# Project

## Background

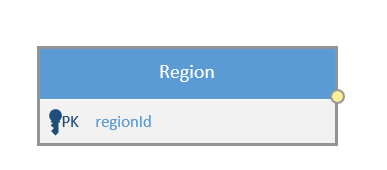
A **region** defines a geographical area, e.g. Auckland. A **branch** is a retail shop, located in a region.

## Requirement

Customers can become members of the retail shop. The customer will be associated with the branch they commonly shop at.

## Task

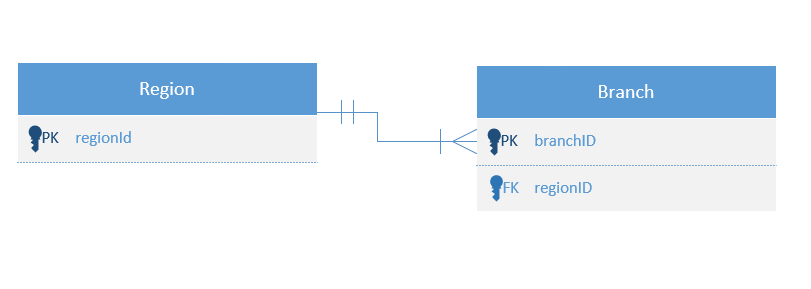
The database needs two new tables: **Region** and **Branch**. Design those tables on the whiteboard, indicating the column names and data types. Also demonstrate knowledge of SQL objects and schema, by describing how to achieve the table rules.

STEP 1. Find all entity in the project.

Region

Branch

STEP 2. Find the relationship of entity



Branch and Region

* Does region have many branches? YES
* Does a branch only belong to one region? YES
* Does branch belong into multiple region? NO

**One to many relationships between region and branch**

STEP 3: Find all the attribute needed per project. Do not add anything.

Entity1: Region

Attrib1 : regionID integer, primary key ,not null ,auto generated

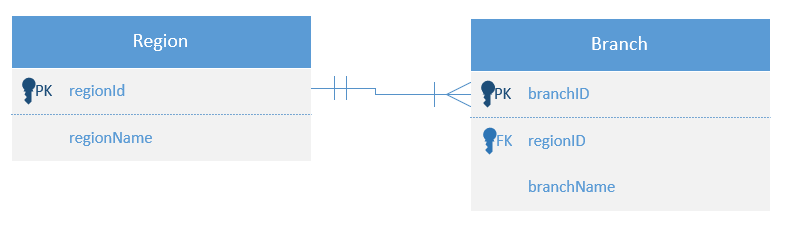
Atttrib2: regionName string, unique, not null

Entitiy2: Branch

Attrib1: branchID integer, primary key ,not null ,auto generated

Attrib2: branchName string, unique, not null

Attrib3: regionID foreign key int not null



Step 4: Build database

### Region

Data to be stored

* An internal region number, only used by the software. The user never sees this.
* The name of the region. Can’t be blank.

Table rules

* The internal number must be different on every record.
* The name must be different on every record.

CREATE TABLE [dbo].[Region]

(

[regionID] [int] IDENTITY(1,1) NOT NULL primary key,

[regionName] [varchar](50) NOT NULL,

CONSTRAINT regionName\_unique UNIQUE (regionName)

)

### Branch

Data to be stored

* An internal branch number, only used by the software. The user never sees this.
* The name of the branch. Can’t be blank.
* The region this branch is located in.

Table rules

* The internal number must be different on every record.
* The name must be different on every record.
* The region must exist in the Region table.

CREATE TABLE [dbo].[Branch]

(

[branchID] [int] IDENTITY(1,1) NOT NULL primary key,

[branchName] [varchar](50) NOT NULL,

[regionId] [int] NOT NULL foreign key references Region(regionID),

CONSTRAINT branchName\_unique UNIQUE (branchName)

)

## Find all branches in the Auckland region

The application displays a list of branches in each region. Write a statement to return all branches in the Auckland region.

SELECT

br.branchName

FROM

[dbo].[Region] reg

JOIN

[dbo].[Branch] br

ON

br.regionID = reg.regionID

WHERE

reg.regionName = 'Auckland'

TYPE OF CONSTRAINT IN SQL

SQL NOT NULL – attribute data is not empty

UNIQUE – no data are the same, accept null value

PRIMARY KEY – no data are the same. No null.

FOREIGN KEY – one to many relationship.

CHECK – This will the data before it will store in the database

DEFAULT – This will store a default value in the table. Example is getdate command