

Листинг кода на языке SQL для создания таблиц

```
CREATE TABLE CategoryClients
(
    CategoryID INT IDENTITY(1,1) NOT NULL PRIMARY KEY,
    CategoryName VARCHAR(100) NOT NULL,
)
GO
```

```
CREATE TABLE StatusClients
(
    StatusID INT IDENTITY(1,1) NOT NULL PRIMARY KEY,
    StatusName VARCHAR(100) NOT NULL,
    StatusDescription VARCHAR(255) NOT NULL
)
GO
```

```
CREATE TABLE Worker
(
    WorkerID INT IDENTITY(1,1) NOT NULL PRIMARY KEY,
    WorkerName VARCHAR(100) NOT NULL,
    WorkerPhone VARCHAR(50) NULL,
    WorkerLogin VARCHAR(50) UNIQUE NOT NULL,
    WorkerPassword VARCHAR(50) UNIQUE NOT NULL,
    WorkerRole VARCHAR(50) NOT NULL
)
GO
```

```
CREATE TABLE RecordedClients
(
    RecClientID INT IDENTITY(1,1) NOT NULL PRIMARY KEY,
    RecClientName VARCHAR(100) NOT NULL,
    RecClientPhone VARCHAR(100) UNIQUE NOT NULL,
    RecClientEmail VARCHAR(50) UNIQUE NOT NULL,
    RecClientStatus VARCHAR(50) NOT NULL
)
GO
```

```
CREATE TABLE PotentialClients
(
    ClientID INT IDENTITY(1,1) NOT NULL PRIMARY KEY,
    ClientName VARCHAR(100) NOT NULL,
    ClientEmail VARCHAR(50) NOT NULL,
    ClientPhone VARCHAR(50) NULL,
    CreationDate DATETIME NULL,
    StatusID INT NULL,
    CategoryID INT NOT NULL,
    WorkerID INT NOT NULL,
    RecClientID INT NULL,
```

```
)  
GO
```

```
CREATE TABLE Documents  
(  
    DocID INT IDENTITY(1,1) NOT NULL PRIMARY KEY,  
    ClientID INT NOT NULL,  
    DocName VARCHAR(50) UNIQUE NOT NULL,  
    DocType VARCHAR(50) NOT NULL,  
    Dateload VARCHAR(50) NOT NULL  
)  
GO
```

```
CREATE TABLE ClientsHistory  
(  
    HistoryID INT IDENTITY(1,1) NOT NULL PRIMARY KEY,  
    RecordDate DATETIME NOT NULL,  
    ResponsibleWorker VARCHAR(100) NOT NULL,  
    FieldChanged VARCHAR(25) NULL,  
    [Value] VARCHAR(100) NULL,  
    ClientID INT NOT NULL  
)  
GO
```

```
ALTER TABLE [dbo].[ClientsHistory] WITH CHECK ADD CONSTRAINT  
[FK_ClientsHistory_PotentialClients] FOREIGN KEY([ClientID])  
REFERENCES [dbo].[PotentialClients] ([ClientID])  
GO
```

```
ALTER TABLE [dbo].[Documents] WITH CHECK ADD CONSTRAINT  
[FK_Documents_PotentialClients] FOREIGN KEY([ClientID])  
REFERENCES [dbo].[PotentialClients] ([ClientID])  
GO
```

```
ALTER TABLE [dbo].[PotentialClients] WITH CHECK ADD CONSTRAINT  
[FK_PotentialClients_CategoryClients] FOREIGN KEY([CategoryID])  
REFERENCES [dbo].[CategoryClients] ([CategoryID])  
GO
```

```
ALTER TABLE [dbo].[PotentialClients] WITH CHECK ADD CONSTRAINT  
[FK_PotentialClients_StatusClients] FOREIGN KEY([StatusID])  
REFERENCES [dbo].[StatusClients] ([StatusID])  
GO
```

```
ALTER TABLE [dbo].[PotentialClients] WITH CHECK ADD CONSTRAINT  
[FK_PotentialClients_Worker] FOREIGN KEY([WorkerID])  
REFERENCES [dbo].[Worker] ([WorkerID])  
GO
```

```
ALTER TABLE [dbo].[PotentialClients] WITH CHECK ADD CONSTRAINT  
[FK_PotentialClients_RecordedClients] FOREIGN KEY([RecClientID])  
REFERENCES [dbo].[RecordedClients] ([RecClientID])  
GO
```

```

CREATE TRIGGER trg_client_history_update
ON PotentialClients
AFTER UPDATE
AS
BEGIN
    SET NOCOUNT ON;
    IF UPDATE(StatusID)
    BEGIN
        INSERT INTO ClientsHistory (RecordDate, FieldChanged, [Value], ClientID,
ResponsibleWorker)
        SELECT
            GETDATE(),
            'StatusID',
            CAST(d.StatusID AS VARCHAR(20)),
            d.ClientID,
            w.WorkerName
        FROM deleted d
        JOIN inserted i ON d.ClientID = i.ClientID
        JOIN Worker w ON i.WorkerID = w.WorkerID
        WHERE d.StatusID <> i.StatusID;
    END
END;
GO

```

```

CREATE TRIGGER trg_PotentialClients_set_creation_date
ON PotentialClients
AFTER INSERT
AS
BEGIN
    SET NOCOUNT ON;

    UPDATE pc
    SET pc.CreationDate = GETDATE()
    FROM PotentialClients pc
    INNER JOIN inserted i ON pc.ClientID = i.ClientID
    WHERE i.CreationDate IS NULL;
END;
GO

```