# Задача 1

1) Кластерный индекс на первичный ключ ID

2) Уникальный некластерный индекс на атрибут Email

3) Некластерный индекс на атрибут TabNum

# Задача 2

Почему поиск с использованием индекса по полю TabNum для данного запроса – это хорошо?

Поиск кортежей происходит по полю TabNum, т.к. поле индексировано скорость поиска должна возрасти

Почему поиск с использованием индекса по полю TabNum для данного запроса – это плохо?

1. По полю TubNum объявлен некластерный индекс, следовательно для получения полей LastName, FirstName и Email придется совершить еще один поиск по кластерному индексу.
2. Поле TubNum не уникально, следовательно поле TubNum может иметь высокую плотность (физически может, логически не должна т.к. как я понял TubNum – это табельный номер)

Что надо сделать, чтобы запрос с использованием индекса гарантированно давал преимущества по сравнению с неиспользованием индекса?

1. Создать покрывающий индекс для поля TubNum который будет содержать поля LastName, FirstName и Email
2. Добавить полю TubName Ограничение UNIQUE

# Задача 3

Индекс никак не повлияет на выполнение запроса, т.к. нам необходимо вычислить год у поля DateOfBrith в каждом картеже.

Можно переписать запрос для поиска не по части даты а по диапазону

SELECT COUNT(\*)

FROM Employees

WHER DateOfBirth BETWEEN ‘1998-01-01’ AND ‘1998-12-31’

# Задача 4

Скорость выполнения запроса возрастет, но для увеличения скорости можно выполнить следующее

CREATE INDEX idx\_Employees\_LastName\_DateOfBirth

ON Employees (LastName, DateOfBirth DESC)

И переписать запрос:

SELECT TOP 1 \*

FROM Employees

WHERE LastName = N'Иванов'

Или создать индекс от поля LastName, что тоже увеличит скорость выполнения:

CREATE INDEX idx\_Employees\_LastName

ON Employees (LastName)

# Задача 5

Скорость запроса вырастет при наличии индекса idx\_Employees\_LastName\_DateOfBirth, но лучше будет создать индекс только от поля DateOfBirth

CREATE INDEX idx\_Employees\_DateOfBirth

ON Employees (DateOfBirth)

# Задача 6

Скорее всего никак так как поле Gender может иметь высокую плотность СУБД не будет использовать, также индекс и не будет использован при GROUP BY, так что индекс не повлияет на скорость выполнения запроса ни в первом ни во втором случае