SQL Server:

Таблицы:

CREATE TABLE Сотрудники (  
 ID INT PRIMARY KEY,  
 Имя NVARCHAR(100),  
 Фамилия NVARCHAR(100),  
 Образование\_ID INT,  
 Семейное\_положение NVARCHAR(100),  
 Страховой\_полюс\_ID INT,  
 Пол NVARCHAR(10),  
 FOREIGN KEY (Образование\_ID) REFERENCES Образование(ID),  
 FOREIGN KEY (Страховой\_полюс\_ID) REFERENCES Страховой\_полюс(ID)  
);  
  
CREATE TABLE Образование (  
 ID INT PRIMARY KEY,  
 Уровень NVARCHAR(100),  
 Специальность NVARCHAR(100),  
 Учебное\_заведение NVARCHAR(100)  
);  
  
CREATE TABLE Страховой\_полюс (  
 ID INT PRIMARY KEY,  
 Номер\_полюса NVARCHAR(50),  
 Страховая\_компания NVARCHAR(100)  
);  
  
CREATE TABLE Предыдущие\_места\_работы (  
 ID INT PRIMARY KEY,  
 Сотрудник\_ID INT,  
 Место\_работы NVARCHAR(100),  
 Должность NVARCHAR(100),  
 FOREIGN KEY (Сотрудник\_ID) REFERENCES Сотрудники(ID)  
);  
  
CREATE TABLE Льготы (  
 ID INT PRIMARY KEY,  
 Сотрудник\_ID INT,  
 Льгота NVARCHAR(100),  
 FOREIGN KEY (Сотрудник\_ID) REFERENCES Сотрудники(ID)  
);  
  
CREATE TABLE Справки (  
 ID INT PRIMARY KEY,  
 Сотрудник\_ID INT,  
 Состояние\_здоровья NVARCHAR(100),  
 FOREIGN KEY (Сотрудник\_ID) REFERENCES Сотрудники(ID)  
);  
  
CREATE TABLE Отпуска (  
 ID INT PRIMARY KEY,  
 Сотрудник\_ID INT,  
 Дата\_начала DATE,  
 Дата\_окончания DATE,  
 FOREIGN KEY (Сотрудник\_ID) REFERENCES Сотрудники(ID)  
);  
  
CREATE TABLE Командировки (  
 ID INT PRIMARY KEY,  
 Сотрудник\_ID INT,  
 Место NVARCHAR(100),  
 Дата\_начала DATE,  
 Дата\_окончания DATE,  
 FOREIGN KEY (Сотрудник\_ID) REFERENCES Сотрудники(ID)  
);  
  
CREATE TABLE Больничные (  
 ID INT PRIMARY KEY,  
 Сотрудник\_ID INT,  
 Дата\_начала DATE,  
 Дата\_окончания DATE,  
 FOREIGN KEY (Сотрудник\_ID) REFERENCES Сотрудники(ID)  
);  
  
CREATE TABLE Поощрения\_и\_наказания (  
 ID INT PRIMARY KEY,  
 Сотрудник\_ID INT,  
 Тип NVARCHAR(100),  
 Описание NVARCHAR(100),  
 FOREIGN KEY (Сотрудник\_ID) REFERENCES Сотрудники(ID)  
);  
  
CREATE TABLE Передвижения\_по\_должностям (  
 ID INT PRIMARY KEY,  
 Сотрудник\_ID INT,  
 Должность NVARCHAR(100),  
 Дата\_начала DATE,  
 Дата\_окончания DATE,  
 FOREIGN KEY (Сотрудник\_ID) REFERENCES Сотрудники(ID)  
);  
  
CREATE TABLE Передвижение\_по\_отделам (  
 ID INT PRIMARY KEY,  
 Сотрудник\_ID INT,  
 Отдел NVARCHAR(100),  
 Дата\_начала DATE,  
 Дата\_окончания DATE,  
 FOREIGN KEY (Сотрудник\_ID) REFERENCES Сотрудники(ID)  
);  
  
CREATE TABLE Повышение\_квалификации (  
 ID INT PRIMARY KEY,  
 Сотрудник\_ID INT,  
 Курс NVARCHAR(100),  
 Дата\_начала DATE,  
 Дата\_окончания DATE,  
 FOREIGN KEY (Сотрудник\_ID) REFERENCES Сотрудники(ID)  
);  
  
CREATE TABLE Должности (  
 ID INT PRIMARY KEY,  
 Название NVARCHAR(100),  
 Описание NVARCHAR(100)  
);  
  
CREATE TABLE Отделы (  
 ID INT PRIMARY KEY,  
 Название NVARCHAR(100),  
 Описание NVARCHAR(100)  
);  
  
CREATE TABLE Паспорт (  
 ID INT PRIMARY KEY,  
 сотрудник\_ID INT,  
 Серия NVARCHAR(50),  
 Номер NVARCHAR(50),  
 Кем\_выдано NVARCHAR(100),  
 Дата\_выдачи DATE,  
 FOREIGN KEY (сотрудник\_ID) REFERENCES Сотрудники(ID)  
);

Индексы, последовательности и представления:

-- Для таблицы Сотрудники  
CREATE INDEX idx\_Сотрудники\_Имя ON Сотрудники (Имя);  
CREATE VIEW v\_Сотрудники AS SELECT ID, Имя, Фамилия FROM Сотрудники;  
CREATE SEQUENCE seq\_Сотрудники START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Образование  
CREATE INDEX idx\_Образование\_Уровень ON Образование (Уровень);  
CREATE VIEW v\_Образование AS SELECT ID, Уровень, Специальность FROM Образование;  
CREATE SEQUENCE seq\_Образование START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Страховой\_полюс  
CREATE INDEX idx\_Страховой\_полюс\_Номер\_полюса ON Страховой\_полюс (Номер\_полюса);  
CREATE VIEW v\_Страховой\_полюс AS SELECT ID, Номер\_полюса, Страховая\_компания FROM Страховой\_полюс;  
CREATE SEQUENCE seq\_Страховой\_полюс START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Предыдущие\_места\_работы  
CREATE INDEX idx\_Предыдущие\_места\_работы\_Сотрудник\_ID ON Предыдущие\_места\_работы (Сотрудник\_ID);  
CREATE VIEW v\_Предыдущие\_места\_работы AS SELECT ID, Сотрудник\_ID, Место\_работы, Должность FROM Предыдущие\_места\_работы;  
CREATE SEQUENCE seq\_Предыдущие\_места\_работы START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Льготы  
CREATE INDEX idx\_Льготы\_Сотрудник\_ID ON Льготы (Сотрудник\_ID);  
CREATE VIEW v\_Льготы AS SELECT ID, Сотрудник\_ID, Льгота FROM Льготы;  
CREATE SEQUENCE seq\_Льготы START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Справки  
CREATE INDEX idx\_Справки\_Сотрудник\_ID ON Справки (Сотрудник\_ID);  
CREATE VIEW v\_Справки AS SELECT ID, Сотрудник\_ID, Состояние\_здоровья FROM Справки;  
CREATE SEQUENCE seq\_Справки START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Отпуска  
CREATE INDEX idx\_Отпуска\_Сотрудник\_ID ON Отпуска (Сотрудник\_ID);  
CREATE VIEW v\_Отпуска AS SELECT ID, Сотрудник\_ID, Дата\_начала, Дата\_окончания FROM Отпуска;  
CREATE SEQUENCE seq\_Отпуска START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Командировки  
CREATE INDEX idx\_Командировки\_Сотрудник\_ID ON Командировки (Сотрудник\_ID);  
CREATE VIEW v\_Командировки AS SELECT ID, Сотрудник\_ID, Место, Дата\_начала, Дата\_окончания FROM Командировки;  
CREATE SEQUENCE seq\_Командировки START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Больничные  
CREATE INDEX idx\_Больничные\_Сотрудник\_ID ON Больничные (Сотрудник\_ID);  
CREATE VIEW v\_Больничные AS SELECT ID, Сотрудник\_ID, Дата\_начала, Дата\_окончания FROM Больничные;  
CREATE SEQUENCE seq\_Больничные START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Поощрения\_и\_наказания  
CREATE INDEX idx\_Поощрения\_и\_наказания\_Сотрудник\_ID ON Поощрения\_и\_наказания (Сотрудник\_ID);  
CREATE VIEW v\_Поощрения\_и\_наказания AS SELECT ID, Сотрудник\_ID, Тип, Описание FROM Поощрения\_и\_наказания;  
CREATE SEQUENCE seq\_Поощрения\_и\_наказания START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Передвижения\_по\_должностям  
CREATE INDEX idx\_Передвижения\_по\_должностям\_Сотрудник\_ID ON Передвижения\_по\_должностям (Сотрудник\_ID);  
CREATE VIEW v\_Передвижения\_по\_должностям AS SELECT ID, Сотрудник\_ID, Должность, Дата\_начала, Дата\_окончания FROM Передвижения\_по\_должностям;  
CREATE SEQUENCE seq\_Передвижения\_по\_должностям START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Передвижение\_по\_отделам  
CREATE INDEX idx\_Передвижение\_по\_отделам\_Сотрудник\_ID ON Передвижение\_по\_отделам (Сотрудник\_ID);  
CREATE VIEW v\_Передвижение\_по\_отделам AS SELECT ID, Сотрудник\_ID, Отдел, Дата\_начала, Дата\_окончания FROM Передвижение\_по\_отделам;  
CREATE SEQUENCE seq\_Передвижение\_по\_отделам START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Повышение\_квалификации  
CREATE INDEX idx\_Повышение\_квалификации\_Сотрудник\_ID ON Повышение\_квалификации (Сотрудник\_ID);  
CREATE VIEW v\_Повышение\_квалификации AS SELECT ID, Сотрудник\_ID, Курс, Дата\_начала, Дата\_окончания FROM Повышение\_квалификации;  
CREATE SEQUENCE seq\_Повышение\_квалификации START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Должности  
CREATE INDEX idx\_Должности\_Название ON Должности (Название);  
CREATE VIEW v\_Должности AS SELECT ID, Название, Описание FROM Должности;  
CREATE SEQUENCE seq\_Должности START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Отделы  
CREATE INDEX idx\_Отделы\_Название ON Отделы (Название);  
CREATE VIEW v\_Отделы AS SELECT ID, Название, Описание FROM Отделы;  
CREATE SEQUENCE seq\_Отделы START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Паспорт  
CREATE INDEX idx\_Паспорт\_сотрудник\_ID ON Паспорт (сотрудник\_ID);  
CREATE VIEW v\_Паспорт AS SELECT ID, сотрудник\_ID, Серия, Номер, Кем\_выдано, Дата\_выдачи FROM Паспорт;  
CREATE SEQUENCE seq\_Паспорт START WITH 1 INCREMENT BY 1;

Функции и процедуры:

-- Функция для добавления нового сотрудника  
CREATE PROCEDURE *ДобавитьСотрудника* @Имя NVARCHAR(100),  
 @Фамилия NVARCHAR(100),  
 @Образование\_ID INT,  
 @Семейное\_положение NVARCHAR(100),  
 @Страховой\_полюс\_ID INT,  
 @Пол NVARCHAR(10)  
AS  
BEGIN  
 INSERT INTO Сотрудники (Имя, Фамилия, Образование\_ID, Семейное\_положение, Страховой\_полюс\_ID, Пол)  
 VALUES (@Имя, @Фамилия, @Образование\_ID, @Семейное\_положение, @Страховой\_полюс\_ID, @Пол);  
END;  
  
-- Процедура для удаления сотрудника  
CREATE PROCEDURE *УдалитьСотрудника* @ID INT  
AS  
BEGIN  
 DELETE FROM Сотрудники WHERE ID = @ID;  
END;  
  
-- Функция для получения количества сотрудников с определенным образованием  
CREATE FUNCTION *Количество\_Сотрудников\_с\_Образованием* (@Образование\_ID INT)  
RETURNS INT  
AS  
BEGIN  
 RETURN (SELECT *COUNT*(\*) FROM Сотрудники WHERE Образование\_ID = @Образование\_ID);  
END;

Oracle:

Таблицы:

CREATE TABLE Сотрудники (  
 ID NUMBER PRIMARY KEY,  
 Имя NVARCHAR2(100),  
 Фамилия NVARCHAR2(100),  
 Образование\_ID NUMBER,  
 Семейное\_положение NVARCHAR2(100),  
 Страховой\_полюс\_ID NUMBER,  
 Пол NVARCHAR2(10)  
);  
  
CREATE TABLE Образование (  
 ID NUMBER PRIMARY KEY,  
 Уровень NVARCHAR2(100),  
 Специальность NVARCHAR2(100),  
 Учебное\_заведение NVARCHAR2(100)  
);  
  
CREATE TABLE Страховой\_полюс (  
 ID NUMBER PRIMARY KEY,  
 Номер\_полюса NVARCHAR2(50),  
 Страховая\_компания NVARCHAR2(100)  
);  
  
CREATE TABLE Предыдущие\_места\_работы (  
 ID NUMBER PRIMARY KEY,  
 Сотрудник\_ID NUMBER,  
 Место\_работы NVARCHAR2(100),  
 Должность NVARCHAR2(100),  
 FOREIGN KEY (Сотрудник\_ID) REFERENCES Сотрудники(ID)  
);  
  
CREATE TABLE Льготы (  
 ID NUMBER PRIMARY KEY,  
 Сотрудник\_ID NUMBER,  
 Льгота NVARCHAR2(100),  
 FOREIGN KEY (Сотрудник\_ID) REFERENCES Сотрудники(ID)  
);  
  
CREATE TABLE Справки (  
 ID NUMBER PRIMARY KEY,  
 Сотрудник\_ID NUMBER,  
 Состояние\_здоровья NVARCHAR2(100),  
 FOREIGN KEY (Сотрудник\_ID) REFERENCES Сотрудники(ID)  
);  
  
CREATE TABLE Отпуска (  
 ID NUMBER PRIMARY KEY,  
 Сотрудник\_ID NUMBER,  
 Дата\_начала DATE,  
 Дата\_окончания DATE,  
 FOREIGN KEY (Сотрудник\_ID) REFERENCES Сотрудники(ID)  
);  
  
CREATE TABLE Командировки (  
 ID NUMBER PRIMARY KEY,  
 Сотрудник\_ID NUMBER,  
 Место NVARCHAR2(100),  
 Дата\_начала DATE,  
 Дата\_окончания DATE,  
 FOREIGN KEY (Сотрудник\_ID) REFERENCES Сотрудники(ID)  
);  
  
CREATE TABLE Больничные (  
 ID NUMBER PRIMARY KEY,  
 Сотрудник\_ID NUMBER,  
 Дата\_начала DATE,  
 Дата\_окончания DATE,  
 FOREIGN KEY (Сотрудник\_ID) REFERENCES Сотрудники(ID)  
);  
  
CREATE TABLE Поощрения\_и\_наказания (  
 ID NUMBER PRIMARY KEY,  
 Сотрудник\_ID NUMBER,  
 Тип NVARCHAR2(100),  
 Описание NVARCHAR2(100),  
 FOREIGN KEY (Сотрудник\_ID) REFERENCES Сотрудники(ID)  
);  
  
CREATE TABLE Передвижения\_по\_должностям (  
 ID NUMBER PRIMARY KEY,  
 Сотрудник\_ID NUMBER,  
 Должность NVARCHAR2(100),  
 Дата\_начала DATE,  
 Дата\_окончания DATE,  
 FOREIGN KEY (Сотрудник\_ID) REFERENCES Сотрудники(ID)  
);  
  
CREATE TABLE Передвижение\_по\_отделам (  
 ID NUMBER PRIMARY KEY,  
 Сотрудник\_ID NUMBER,  
 Отдел NVARCHAR2(100),  
 Дата\_начала DATE,  
 Дата\_окончания DATE,  
 FOREIGN KEY (Сотрудник\_ID) REFERENCES Сотрудники(ID)  
);  
  
CREATE TABLE Повышение\_квалификации (  
 ID NUMBER PRIMARY KEY,  
 Сотрудник\_ID NUMBER,  
 Курс NVARCHAR2(100),  
 Дата\_начала DATE,  
 Дата\_окончания DATE,  
 FOREIGN KEY (Сотрудник\_ID) REFERENCES Сотрудники(ID)  
);  
  
CREATE TABLE Должности (  
 ID NUMBER PRIMARY KEY,  
 Название NVARCHAR2(100),  
 Описание NVARCHAR2(100)  
);  
  
CREATE TABLE Отделы (  
 ID NUMBER PRIMARY KEY,  
 Название NVARCHAR2(100),  
 Описание NVARCHAR2(100)  
);  
  
CREATE TABLE Паспорт (  
 ID NUMBER PRIMARY KEY,  
 сотрудник\_ID NUMBER,  
 Серия NVARCHAR2(50),  
 Номер NVARCHAR2(50),  
 Кем\_выдано NVARCHAR2(100),  
 Дата\_выдачи DATE,  
 FOREIGN KEY (сотрудник\_ID) REFERENCES Сотрудники(ID)  
);

Индексы, последовательности и представления:

-- Для таблицы Сотрудники  
CREATE INDEX idx\_Сотрудники\_Имя ON Сотрудники (Имя);  
CREATE VIEW v\_Сотрудники AS SELECT ID, Имя, Фамилия FROM Сотрудники;  
CREATE SEQUENCE seq\_Сотрудники START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Образование  
CREATE INDEX idx\_Образование\_Уровень ON Образование (Уровень);  
CREATE VIEW v\_Образование AS SELECT ID, Уровень, Специальность FROM Образование;  
CREATE SEQUENCE seq\_Образование START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Страховой\_полюс  
CREATE INDEX idx\_Страховой\_полюс\_Номер\_полюса ON Страховой\_полюс (Номер\_полюса);  
CREATE VIEW v\_Страховой\_полюс AS SELECT ID, Номер\_полюса, Страховая\_компания FROM Страховой\_полюс;  
CREATE SEQUENCE seq\_Страховой\_полюс START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Предыдущие\_места\_работы  
CREATE INDEX idx\_Предыдущие\_места\_работы\_Сотрудник\_ID ON Предыдущие\_места\_работы (Сотрудник\_ID);  
CREATE VIEW v\_Предыдущие\_места\_работы AS SELECT ID, Сотрудник\_ID, Место\_работы, Должность FROM Предыдущие\_места\_работы;  
CREATE SEQUENCE seq\_Предыдущие\_места\_работы START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Льготы  
CREATE INDEX idx\_Льготы\_Сотрудник\_ID ON Льготы (Сотрудник\_ID);  
CREATE VIEW v\_Льготы AS SELECT ID, Сотрудник\_ID, Льгота FROM Льготы;  
CREATE SEQUENCE seq\_Льготы START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Справки  
CREATE INDEX idx\_Справки\_Сотрудник\_ID ON Справки (Сотрудник\_ID);  
CREATE VIEW v\_Справки AS SELECT ID, Сотрудник\_ID, Состояние\_здоровья FROM Справки;  
CREATE SEQUENCE seq\_Справки START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Отпуска  
CREATE INDEX idx\_Отпуска\_Сотрудник\_ID ON Отпуска (Сотрудник\_ID);  
CREATE VIEW v\_Отпуска AS SELECT ID, Сотрудник\_ID, Дата\_начала, Дата\_окончания FROM Отпуска;  
CREATE SEQUENCE seq\_Отпуска START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Командировки  
CREATE INDEX idx\_Командировки\_Сотрудник\_ID ON Командировки (Сотрудник\_ID);  
CREATE VIEW v\_Командировки AS SELECT ID, Сотрудник\_ID, Место, Дата\_начала, Дата\_окончания FROM Командировки;  
CREATE SEQUENCE seq\_Командировки START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Больничные  
CREATE INDEX idx\_Больничные\_Сотрудник\_ID ON Больничные (Сотрудник\_ID);  
CREATE VIEW v\_Больничные AS SELECT ID, Сотрудник\_ID, Дата\_начала, Дата\_окончания FROM Больничные;  
CREATE SEQUENCE seq\_Больничные START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Поощрения\_и\_наказания  
CREATE INDEX idx\_Поощрения\_и\_наказания\_Сотрудник\_ID ON Поощрения\_и\_наказания (Сотрудник\_ID);  
CREATE VIEW v\_Поощрения\_и\_наказания AS SELECT ID, Сотрудник\_ID, Тип, Описание FROM Поощрения\_и\_наказания;  
CREATE SEQUENCE seq\_Поощрения\_и\_наказания START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Передвижения\_по\_должностям  
CREATE INDEX idx\_Передвижения\_по\_должностям\_Сотрудник\_ID ON Передвижения\_по\_должностям (Сотрудник\_ID);  
CREATE VIEW v\_Передвижения\_по\_должностям AS SELECT ID, Сотрудник\_ID, Должность, Дата\_начала, Дата\_окончания FROM Передвижения\_по\_должностям;  
CREATE SEQUENCE seq\_Передвижения\_по\_должностям START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Передвижение\_по\_отделам  
CREATE INDEX idx\_Передвижение\_по\_отделам\_Сотрудник\_ID ON Передвижение\_по\_отделам (Сотрудник\_ID);  
CREATE VIEW v\_Передвижение\_по\_отделам AS SELECT ID, Сотрудник\_ID, Отдел, Дата\_начала, Дата\_окончания FROM Передвижение\_по\_отделам;  
CREATE SEQUENCE seq\_Передвижение\_по\_отделам START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Повышение\_квалификации  
CREATE INDEX idx\_Повышение\_квалификации\_Сотрудник\_ID ON Повышение\_квалификации (Сотрудник\_ID);  
CREATE VIEW v\_Повышение\_квалификации AS SELECT ID, Сотрудник\_ID, Курс, Дата\_начала, Дата\_окончания FROM Повышение\_квалификации;  
CREATE SEQUENCE seq\_Повышение\_квалификации START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Должности  
CREATE INDEX idx\_Должности\_Название ON Должности (Название);  
CREATE VIEW v\_Должности AS SELECT ID, Название, Описание FROM Должности;  
CREATE SEQUENCE seq\_Должности START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Отделы  
CREATE INDEX idx\_Отделы\_Название ON Отделы (Название);  
CREATE VIEW v\_Отделы AS SELECT ID, Название, Описание FROM Отделы;  
CREATE SEQUENCE seq\_Отделы START WITH 1 INCREMENT BY 1;  
  
-- Для таблицы Паспорт  
CREATE INDEX idx\_Паспорт\_сотрудник\_ID ON Паспорт (сотрудник\_ID);  
CREATE VIEW v\_Паспорт AS SELECT ID, сотрудник\_ID, Серия, Номер, Кем\_выдано, Дата\_выдачи FROM Паспорт;  
CREATE SEQUENCE seq\_Паспорт START WITH 1 INCREMENT BY 1;

Таблицы и процедуры:

-- Процедура для добавления нового сотрудника  
CREATE OR REPLACE PROCEDURE *ДобавитьСотрудника* (  
 Имя IN NVARCHAR2,  
 Фамилия IN NVARCHAR2,  
 Образование\_ID IN NUMBER,  
 Семейное\_положение IN NVARCHAR2,  
 Страховой\_полюс\_ID IN NUMBER,  
 Пол IN NVARCHAR2  
) AS  
BEGIN  
 INSERT INTO Сотрудники (Имя, Фамилия, Образование\_ID, Семейное\_положение, Страховой\_полюс\_ID, Пол)  
 VALUES (Имя, Фамилия, Образование\_ID, Семейное\_положение, Страховой\_полюс\_ID, Пол);  
END;  
/  
  
-- Процедура для удаления сотрудника  
CREATE OR REPLACE PROCEDURE *УдалитьСотрудника* (  
 ID IN NUMBER  
) AS  
BEGIN  
 DELETE FROM Сотрудники WHERE ID = ID;  
END;  
/  
  
-- Функция для получения количества сотрудников с определенным образованием  
CREATE OR REPLACE FUNCTION *Количество\_Сотрудников\_с\_Образованием* (  
 Образование\_ID IN NUMBER  
) RETURN NUMBER IS  
 Количество NUMBER;  
BEGIN  
 SELECT *COUNT*(\*) INTO Количество FROM Сотрудники WHERE Образование\_ID = Образование\_ID;  
 RETURN Количество;  
END;  
/