

Итак, привет, кандидат в внедренцы Диадока. Чтобы попасть к нам на работу, помимо всего прочего, нужно выполнить тестовое задание. Его суть заключается в том, чтобы понять, как ты будешь справляться с нашими основными задачами.

Все, с чем мы работаем, можно разбить на три большие категории:

- 1) Разработка печатных форм (она ведется в инструменте FastReport). Мы не требуем знания принципов работы этого инструмента. Тем не менее, поскольку в нем используется синтаксис С#, базовые знания по С# потребуются.
- 2) Преобразования одних XML файлов в другие XML файлы. Для этого используется XSL-преобразования (версия 1.0).
- 3) Работа с публичным API. Нужно будет хоть немного понимать, как работает HTTP.

Задания по С#

Есть список данных (`<List>`), заполненный датами.

Например:

```
{  
  03.09.2019  
  11.06.1990  
  03.09.2019  
  31.12.2018  
  09.05.2017  
}
```

- 1) Нужно узнать количество элементов в этом списке.
- 2) Мне нужно получить другой список, который будет содержать только уникальные даты в порядке возрастания. Причем, я хочу, чтобы дата была в формате «03 сентября 2019 года».
- 3) Возьмем тот же список с датами из п.1. Мне хочется узнать, сколько Ticks в каждой дате (считаем, что 03.09.2019 = 03.09.2019 00:00:00.000).

В всех случаях я хочу увидеть рабочий метод, написанный на С#, который на вход принимает список и на выходе выдает список (или целое число, отражающее количество элементов в списке).

Задания по XSLT

При работе с XSLT крайне важно понимать, как устроены XML-документы. Для этого используется синтаксис XPath-выражений, который позволяет обращаться к узлам и атрибутам XML-документа, а также проверять их значения.

Есть у меня такой XML-файл:

```
<Table>
  <Item>
    <Type>A</Type>
    <Product>KETTLE</Product>
    <Quantity>31</Quantity>
    <RepairSum>7153.07</RepairSum>
    <DetailSum>12659.52</DetailSum>
    <Subtotal>19812.59</Subtotal>
  </Item>
  <Item>
    <Type>B</Type>
    <Product>ESPRESSO FULL AUTOM</Product>
    <Quantity>15</Quantity>
    <RepairSum>19830.56</RepairSum>
    <DetailSum>1021.12</DetailSum>
    <Subtotal>20851.68</Subtotal>
  </Item>
  <Item>
    <Type>C</Type>
    <Product>Мясорубки</Product>
    <Quantity>8</Quantity>
    <RepairSum>1222.72</RepairSum>
    <DetailSum>0</DetailSum>
    <Subtotal>1222.72</Subtotal>
  </Item>
  <TotalQuantity>635</TotalQuantity>
  <TotalRepairSum>202356.88</TotalRepairSum>
  <TotalDetailSum>96088.55</TotalDetailSum>
  <Total>298445.43</Total>
</Table>
```

Я хочу получить из этого файла другой файл:

```
<Result>
  <DocumentInfo NumberOfItems="{число элементов Item в исходном файле}">
    <Total>{содержимое узла Total в исходном файле}</Total>

    <RealTotal>{сумма узлов Subtotal в каждом из узлов Item исходного
файла}</RealTotal>
    <IsEqual>{здесь я хочу увидеть true или false в зависимости от того, совпадает ли
содержимое Total и RealTotal в преобразованном документе}</IsEqual>

    {Ниже я хочу получить все различные значения узлов Type исходного файла, т.е.
подразумевается, что преобразование будет работать, если я поменяю значение одного из этих
узлов или добавлю новые}
    <DifferentTypes>
      <Type>A</Type>
      <Type>B</Type>
      <Type>C</Type>
    </DifferentTypes>
  </DocumentInfo>
</Result>
```

Сложность заданий: красное – «Пешка», голубое – «Ладья», зеленое – «Ферзь».

Задания с HTTP-запросами

Иногда нам нужно консультировать клиентов по вопросам работы с API. Фактически нужно уметь самому отправлять запросы в API и понимать, что возвращается в ответ.

Не важно, каким способом, но мне нужно получить информацию, которая возвращается при простом входе из браузера на страницу <https://lenta.ru>:

- 1) Какой тип запроса при этом уходит на сервер? Чем этот отличается от других?
- 2) Какой код ответа возвращает сервер? Какие коды ты еще знаешь?
- 3) Что такое cookie? Устанавливает ли сервер какие-либо cookie при входе на lenta.ru?
- 4) За какое время сервер обработал запрос?

Подсказка: для сбора логов HTTP можно использовать Fiddler либо встроенные инструменты браузеров.