**Mindit Interview Exercise**

**– SQL and Data Transfer –**

xxc

**Mindit Services SRL**

5, Intrarea IL. Caragiale Street,

Bucharest, District 2, 020047

Romania

|  |  |
| --- | --- |
| **Date of interview** |  |
| **Candidate's name** |  |
| **Applying for** |  |
| **Time allowed for test** |  |

# Initial Data

We have the following three systems, each one with its own database:

1. Item Catalogue – contains information about items
2. Pricing Application – contains information about item prices
3. Promotion Application – is an application that allows user to define promotions

# Promotion Application

In order to be able to define promotions, the Promotion Application database must have the following structure for items:

## PA\_ITEM

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Column** | **PK/Unique** | **Type** | **NOT NULL** | **Description** |
| **ITEM\_ID** | PK | INTEGER | Yes | The item id. Uniquely identifies an item |
| **ITEM\_NAME** | Unique | VARCHAR (100) | Yes | The item name |
| **ITEM\_DESCRIPTION** |  | VARCHAR (1000) | Yes | The item description |
| **BRAND\_ID** |  | INTEGER | Yes | The id of the brand associated with this item |
| **BRAND\_NAME** |  | VARCHAR (100) | Yes | The brand name |
| **CATEGORY\_ID** |  | INTEGER | Yes | The id of the category associated with this item |
| **CATEGORY\_NAME** |  | VARCHAR (100) | Yes | The category name |
| **PRICE\_DATE** |  | DATE |  | The date of the last price for the current item |
| **PRICE\_VALUE** |  | FLOAT |  | The last price value for the current item |

The data from the table above must be brought from the Item Catalogue and Pricing Application combined.

# Item Catalogue

The item catalogue contains the following tables:

## IC\_CATEGORY

| **Column** | **PK/Unique** | **Type** | **NOT NULL** | **Description** |
| --- | --- | --- | --- | --- |
| **CATEGORY\_ID** | PK | INTEGER | Yes | The category id |
| **CATEGORY\_NAME** | Unique | VARCHAR (100) | Yes | The category name, e.g. Food, Perfumes |
| **CATEGORY\_DESC** |  | VARCHAR (1000) | No | The category description |

## IC\_BRAND

| **Column** | **PK/Unique** | **Type** | **NOT NULL** | **Description** |
| --- | --- | --- | --- | --- |
| **BRAND\_ID** | PK | INTEGER | Yes | The brand id |
| **BRAND\_NAME** | Unique | VARCHAR (100) | Yes | The brand name, e.g. Chanel, Toblerone, Johnny Walker |
| **BRAND\_DESC** |  | VARCHAR (1000) | No | The brand description |

## IC\_ITEM

| **Column** | **PK/Unique** | **Type** | **Is Mandatory** | **Description** |
| --- | --- | --- | --- | --- |
| ITEM\_ID | PK | INTEGER | Yes | The item id. Uniquely identifies an item |
| ITEM\_NAME | Unique | VARCHAR (100) | Yes | The item name |
| ITEM\_DESCRIPTION |  | VARCHAR (1000) | Yes | The item description |
| BRAND\_ID |  | INTEGER | Yes | The id of the brand associated with this item |
| CATEGORY\_ID |  | INTEGER | Yes | The id of the category associated with this item |

# Pricing Application

This is a separate database which keeps the history of prices for all the item at a certain date. The table structure is defined below:

## PR\_PRICE

| **Column** | **PK/Unique** | **Type** | **NOT NULL** | **Description** |
| --- | --- | --- | --- | --- |
| **ITEM\_ID** | PK | INTEGER | Yes | The item id |
| **PRICE\_DATE** | PK | DATE | Yes | The date when the current price was valid |
| **PRICE\_VALUE** |  | FLOAT | Yes | The price (expressed in EUR) for the current item at the indicated date |

# Requirements

1. Create a script that builds three separate MySQL / MS SQL / Oracle (developer’s choice) databases for each of the three applications:
   1. Promotion Application
   2. Item Catalogue
   3. Pricing Application
2. Populate following tables with test data (you decide what the data should look like), as described in the table below.

|  |  |  |
| --- | --- | --- |
| **Application** | **Table** | **Number of Generated Records** |
| **Item Catalogue** | IC\_CATEGORY | 5 distinct categories |
| **Item Catalogue** | IC\_BRAND | 10 distinct brands |
| **Item Catalogue** | IC\_ITEM | 10.000 distinct items |
| **Pricing Application** | PR\_PRICE | 100000 distinct items (10 prices at 10 distinct dates) |

1. Create an ETL that populates the table PA\_ITEM from the Promotion Application with data from the Item Catalogue and Pricing applications. Please respect the definition of the PA\_ITEM table.

You can choose whatever ETL technology you are familiar with.

The candidate will provide:

1. Scripts to create databases
2. Scripts to generate test data
3. ETL source code
4. Preferably the source code should versioned, e.g. with Github/Bitbucket candidate’s personal account

Estimated time for completion – 4 to 8 hours.