

Q1

1. Python: библиотеки Pygame и NumPy, модули Itertools, Collections, random, math и тд. На Ruby работаю пока мало, конкретно сказать ничего не могу.
Для себя могу выделить удобство использования готовых решений из библиотек и модулей, что позволяет сосредоточиться на общей картине(концепте) программы, нежели на мелких рутинных деталях исполнения. Кратко – сторонними библиотеками и модулями повышаю уровень абстракции проектов.
2. Linux, Windows.
Работал: Raspbian Jessie (Debian Jessie), Bash встроенная в Windows, Git Bash.
Утилиты: Screen, Wget, cmp.
3. Прохожу несколько курсов одновременно на Coursera, EdX, решаю алгоритмические задачи на HackerRank, CoddeWars, участвую в BootCamp'ах, посещаю конференции и хакатоны.

Q2

1. Пока только встроенные в различные IDE, такие как PyCharm, RubyMine, Anaconda. Мои знания недостаточно глубоки в этой области, чтобы отметить какие-то явные минусы.
2. Первый случай: 1660 и 100 оба целочисленные (integer), соответственно и деление происходит целочисленное. В Python аналог – `“//”`. Решается приведением одного из чисел к типу с плавающей точкой (float). $1660.0 / 100 = 16.6$ or $1660 / 100.0$
Второй случай – известное ограничение арифметики с числами с плавающей точкой (float), где ограниченное количество битов может воспроизвести только ограниченное количество цифр после запятой, которых математически может быть бесконечно (3.3333333 и тд).

Q3

Во-первых, использование https и более строгих SSL-сертификатов. Даже в выданном мне задании со скраппингом это доставило ряд хлопот, вынудив использовать не безопасные (curl -k) GET запросы, которые в ряде случаев могут банально блокироваться на стороне сервера.

Во-вторых, можно на серверной части мониторить сами GET/POST методы для фильтрации известных способов скраппинга через них. Сюда же относиться и защита с помощью htaccess файлов и проверка хедеров (headers) (User-Agent, Cookie)

В-третьих, можно ограничить доступ по индивидуальным IP. Минус тут – конкуренты могут использовать ряд VPN для изменения IP.

В-четвёртых, уже более безумная идея – конвертировать цены в маленькие изображения этих цен, что предотвратит лёгкий доступ к численному значению цены. Здесь конкуренты могут использовать image processing, но это достаточно ресурсозатратно и не каждый это будет использовать. Как развитие идеи использования изображений – добавление CAPTCHAs в ключевых местах сайта.