**Внимание:**

**Данная задача преподом придумалась только что, и еще не решалась… Будьте осторожны – может она и не решается совсем… ☺**

**Инструменты сборки приложений.**

**Задание-1.**

**Сборка приложения c Apache ANT.**

Дамы и господа, в рамках этого задания мы научимся собирать наши проекты при помощи инструмента Apache ANT. Так как основной задачей будет знакомство с ANT-ом, необходимо максимально снизить количество программирования, дабы не распылять свое внимание и ресурсы головного мозга. Поэтому в этом задании мы будем использовать исходный код задачи №1 из темы «Servlet API». (И вот тут люди, которые уже сдали эту задачу по сервлетам, радостно кричат – «Тряяяяяяху!» ☺).

Как Вы помните, в задаче по сервлетам мы должны были разработать Web-приложение, предоставляющее пользователям возможность заполнить анкету о качестве образования и посмотреть результаты анкетирования. Но если мы полностью перенесем условие задачи на текущее задание, то все сведется к банальному использованию ANT-овского Task-а «war», на работу Вы потратите не более десяти минут, и никакого морального удовлетворения от проделанного труда не получите.

Следовательно, мы должны внести некоторые корректировки в условия, которые сделают ее более интересной. Эти корректировки будут следующими:

1. Главное приложение будет называться **Questionnaire**, что, как оказалось, в переводе означает «Анкета».

2. Приложение должно содержать ANT-файл сборки, который будет располагаться в корневом каталоге.

3. Однако само приложение **Questionnaire** не должно содержать исходного кода, а вместо этого, должно представлять собой родительский проект для трех следующих модулей:

* **Questions** – отдельное приложение, которое должно собираться в WAR файл и отображать пользователю форму для ввода данных.
* **Answers** – отдельное приложение, которое должно собираться в WAR файл и отображать пользователю форму с результатами его ввода в приложении Questions.
* **Utils** – отдельное приложение, которое должно собираться в JAR файл и содержать класс, единственный метод которого должен вернуть текущую версию приложения-анкеты.

И вот тут Вы можете подумать:

«Тысяча, чертей! Программировать все-таки придется!!!»

Пожалуйста, не волнуйтесь. Эту самую сложную часть работы я взял на себя ☺. Рядом с текстом задания Вы сможете найти Eclipse-проект, который уже содержит необходимый класс с полностью (я еще раз подчеркиваю – ***полностью***) запрограммированным методом ☺.

Так что тут дело за малым – извлечь класс из Eclipse-проекта, интегрировать его в структуру каталогов своего проекта, написать для него файл сборки и использовать как JAR библиотеку в двух оставшихся проектах.

Что ж, продолжаем с дополнительными условиями:

4. Каждый из описанных выше модулей должен иметь свой собственный файл сборки.

5. JAR файл получившийся поле сборки приложения Utils должен использоваться как 3-rd party library ***в обоих*** WAR файлах - т. е. Необходимо будет уложить этот JAR файл в CLASS PATH обоих WAR файлов.

6. Оба WAR файла должны отображать версию анкеты на своих страницах - этим Вы докажете то, что действительно используете разработанный Вами API, а не просто кладете его мертвым грузом в CLASS PATH.

7. Выше мы уже сказали, что каждый модуль должен обладать своим собственный сборочным файлом, однако обратите особое внимание на то, что

Запускаться сборка должна из главного сборочного файла приложения Questionnaire, и все три модуля должны быть собраны в правильной последовательности.

Уверен, Вы уже догадались, что вопрос с «правильной последовательностью» строится в основном вокруг приложения **Utils**, которое требуется для корректной сборки двух оставшихся модулей.

8. В присланном Вами архиве не должно быть упоминаний о какой-либо IDE, то есть, хотя бы раз, но мы все-таки поработаем с блокнотом. Тем более что весь исходный код уже был разработан и протестирован Вами ранее.

Ну вот, вроде, и все дополнительные требования. Друзья, описанная задача, несомненно, является учебной. Однако обратите внимание на то, что в реальной работе, Вы, скорее всего, будете действовать именно в такой стратегии – весь Ваш проект будет разбит на отдельные модули, а задача основного проекта (его иногда еще называют «родительским объетом») будет заключаться в том, чтобы следить за корректной последовательностью сборки этих модулей.

Ну что же ­– приятных Вам сборок и побольше BUILD SUCCESS-ов! ☺