```
/* БАЗА */

CREATE DATABASE [TASK]
ON PRIMARY
(NAME = N'TASK', FILENAME = N'D:\task9\task9.mdf', SIZE = 51200
KB, FILEGROWTH = 10%),
FILEGROUP GENERAL
(NAME = 'stable', FILENAME = 'D:\task9\stable.ndf', SIZE = 1 MB,
FILEGROWTH = 10%),
FILEGROUP ONLYREAD
(NAME = 'read_only', FILENAME = 'D:\task9\read_only.ndf', SIZE =
1 MB, FILEGROWTH = 10%)
LOG ON
(NAME = N'TASK_log', FILENAME = N'D:\task9\TASK_log.ldf', SIZE =
10240 KB, FILEGROWTH = 10%)
COLLATE Cyrillic_General_100_CI_AI
GO
ALTER DATABASE [TASK] SET RECOVERY SIMPLE WITH NO_WAIT;
GO
ALTER DATABASE [TASK] SET AUTO_SHRINK OFF
GO

/* ТАБЛИЦЫ */

CREATE TABLE [Users] (
     ID int NOT NULL,
     Name varchar(100) NOT NULL,
     Date date NOT NULL,
     Address varchar(100) NOT NULL,
     ID_Country int NOT NULL,
     ID_City int NOT NULL,
     Link varchar(50) NOT NULL,
     Email varchar(50) NOT NULL,
     Raiting float(3) NOT NULL,
     Count_Followers int NOT NULL,
     Count_Likes int NOT NULL,
     Count_Posts int NOT NULL,
  CONSTRAINT [PK_USERS] PRIMARY KEY CLUSTERED
  (
  [ID] ASC
```

```
  ) WITH (IGNORE_DUP_KEY = OFF)

)
GO
CREATE TABLE [Themes] (
     ID int NOT NULL,
     Name varchar(255) NOT NULL,
     Description text NOT NULL,
  CONSTRAINT [PK_THEMES] PRIMARY KEY CLUSTERED
  (
  [ID] ASC
  ) WITH (IGNORE_DUP_KEY = OFF)

)
GO
CREATE TABLE [Comments] (
     ID int NOT NULL,
     ID_User int NOT NULL,
     Date datetime NOT NULL,
     Comment text NOT NULL,
  CONSTRAINT [PK_COMMENTS] PRIMARY KEY CLUSTERED
  (
  [ID] ASC
  ) WITH (IGNORE_DUP_KEY = OFF)

)
GO
CREATE TABLE [Followers] (
     ID int NOT NULL,
     ID_User int NOT NULL,
     ID_Followers int NOT NULL,
  CONSTRAINT [PK_FOLLOWERS] PRIMARY KEY CLUSTERED
  (
  [ID] ASC
  ) WITH (IGNORE_DUP_KEY = OFF)

)
GO
CREATE TABLE [Post] (
     ID int NOT NULL,
```

```
     ID_Users int NOT NULL,
     ID_Themes int NOT NULL,
     Name varchar(100) NOT NULL,
     Description text NOT NULL,
     Data datetime NOT NULL,
  CONSTRAINT [PK_POST] PRIMARY KEY CLUSTERED
  (
  [ID] ASC
  ) WITH (IGNORE_DUP_KEY = OFF)

)
GO
CREATE TABLE [City] (
     ID int NOT NULL,
     Name varchar(100) NOT NULL,
     ID_Country int NOT NULL,
  CONSTRAINT [PK_CITY] PRIMARY KEY CLUSTERED
  (
  [ID] ASC
  ) WITH (IGNORE_DUP_KEY = OFF)

)
GO
CREATE TABLE [Country] (
     ID int NOT NULL,
     Name varchar(100) NOT NULL,
  CONSTRAINT [PK_COUNTRY] PRIMARY KEY CLUSTERED
  (
  [ID] ASC
  ) WITH (IGNORE_DUP_KEY = OFF)

)
GO
CREATE TABLE [Rating] (
     ID int NOT NULL,
     ID_Users int NOT NULL,
     ID_FromWho int NOT NULL,
     Mark int NOT NULL,
  CONSTRAINT [PK_RATING] PRIMARY KEY CLUSTERED
  (
```

```
  [ID] ASC
  ) WITH (IGNORE_DUP_KEY = OFF)

)
GO
CREATE TABLE [Likes] (
     ID int NOT NULL,
     ID_Users int NOT NULL,
     ID_FromWho int NOT NULL,
     ID_Post int NOT NULL,
  CONSTRAINT [PK_LIKES] PRIMARY KEY CLUSTERED
  (
  [ID] ASC
  ) WITH (IGNORE_DUP_KEY = OFF)

)
GO
ALTER TABLE [Users] WITH CHECK ADD CONSTRAINT [Users_fk0]
FOREIGN KEY ([ID_Country]) REFERENCES [Country]([ID])
ON UPDATE CASCADE
GO
ALTER TABLE [Users] CHECK CONSTRAINT [Users_fk0]
GO
ALTER TABLE [Users] WITH CHECK ADD CONSTRAINT [Users_fk1]
FOREIGN KEY ([ID_City]) REFERENCES [City]([ID])
ON UPDATE CASCADE
GO
ALTER TABLE [Users] CHECK CONSTRAINT [Users_fk1]
GO


ALTER TABLE [Comments] WITH CHECK ADD CONSTRAINT [Comments_fk0]
FOREIGN KEY ([ID_User]) REFERENCES [Users]([ID])
ON UPDATE CASCADE
GO
ALTER TABLE [Comments] CHECK CONSTRAINT [Comments_fk0]
GO

ALTER TABLE [Followers] WITH CHECK ADD CONSTRAINT
[Followers_fk0] FOREIGN KEY ([ID_User]) REFERENCES [Users]([ID])
```

```
ON UPDATE CASCADE
GO
ALTER TABLE [Followers] CHECK CONSTRAINT [Followers_fk0]
GO
ALTER TABLE [Followers] WITH CHECK ADD CONSTRAINT
[Followers_fk1] FOREIGN KEY ([ID_Followers]) REFERENCES [Users]
([ID])
ON UPDATE CASCADE
GO
ALTER TABLE [Followers] CHECK CONSTRAINT [Followers_fk1]
GO

ALTER TABLE [Post] WITH CHECK ADD CONSTRAINT [Post_fk0] FOREIGN
KEY ([ID_Users]) REFERENCES [Users]([ID])
ON UPDATE CASCADE
GO
ALTER TABLE [Post] CHECK CONSTRAINT [Post_fk0]
GO
ALTER TABLE [Post] WITH CHECK ADD CONSTRAINT [Post_fk1] FOREIGN
KEY ([ID_Themes]) REFERENCES [Themes]([ID])
ON UPDATE CASCADE
GO
ALTER TABLE [Post] CHECK CONSTRAINT [Post_fk1]
GO

ALTER TABLE [City] WITH CHECK ADD CONSTRAINT [City_fk0] FOREIGN
KEY ([ID_Country]) REFERENCES [Country]([ID])
ON UPDATE CASCADE
GO
ALTER TABLE [City] CHECK CONSTRAINT [City_fk0]
GO


ALTER TABLE [Rating] WITH CHECK ADD CONSTRAINT [Rating_fk0]
FOREIGN KEY ([ID_Users]) REFERENCES [Users]([ID])
ON UPDATE CASCADE
GO
ALTER TABLE [Rating] CHECK CONSTRAINT [Rating_fk0]
GO
```

```
ALTER TABLE [Rating] WITH CHECK ADD CONSTRAINT [Rating_fk1]
FOREIGN KEY ([ID_FromWho]) REFERENCES [Users]([ID])
ON UPDATE CASCADE
GO
ALTER TABLE [Rating] CHECK CONSTRAINT [Rating_fk1]
GO

ALTER TABLE [Likes] WITH CHECK ADD CONSTRAINT [Likes_fk0]
FOREIGN KEY ([ID_Users]) REFERENCES [Users]([ID])
ON UPDATE CASCADE
GO
ALTER TABLE [Likes] CHECK CONSTRAINT [Likes_fk0]
GO
ALTER TABLE [Likes] WITH CHECK ADD CONSTRAINT [Likes_fk1]
FOREIGN KEY ([ID_FromWho]) REFERENCES [Users]([ID])
ON UPDATE CASCADE
GO
ALTER TABLE [Likes] CHECK CONSTRAINT [Likes_fk1]
GO
ALTER TABLE [Likes] WITH CHECK ADD CONSTRAINT [Likes_fk2]
FOREIGN KEY ([ID_Post]) REFERENCES [Post]([ID])
ON UPDATE CASCADE
GO
ALTER TABLE [Likes] CHECK CONSTRAINT [Likes_fk2]
GO


/* ГРУППА READ ONLY */

ALTER TABLE [City] DROP CONSTRAINT PK_CITY WITH (MOVE TO
read_only)
GO
ALTER TABLE [Country] DROP CONSTRAINT PK_Country WITH (MOVE TO
read_only)
GO
ALTER DATABASE TASK MODIFY FILEGROUP [read_only] READ_ONLY


/* СЕНКЦИОНИРОВАНИЕ */
```

```
CREATE PARTITION FUNCTION PartFunctionFactSales_Date (bigint) AS
RANGE RIGHT FOR VALUES (20100101)
CREATE PARTITION SCHEME PartSchFactSales_Date AS PARTITION
PartFunctionFactSales_Date TO ([OLDGENERAL], [GENERAL],
[GENERAL])


/* МЕТОД СКОЛЬЗЯЩЕГО ОКНА */
CREATE PARTITION FUNCTION PartFunctionFactSales_Date (bigint) AS
RANGE RIGHT FOR VALUES (20100101)
CREATE PARTITION SCHEME PartSchFactSales_Date AS PARTITION
PartFunctionFactSales_Date TO ([OLDGENERAL], [GENERAL],
[GENERAL])

CREATE PROCEDURE PR_SLIDINGWINDOW
AS
DECLARE @DAYFORPART_ORDERS VARCHAR(8)
DECLARE @DAYFORPART_ARH VARCHAR(8)
SET @DAYFORPART_ORDERS = CAST((SELECT TOP 1 [value] FROM
sys.partition_range_values
WHERE function_id = (SELECT function_id
FROM sys.partition_functions
WHERE name = 'PartFunctionFactSales_Date')
ORDER BY boundary_id DESC) AS VARCHAR(8))
SET @DAYFORPART_ARH = CAST((SELECT TOP 1 [value] FROM
sys.partition_range_values
WHERE function_id = (SELECT function_id
FROM sys.partition_functions
WHERE name = 'PartFunctionFactSales_Date')
ORDER BY boundary_id ASC) AS VARCHAR(8))
DECLARE @DAY_DT DATE
SET @DAY_DT = DATEADD(YEAR,1,CAST(@DAYFORPART_ORDERS AS DATE))
DECLARE @DAY_ARH DATE
SET @DAY_ARH = DATEADD(YEAR,1,CAST(@DAYFORPART_ARH AS DATE))
ALTER PARTITION SCHEME PARTSCH_ORDERS_DATE
NEXT USED [Frequently]
ALTER PARTITION SCHEME PARTSCH_ARH_DATE
NEXT USED [Fast_Growing]
ALTER PARTITION FUNCTION PARTUNC_DATE()
SPLIT RANGE (CAST(CONVERT(VARCHAR(8),@DAY_DT,112) AS INT))
```

```
ALTER PARTITION FUNCTION PARTUNC_ARH()
SPLIT RANGE (CAST(CONVERT(VARCHAR(8),@DAY_ARH,112) AS INT))
```