1. **Module 09. Indexes, execution plan**
   1. Написать скрипт создания индекса для таблицы dbo.Employees для поля PostalCode (имя индекса должно быть PostalCode). Сделать обязательную проверку при создании индекса на существование данного индекса в таблице.
   2. Написать скрипт, который обновит в поле PostalCode таблицы dbo.Employees все не числоввые символы на любые числовые.
   3. Построить план и оптимизировать запрос, представленный ниже, так чтобы индекс индекс PostalCode работал не по табличному сканированию (Index Scan), а по Index Seek. Необходимо пояснить, почему вы оптимизировали запрос именно так?

SELECT EmployeeID

FROM dbo.Employees

WHERE LEFT(PostalCode, 2) = '98';

* 1. Разобраться с планом запроса, представленного ниже скрипта. Оптимизировать запрос. Пояснить подробно почему вы считаете, что ваш вариант оптимизации наиболее оптимизирует данный запрос и увеличит его быстродействие?

DECLARE @OrderDate DATETIME = N'1996-01-01 00:00:00'

SELECT OrderId = ordr.OrderID,

EmployeeName = ISNULL(empl.FirstName, '') + ' ' + ISNULL(empl.LastName, ''),

CustomerId = ordr.CustomerID,

CompanyName = cust.CompanyName,

ShippedDate = ordr.ShippedDate,

ProductName = prod.ProductName

FROM dbo.Orders ordr

INNER JOIN dbo.[Order Details] ord ON ord.OrderID = ordr.OrderID

INNER JOIN dbo.Products prod ON ord.ProductID = prod.ProductID

INNER JOIN dbo.Customers cust ON ordr.CustomerID = cust.CustomerID

INNER JOIN dbo.Employees empl ON ordr.EmployeeID = empl.EmployeeID

WHERE ordr.OrderDate >= @OrderDate

Для проверки быстродействия необходимо вставить в задействованные таблицы 1000000+ записей.