# Упражнение: Съхранени процедури

# Заявки за БД SoftUni

## Employees with Salary Above 35000

Създайте съхранена процедура usp\_ **GetEmployeesSalaryAbove35000,** която връща **първото и фамилното име** на всички служители, чиято заплата е **над 35000**.

#### Пример

|  |  |
| --- | --- |
| **First Name** | **Last Name** |
| Roberto | Tamburello |
| David | Bradley |
| Terri | Duffy |
| … | … |

## Employees with Salary Above Number

Създайте съхранена процедура usp\_GetEmployeesSalaryAboveNumber, която приема **число**  (от тип **DECIMAL(18,4)**) като параметър и връща име и фамилия на всички **служители,** чиято заплата е над или **равна** на даденото число.

#### Пример

Примерът е със стойност 48100.

|  |  |
| --- | --- |
| **First Name** | **Last Name** |
| Terri | Duffy |
| Jean | Trenary |
| Ken | Sanchez |
| … | … |

## Town Names Starting With

Напишете съхранена процедура **usp\_GetTownsStartingWith,** която приема низ **като параметър** и връща всички имена **на градове, започващи с този низ.**

#### Пример

Ето списъка на всички **градове, започващи с "b".**

|  |
| --- |
| **Town** |
| Bellevue |
| Bothell |
| Bordeaux |
| Berlin |

## Employees from Town

Напишете съхранена процедура **usp\_GetEmployeesFromTown,** която приема името на **града** като параметър и връща име и фамилия на **служителите, които живеят в дадения град.**

#### Пример

Ето го списък на служителите, **живеещи в София.**

|  |  |
| --- | --- |
| **First Name** | **Last Name** |
| Svetlin | Nakov |
| Martin | Kulov |
| George | Denchev |

## Employees by Salary Level

Напишете съхранена процедура **usp\_EmployeesBySalaryLevel,** която получава като параметър ниво **на заплата** (ниско, средно или високо) и отпечатва **имената на всички служители,** които са със съответното ниво на заплата. Трябва да използвате функцията - "**dbo.ufn\_GetSalaryLevel(@Salary)** ", която беше част от предишната задача, вътре в вашата "**CREATE PROCEDURE …**" заявка.

#### Пример

Ето списъка на всички служители с висока заплата.

|  |  |
| --- | --- |
| **First Name** | **Last Name** |
| Terri | Duffy |
| Jean | Trenary |
| Ken | Sanchez |
| … | … |

## \* Delete Employees and Departments

Напишете **процедура**  с името **usp\_DeleteEmployeesFromDepartment**  **(@departmentId** INT**),** която изтрива всички **Служители** от даден **отдел**. **Изтрийте тези отдели**  и от таблицата **Departments,** също. **Накрая ИЗБЕРЕТЕ**  **броя** на **служителите** от дадения **отдел**. Ако декларациите за изтриване са правилни заявката за избор трябва да върне 0.

След завършване на това упражнение възстановете базата данни, за да възстановите всички промени.

#### Подсказка:

Може да зададете колона **ManagerID** в таблица Departments на **nullable** (с помощта на заявка "ALTER TABLE ...").

# Заявки за БД Bank

## Find Full Name

Дадена ви е схема на база данни с таблици **AccountHolders(Id (PK), FirstName, LastName, SSN)** и **акаунти(Id (PK), AccountHolderId (FK), Ballance)**. Напишете съхранена процедура **usp\_GetHoldersFullName,** която избира пълните имена на всички хора.

#### Пример

|  |
| --- |
| **Full Name** |
| Susan Cane |
| Kim Novac |
| Jimmy Henderson |
| … |

## People with Balance Higher Than

Вашата задача е да създадете съхранена процедура  **usp\_GetHoldersWithBalanceHigherThan,** която приема число като **параметър** и връща **всички хора, които имат в наличност повече пари от посоченото число**. Подредете ги по собствено име, след това по фамилия

#### Пример

|  |  |
| --- | --- |
| **First Name** | **Last Name** |
| Monika | Miteva |
| Petar | Kirilov |
| … | … |

## Calculating Interest

Вашата задача е да създадете съхранена процедура **usp\_CalculateFutureValueForAccount,** която използва функцията от предходната задача, за да даде лихва на сметката на дадено лице в продължение на **5 години**, заедно с информация за неговия/нейния идентификационен номер на **акаунт, собствено име, фамилия и текущ баланс,** както е показано в примера по-долу. Тя следва да приема **AccountId** и лихвения **процент** като параметри. Отново ви се предоставя "**dbo.ufn\_CalculateFutureValue**" функция, която е била част от предишната задача.

#### Пример

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Account Id** | **First Name** | **Last Name** | **Current Balance** | **Balance in 5 years** |
| 1 | Susan | Cane | 123.12 | 198.2860 |

\*За примера лихвата е 0.1