1. Да, читал, в основном код библиотек. Когда я делал бота для телеграмма, не получалось подключаться к прокси, поэтому надо было разобраться в устройстве его работы в этой библиотеки. Также мой одногруппник в течение года делает игровой движок на C++(<https://github.com/Cvostr/zsgamedit> , <https://github.com/Cvostr/zspirengine>), я решил ему помогать. Для этого пришлось покопаться в коде. Разбираться в чужом коде всегда очень интересно и главное-полезно, вне зависимости от того, хороший это код или плохой, всегда можно подчерпнуть полезные приемы и как надо писать(и как не надо)).

2. C#, C++, Java, Python - crossplatform;

Kotlin - Mobile;

Haskell, Scala - server; Javascript - frontend;

3.Компилятор служит для того, чтобы переводить высокоуровневые программы в понятный машине код. Помимо компиляторов для этой цели есть еще и интерпретаторы, главное их отличие в том, что компиляторы перед запуском и переводом программы проверяют весь код на ошибки в синтаксисе, а интерпретаторы запускают код по одной строчке. На мой взгляд, компиляторы тут удобнее(особенно когда пишешь длинный парсер и не ожидаешь ошибки, у меня был парсер на Python’e в 400 строк, который пришлось долго отлаживать в т.ч. и из-за такого способа “перевода” программы). Также компилируемые языки(С++, С#) в разы быстрее интерпретируемых(Javascript, Perl, Python).

4.Фреймворк - это не просто библиотека или их набор. Фреймворк - это целый ряд инструментов и шаблонов, которые позволяют значительно ускорить разработку и облегчить жизнь программисту. Вот примеры фреймворков:

* .NET Framework(Core - crossplatform), WPF, Xamarin, Entity - C#;
* Django, Flask - Python;
* Qt - Multilanguage(C++, Python e.t.c);

5. Для более безопасной передачи данных по протоколу http используется зашифрованное соединение, наличие шифрования ssl подтверждает сертификат, https - это более безопасное расширение протокола http (s - secure).

6. Я бы разбил страницу ВК так:

1 - раздел со стеной;

2 - раздел с фото;

3 - раздел с друзьями(располагался бы сбоку в виде списка, который можно развернуть, а не отдельной страницей)

7.SQL - давняя, проверенная временем технология, которая смогла себя зарекомендовать как надежная, также за долгое время для нее появилось много стандартов. Поэтому компании, такие как Oracle, Teradata, Microsoft используют SQL в своих продуктах.

Однако в таких вещах как масштабируемость и скорость выигрывают NoSQL системы.

Каждый тип БД используется в своих задачах, на сегодняшний день многие компании даже разрабатывают средства для совмещения обоих подходов.

8. Я занимаюсь разработкой клиента для ВК, который шифрует сообщения, для шифрования используются Camellia (симметричный алгоритм, понравился больше чем AES), а также RSA(для передачи симметричного ключа). Пока сделал только прием сообщений и картинок, в дальнейшем планирую сделать беседы с шифрованием.

<https://github.com/kachan28/VKCrypto> - мой клиент для вк(на сайте также написано - <https://kachan28.github.io/> ), сейчас занимаюсь переорганизацией кода, переписываю с обычных запросов на longpoll, распихиваю функционал по классам, уже перенес класс криптографии в отдельный раздел. Технология WPF. Двухфакторки пока нет.

<https://github.com/kachan28/VK-crypto-chat/blob/master/NewRel.cs> - тот же самый клиент, только на .NET Core(поэтмоу консольный), есть двухфакторка.

9. <https://pastebin.com/ujstGrwH>