Тестовое задание на должность младшего разработчика Java

# Общее описание

Требуется разработать java-приложение, которое на вход принимает название xml-файла, в котором хранится список различных геометрических фигур и их параметры. Приложение должно на основе данных из файла вывести площади этих фигур

# Общие требования

1. Приложение должно быть написано на Java версии не ниже 6
2. Приложение не должно использовать сторонние библиотеки
3. Приложение должно быть собрано в виде одного запускаемого jar-файла
4. Приложение должно запускаться из командной строки.
5. Формат запуска приложения:

***java –jar <jar-name> <file-name>***

где

1. ***<jar-name>*** - путь к запускаемому jar-файлу приложения,
2. ***<file-name>*** - путь к xml-файлу с входными данными.

Пример: ***java –jar app.jar text.xml***

# Формат входных данных

Входной xml-файл удовлетворяет приложенной xsd-схеме.

Пример входного файла:

<**shapes**>  
 <**triangle**>  
 <**color**>red</**color**>  
 <**side**>1.5</**side**>  
 <**side**>1.5</**side**>  
 <**side**>1.5</**side**>  
 </**triangle**>  
 <**circle**>  
 <**color**>orange</**color**>  
 <**diameter**>1.5</**diameter**>  
 </**circle**>  
</**shapes**>

# Формат выходных данных

По мере обработки информации о фигуре приложение должно печатать в стандартный вывод строку следующего формата:

***<i>: <color> - <area>\n***

где

1. ***<i>*** - порядковый номер обрабатываемой фигуры
2. ***<color>*** - значение элемента *<color>* описания фигуры
3. ***<area>*** - площадь фигуры, округлённая до сотых, в формате *#.##*

# Требования к поставке

Тестовое задание должно быть предоставлено в виде

1. Исходных кодов
2. Проекта (в случае использования инструментов для автоматизации сборки)
3. Инструкции по сборке
4. Собранного jar-файла

# Приветствуется

1. Использование инструментов для автоматизации сборки приложения. Например: maven, ant, gradle, etc
2. Наличие unit-тестов
3. (\*если предложенное задание кажется чересчур лёгким) Поддержка работоспособности приложения в случае очень большого входного файла (более 1Гб)
4. (\*если предложенное задание кажется чересчур лёгким) Разделение обработки чтения файла и вычисления площади в параллельные, независимые потоки.