCT

CREATE TABLE PRODUCT

( product\_id INTEGER NOT NULL IDENTITY(1,1),

name VARCHAR(256) NOT NULL,

description VARCHAR(8000) NOT NULL,

definition VARCHAR(8000) NOT NULL,

limits VARCHAR(8000) NOT NULL,

exclusions VARCHAR(8000) NOT NULL,

general\_condition VARCHAR(8000) NOT NULL,

indenization\_procedure VARCHAR(8000) NOT NULL,

data\_protection\_advice VARCHAR(8000) NOT NULL,

emergency\_procedure VARCHAR(8000) NOT NULL,

available BIT NOT NULL,

usr\_create VARCHAR(3) NOT NULL,

date\_create DATE NOT NULL DEFAULT GETDATE(),

usr\_update VARCHAR(3) NOT NULL,

date\_update DATE NOT NULL DEFAULT GETDATE()

);

ALTER TABLE PRODUCT

ADD CONSTRAINT product\_product\_id\_pk

PRIMARY KEY( product\_id );

--TABLE STAFF

CREATE TABLE STAFF

( staff\_id INTEGER NOT NULL IDENTITY(1,1),

date\_of\_birth DATE NOT NULL,

last\_name VARCHAR(256) NOT NULL,

first\_name VARCHAR(256) NOT NULL,

phone NUMERIC(10,0) NOT NULL,

email VARCHAR(256),

address VARCHAR(256) NOT NULL,

zipcode VARCHAR(7) NOT NULL,

isemployee BIT NOT NULL,

hire\_date DATE NOT NULL DEFAULT getdate(),

department\_id INTEGER NOT NULL,

usr\_create VARCHAR(3) NOT NULL,

date\_create DATE NOT NULL DEFAULT GETDATE(),

usr\_update VARCHAR(3) NOT NULL,

date\_update DATE NOT NULL DEFAULT GETDATE()

);

ALTER TABLE STAFF

ADD CONSTRAINT staff\_staff\_id\_pk

PRIMARY KEY( staff\_id );

ALTER TABLE STAFF

DROP COLUMN fire\_date;

ALTER TABLE STAFF

ADD job\_ended date;

--TABLE CUSTOMER

CREATE TABLE CUSTOMER

( customer\_id INTEGER NOT NULL IDENTITY(1,1),

last\_name VARCHAR(256) NOT NULL,

first\_name VARCHAR(256) NOT NULL,

phone NUMERIC(18) NOT NULL,

email VARCHAR(256),

address VARCHAR(256) NOT NULL,

zipcode VARCHAR(9) NOT NULL,

date\_of\_birth DATE NOT NULL,

city\_id INTEGER NOT NULL,

usr\_create VARCHAR(3) NOT NULL,

date\_create DATE NOT NULL DEFAULT GETDATE(),

usr\_update VARCHAR(3) NOT NULL,

date\_update DATE NOT NULL DEFAULT GETDATE()

);

ALTER TABLE CUSTOMER

ADD CONSTRAINT customer\_customer\_id\_pk

PRIMARY KEY( customer\_id );

--TABLE POLICY

CREATE TABLE POLICY

( policy\_id INTEGER NOT NULL IDENTITY(1,1),

issue\_date DATE NOT NULL DEFAULT GETDATE(),

expire\_date DATE NOT NULL,

previous\_policy INTEGER,

staff\_id INTEGER NOT NULL,

customer\_id INTEGER NOT NULL,

product\_id INTEGER NOT NULL,

category\_id VARCHAR(3) NOT NULL,

cause\_death\_id INTEGER,

usr\_create VARCHAR(3) NOT NULL,

date\_create DATE NOT NULL DEFAULT GETDATE(),

usr\_update VARCHAR(3) NOT NULL,

date\_update DATE NOT NULL DEFAULT GETDATE()

);

ALTER TABLE POLICY

ADD CONSTRAINT chkTerm\_expire\_date

CHECK (expire\_date >= issue\_date)

ALTER TABLE POLICY

ADD CONSTRAINT policy\_policy\_id\_pk

PRIMARY KEY( policy\_id );

ALTER TABLE POLICY

ADD CONSTRAINT policy\_previous\_policy\_fk

FOREIGN KEY( previous\_policy )

REFERENCES policy( policy\_id );

ALTER TABLE POLICY

ADD CONSTRAINT policy\_staff\_id\_fk

FOREIGN KEY( staff\_id )

REFERENCES staff( staff\_id );

ALTER TABLE POLICY

ADD CONSTRAINT policy\_customer\_id\_fk

FOREIGN KEY( customer\_id )

REFERENCES customer( customer\_id );

ALTER TABLE POLICY

ADD CONSTRAINT policy\_product\_id\_fk

FOREIGN KEY( product\_id )

REFERENCES product( product\_id );

ALTER TABLE POLICY

ADD CONSTRAINT policy\_category\_id\_fk

FOREIGN KEY( category\_id )

REFERENCES category( category\_id );

ALTER TABLE STAFF

ADD CONSTRAINT staff\_department\_id\_fk

FOREIGN KEY( department\_id )

REFERENCES department( department\_id );

ALTER TABLE CUSTOMER

ADD CONSTRAINT customer\_city\_id\_fk

FOREIGN KEY( city\_id )

REFERENCES city( city\_id );

CREATE TABLE BENEFIT

(

benefit\_id INTEGER NOT NULL IDENTITY(1,1),

benefit\_name VARCHAR(250) NOT NULL,

benefit\_description VARCHAR(3200) NOT NULL,

usr\_create VARCHAR(3) NOT NULL,

date\_create DATE NOT NULL,

usr\_update VARCHAR(3) NOT NULL,

date\_update DATE NOT NULL

);

ALTER TABLE BENEFIT

ADD CONSTRAINT benefit\_id\_pk

PRIMARY KEY( benefit\_id );

CREATE TABLE CATEGORY\_BENEFIT

(

category\_ben\_id INTEGER NOT NULL IDENTITY(1,1),

category\_id VARCHAR(3) NOT NULL,

benefit\_id INTEGER NOT NULL,

usr\_create VARCHAR(3) NOT NULL,

date\_create DATE NOT NULL,

usr\_update VARCHAR(3) NOT NULL,

date\_update DATE NOT NULL

);

ALTER TABLE CATEGORY\_BENEFIT

ADD CONSTRAINT category\_ben\_id\_pk

PRIMARY KEY( category\_ben\_id );

ALTER TABLE CATEGORY\_BENEFIT

ADD CONSTRAINT category\_catego\_id\_fk

FOREIGN KEY( category\_id )

REFERENCES category ( category\_id );

ALTER TABLE CATEGORY\_BENEFIT

ADD CONSTRAINT benefit\_catego\_id\_fk

FOREIGN KEY( benefit\_id )

REFERENCES benefit ( benefit\_id );

/\*

BENEFICIARY Table

1. add a "BENEFICIARY" and "CAUSE\_OF\_DEATH" tables

2. the primary key inside "CAUSE\_OF\_DEATH" should be a FK inside the POLICY table\*/

CREATE TABLE BENEFICIARY

(

beneficiary\_id INTEGER NOT NULL IDENTITY(1,1),

beneficiary\_last\_name VARCHAR(256) NOT NULL,

beneficiary\_name VARCHAR(256) NOT NULL,

beneficiary\_middle\_name VARCHAR(256),

beneficiary\_brith\_day DATE NOT NULL,

beneficiary\_relation VARCHAR(100) NOT NULL,

beneficiary\_percentage NUMERIC(5,2) NOT NULL,

policy\_id INTEGER NOT NULL, --Foreing key with policy

usr\_create VARCHAR(3) NOT NULL,

date\_create DATE NOT NULL DEFAULT GETDATE(),

usr\_update VARCHAR(3) NOT NULL,

date\_update DATE NOT NULL DEFAULT GETDATE()

);

ALTER TABLE BENEFICIARY

ADD CONSTRAINT beneficiary\_id\_pk

PRIMARY KEY( beneficiary\_id );

ALTER TABLE BENEFICIARY

ADD CONSTRAINT policy\_id\_fk

FOREIGN KEY( policy\_id )

REFERENCES policy ( policy\_id );

/\*

CAUSE\_OF\_DEATH Table

\*/

CREATE TABLE CAUSE\_OF\_DEATH

(

cause\_death\_id INTEGER NOT NULL IDENTITY(1,1),

cause\_death\_description VARCHAR(8000) NOT NULL,

usr\_create VARCHAR(3) NOT NULL,

date\_create DATE NOT NULL DEFAULT GETDATE(),

usr\_update VARCHAR(3) NOT NULL,

date\_update DATE NOT NULL DEFAULT GETDATE(),

payable BIT NOT NULL

);

ALTER TABLE CAUSE\_OF\_DEATH

ADD CONSTRAINT cause\_death\_id\_pk

PRIMARY KEY( cause\_death\_id );

ALTER TABLE POLICY

ADD CONSTRAINT cause\_death\_id\_fk

FOREIGN KEY( cause\_death\_id )

REFERENCES CAUSE\_OF\_DEATH ( cause\_death\_id );

### Insert / Populating Tables

### Constraints

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*01 - Constraint to avoid that the user insert an incorrect value for previous policy\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

ALTER TABLE POLICY

ADD CONSTRAINT policy\_previous\_policy\_fk

FOREIGN KEY( previous\_policy )

REFERENCES policy( policy\_id );

-- TESTING THE CONSTRAINT

INSERT INTO POLICY VALUES

('2019-03-21','2020-03-20',500,3,4,1,'GLD',2,'DPM',DEFAULT,'DAP','03-21-2019');

/\* RESULT:

Msg 547, Level 16, State 0, Line 10

The INSERT statement conflicted with the FOREIGN KEY SAME TABLE constraint "policy\_previous\_policy\_fk".

The conflict occurred in database "INSURANCE\_COMPANY", table "dbo.POLICY", column 'policy\_id'.

The statement has been terminated.\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*02 - Constraint to make valid our business rule that customers need to be 18 years or older\*\*

\*\*Maximum 75 years old

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

ALTER TABLE CUSTOMER

ADD CONSTRAINT customer\_date\_of\_birth\_ck

CHECK (date\_of\_birth<=GETDATE()-6570 AND date\_of\_birth>GETDATE()-27375)

-- TESTING THE CONSTRAINT

INSERT INTO CUSTOMER VALUES

('Mafra', 'Deivid',111-2222,'deividmafra@aol.com', '999, X street', 'M1Q2W3', '1900-04-06', '1','DPM', GETDATE(), 'DPM', GETDATE());

/\* RESULT:

Msg 547, Level 16, State 0, Line 27

The INSERT statement conflicted with the CHECK constraint "customer\_date\_of\_birth\_ck".

The conflict occurred in database "INSURANCE\_COMPANY", table "dbo.CUSTOMER", column 'date\_of\_birth'.

The statement has been terminated.\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*03 - Constraint to avoid that the user insert a job ended date previously that hire date \*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

ALTER TABLE STAFF

ADD CONSTRAINT staff\_job\_ended\_ck

CHECK(job\_ended>=hire\_date OR job\_ended IS NULL);

-- TESTING THE CONSTRAINT

INSERT INTO STAFF VALUES

('1980-10-16','Walker', 'John',111-2222,'wj@aol.com', '999, X street', 'M1Q2W3', '2015-03-01', 1,'DPM', GETDATE(), 'DPM', GETDATE(), '2015-02-01');

/\* RESULT:

Msg 547, Level 16, State 0, Line 44

The INSERT statement conflicted with the CHECK constraint "staff\_job\_ended\_ck".

The conflict occurred in database "INSURANCE\_COMPANY", table "dbo.STAFF".

The statement has been terminated.\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*04 - Constraint to confirm beneficiary percentage greater then 0 and maximum 100% \*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

ALTER TABLE BENEFICIARY

ADD CONSTRAINT beneficiary\_beneficiary\_percentage\_ck

CHECK(beneficiary\_percentage>0 AND beneficiary\_percentage<=100);

-- TESTING THE CONSTRAINT

INSERT INTO BENEFICIARY VALUES

('Paris','Nathalia', 'Andrea', '1988-04-06', 'Spouse', 101, 14, 'DAP',GETDATE(), 'DAP',GETDATE());

/\* RESULT:

Msg 547, Level 16, State 0, Line 61

The INSERT statement conflicted with the CHECK constraint "beneficiary\_beneficiary\_percentage\_ck".

The conflict occurred in database "INSURANCE\_COMPANY", table "dbo.BENEFICIARY", column 'beneficiary\_percentage'.

The statement has been terminated.\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*05 - Constraint to confirm the maximum value of coverage to be paid is 4,000,000.00 \*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

ALTER TABLE CATEGORY

ADD CONSTRAINT category\_coverage\_ck

CHECK(coverage<=4000000);

-- TESTING THE CONSTRAINT

INSERT INTO CATEGORY VALUES

('DIA', 'DIAMOND', 20, 5000000, 5, 'DPM', GETDATE(), 'DPM', GETDATE());

/\* RESULT:

Msg 547, Level 16, State 0, Line 79

The INSERT statement conflicted with the CHECK constraint "category\_coverage\_ck".

The conflict occurred in database "INSURANCE\_COMPANY", table "dbo.CATEGORY", column 'coverage'.

The statement has been terminated.\*/