**Работа с паролями при аутентификации:**

В базе данных**, всегда должен быть зашифрованный пароль**. Т.e никто, в том числе тот человек, у которого есть прямой доступ к базе данных, не должен видеть пароли пользователей.

**Логика**: Клиент вводит пароль, этот пароль шифруется и сравнивается с зашифрованным паролем в базе данных. Сравнение двух зашифрованных паролей происходит каждый раз.

Чтобы посмотреть все encoder, которые будут шифровать пароли нужно зайти в класс PasswordEncoderFactories. По умолчанию используется BCryptPasswordEncoder. (префикс noop означает, что мы не хотим использовать методы шифрования, а значит пароли будут хранится так как они есть БЕЗ ШИФРОВАНИЯ)

PasswordEncoderFactories не является бином, а значит, если мы хотим настраивать шифрование вручную, то нам нужно создать бин, который будет инициализировать PasswordEncoderFactories.

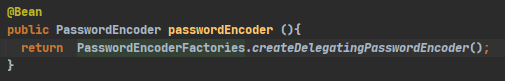
**Реализация:**

1. Создать в createDto/entity поле password. В createDto мы должны установить над полем

Password @NotBlank (groups = Create Actions.class) (подробнее об этом в директории валидации)



1. Так как PasswordEncoder не является бином, а мы хотим, чтобы он был именно бином, томы должны в том конфиге, где мы реализовали UsernamePasswordFilter создать бин, который будет возвращать PasswordEncoder.



1. Мы должны подправить наш userCreateMapper

