Вопрос №1  
Читали ли вы чужой код? Если да, то что это был за код? Опишите свои впечатления от него.

--Читал, по работе иногда приходится поправлять ошибки в уже сделанных до меня скриптах. Также в университете приходилось изучать код одногруппников. В разных случаях был разный код – где-то реализованная бизнес-логика, где-то низкоуровневый язык на C(в момент разработки кода для микроконтроллера), где-то изучал как тот или иной человек реализовал ту или иную функцию. Трудно говорить о впечатлениях, могу заметить, что код бывает по разному оформлен, закоментирован. Также видел, как иногда теряется читаемость кода в угоду быстроте разработки\выполнения кода.

Вопрос №2  
Опишите известные вам языки программирования и их предназначение.

--C++,C,C#,F#,MatLab, PHP, JavaScript, Python, Pascal, Delphi, Assembler, VB, Java, Ruby, Swift, X++, E-Script.

Категоризировать языки можно по многим категориям, к примеру – Высокоуровневые(C++,C,C#) и низкоуровневые(Assembler). Объектно-ориентированные языки(С++, C#, Delphi, JavaScript, Java, Ruby, Swift) или процедурные (PHP, Pascal, C) и тд.

Определенные языки служат определенным целям, к примеру MatLab или MatCad в основном используются для расчета математических и физических формул. Языки такие как PHP – для бэкенда сайта, а JavaScript – для фронтэнда, хотя На данный момент JavaScript стал уже универсальным языком, который используется во многих сферах. Также универсальные языки- C++,C#, Java, Python – все зависит от подключаемых к ним библиотек и используемых фреймворков.

Вопрос №3  
Что такое "компилятор", зачем он нужен и почему некоторые языки обходятся без него?

--Когда мы пишем ту или иную высокоуровневую программу, на самом деле, мы пытаемся обратиться к процессору с определенными инструкциями. На прямую мы не имеем права обращаться, существует определенная операционная система, которая служит некоторой прослойкой между нашими желаниями и процессором. Поэтому существуют компиляторы и интерпретаторы, чтобы преобразовать наш высокоуровневый код в команды, понятные нашей ОС и конкретно нашему процессору, ведь у каждого процессора своя реализованная логика и свои команды, по которому он работает.

Отличие компиляторов и интерпретаторов в том, что компилятор сразу преобразует написанный код в исполняемый под определенную платформу, что быстрее относительно интерпретатора, но теряет кроссплатформенность. А интерпретатор построчно считывает код и интерпретирует его realtime.

Вопрос №4  
Что такое "фреймворк" и для чего он нужен? Приведите примеры известных вам фреймворков.

--Набор библиотек и\или средств разработки, написанные сторонним человеком\компанией для ускорения разработки определенных программ, служит еще одним уровнем абстракции над языком. К примеру – NodeJS, Django, ASP.NET, Bootstrap, ADO.net

Вопрос №5  
Что за приставка "http://" перед адресами сайтов и почему она всё чаще теперь становится "https://"?

Это обозначение использования протокола http прикладного уровня. Служит для соединения сервера и клиента и имеет некоторые функции такие как GET, POST, DELETE и тд.

https – тот же что и http, только в добавок поддерживает современное шифрование, к примеру ssl

Вопрос №6  
Самая популярная библиотека для разработки фронтенд-приложений, ReactJS, моделирует логику в виде компонентов. Если бы вам нужно было на ReactJS разработать страницу профиля ВКонтакте (например, <https://vk.com/id1>), на какие компоненты вы бы её разбили? Почему именно так?

--Не сталкивался с глубокой разработкой на фронтенде к сожалению. На данный вопрос не могу ответить.

Вопрос №7  
SqlServer, PostgreSQL, SQLLite, MySQL, Oracle, Microsoft Access - разные базы данных с разным функционалом, которые разрабатываются, в основном, разными компаниями с разным видением своего продукта.  
Однако все эти базы используют один и тот же язык запросов - SQL, и не планируют от него отказываться. Как так получается? Что такого в SQL, что он подходит всем этим базам?  
А если он такой чудесный, то почему многие другие базы данных, вроде MongoDB или Cassandra, его не используют?

--Использование многими компаниями одного языка позволяет перенести готовое решение на бд другой компании. Также вроде как SQL признан международным стандартом. К тому же, написано множество стороннего ПО для взаимодействия с SQL.

Скорее всего, некоторые компании не используют SQL, потому что во первых, на рынке должна существовать альтернатива, во-вторых это попытка решить некоторые проблемы SQL, такие как масштабируемость и удобство. Если есть большое количесвто однотипной информации, то лучше использовать NoSql

Вопрос №8  
Пришлите ссылку на пример вашего кода на C#, за который вам не стыдно. Если кода нет, выполните задание ниже. Оно также поможет, если код есть.  
Пожалуйста, не пишите код внутри форм ответов, разместите его на Github и приложите ссылку. Если в задании что-то непонятно, опишите возникшие вопросы и сделанные предположения. Например, в комментариях в коде.  
Задание:  
Напишите библиотеку для поставки внешним клиентам, которая умеет вычислять площадь круга по радиусу и треугольника по трем сторонам. Дополнительно к работоспособности оценим:

* Юнит-тесты
* Легкость добавления других фигур
* Вычисление площади фигуры без знания типа фигуры
* Проверку на то, является ли треугольник прямоугольным

--Добавил ссылку на часть своего проекта –

<https://github.com/gagurolle/ByBitBotTest>

Вопрос №9  
В базе данных MS SQL Server есть продукты и категории. Одному продукту может соответствовать много категорий, в одной категории может быть много продуктов.  
Напишите SQL запрос для выбора всех пар «Имя продукта – Имя категории». Если у продукта нет категорий, то его имя все равно должно выводиться.

Product -- | id | name | category\_id |

Category -- | id | name |

SELECT p.name, c.name FROM product p LEFT JOIN category c ON p.category\_id = c.id