1. Какие классы содержаться с System.IO?

Классы, расположенные в этом пространстве имен (такие как Stream, StreamWriter, FileStream и др.), позволяют управлять файловым вводом-выводом.

2. Для чего используются классы Directory и DirectoryInfo? В чем отличие?

Directory – статический, DirectoryInfo – нет, функционал почти одинаковый. Если нужно использовать пару функций, лучше использовать Directory, если постоянная и долгая работа с одним каталогом, то DricetoryInfo, так как у него нету дополнительных проверок безапосности.

3. Для чего используются классы File и FileInfo? Какие методы они содержат.

Удалять, создавать, перемещать файлы, получить их свойства, проверка на наличие и тд.

4. Для чего используются классы StreamReader и StreamWriter?

Первый используется для чтения файлов (указывая путь и кодировку, чтение посимвольно, построчно или весь файл сразу), второй для записи в файл (создает или записывает в уже существующий файл, указывая путь, перезаписывая полностью либо продолжая в конце с указанной кодировкой)

5. Для чего используются классы BinaryWriter и BinaryReader?

Записывают и считывают бинарные файлы, при считывании явно указываешь какой тип считываешь)

6. Как можно сжимать и восстанавливать файлы?

Это классы ZipFile, DeflateStream и GZipStream, которые находятся в пространстве имен System.IO.Compression и представляют реализацию одного из алгоритмов сжатия Deflate или GZip.

7. Расскажите алгоритм сжатия GZip.

Работает на основе алгоритма Хаффмана.

8. Для чего служит класс Path?

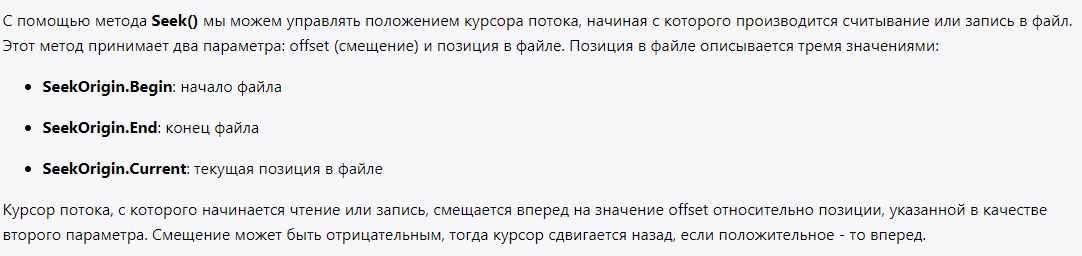
Для удобной работы с путями к файлам, создает пути из двух строк, понимает разные форматы записи пути.

9. Что такое тестовый и бинарный файл. Как можно выполнить чтение и запись в эти файлы.

StreamReader и StreamWriter для текстовых, BinaryWriter и BinaryReader для бинарных.

10.Что такое произвольный доступ к файлу? Приведите пример.

Доступ к произвольному месту в файле с помощью метода Seek.



11.Как применяется конструкция using (не директива) при работе с файловыми потоками? Для чего ее используют.

При работе с файловыми потоками класса FileStream, при создании или открытии файла мы автоматически открываем поток, чтобы его закрыть нужно вызывать метод Close(). Если использовать using, он автоматически закроет поток при выходе из конструкции.