

Задание для учебной практики – 2022

Создание консольного приложения на OSGi

Технологии

- Java SE 1.8+
- OSGi
- Apache Felix
- Apache Maven

Цель задания

Познакомить студента с базовыми концепциями OSGi, попробовать эти концепции на примерах и с их использованием создать небольшое консольное приложение (этап 5).

Описание

Практика состоит из 5 этапов. На первом этапе требуется познакомиться с OSGi, Apache Felix. На 2-4 этапах требуется создать несколько простых приложений, чтобы на практике познакомиться с основными концепциями. На 5-м этапе требуется создать небольшое, но полноценное приложение, применив знания и умения, полученные на этапах 1-4.

Этапы выполнения задания

Этап 1. Подготовительный

1. Прочитайте главы 1-4 «OSGi in Action». Постарайтесь понять основные концепции OSGi: modularity, lifecycle, bundles, services, components, пр.
2. Скачайте и установите [Apache Felix](#). Установите Felix Web Console.
3. Запустите Apache Felix в режиме debug (`java -Xdebug -Xrunjdwp:transport=dt_socket,server=y,suspend=n,address=<port-number> -jar bin/felix.jar`)

Этап 2. Реализация OSGi-сервиса

1. Создайте бандл, содержащий сервис (интерфейс с одним методом и его реализацию), зарегистрируйте сервис, используя активатор бандла. Метод должен выводить на консоль сообщение «Hello OSGi World!»
2. Установите бандл в Felix.
3. Создайте новый бандл. Бандл должен потреблять созданный ранее сервис. Метод сервиса, созданного на 1 шаге, должен вызываться при активации нового бандла.
4. Установите бандл в Felix. Убедитесь, что метод сервиса был вызван.

Этап 3. Apache Felix Service Component Runtime

Аналогично этапу 2, создайте два бандла: который содержит реализацию сервиса и который использует этот сервис. Используйте [Apache Felix Service Component Runtime](#), [SCR Annotations](#) и [Apache Felix Maven SCR Plugin](#) для сборки.

Этап 4. Создание собственной команды для Apache Felix Gogo

Требуется создать собственную команду «practice:hello» с одним параметром, которая при вызове печатает на консоль «Hello, <param>», где <param> - введенный пользователем параметр.

См. <http://coders-kitchen.com/2012/04/06/tutorial-an-own-gogo-shell-command/>

Этап 5. Создание приложения

Требуется создать приложение, которое получает через API новостных порталов список актуальных новостей и выводит на консоль 10 самых часто встречающихся слов из заголовков новостей. Данные берутся из следующих источников:

1. Lenta.ru API (<https://lenta.ru/rss/news>)
2. RSS АиФ (<https://www.aif.ru/rss/news.php>)
3. Прочие источники на усмотрение студента.

Процесс подсчета инициируется пользователем приложения с помощью консольной команды «news:stats». Пользователь может передать источник в качестве параметра. Если команда вводится без параметров, пользователю предлагается выбрать источник данных (один из доступных в системе, или все сразу). Список источников выводится на консоль.

Если пользователь выбирает незарегистрированный (недоступный) источник данных, в системе не зарегистрировано ни одного источника, или сеть недоступна, пользователю выводится соответствующее сообщение.

Реализация получения данных из каждого источника должна быть отдельной имплементацией общего для всех источников сервиса (интерфейса).

Рекомендация: сервисы доступа к данным из различных источников реализуйте в разных бандлах

Общие требования для всех этапов

1. Следуйте [Java Code Conventions](#) (обязательное условие для получения зачета)
2. Система сборки – Maven
3. IDE – IntelliJ IDEA или Eclipse

Предоставление результатов

- Результатом каждого этапа (кроме 1-го) должен быть архив с проектом (либо – ссылка на проект на Github) и собранный(ые) бандлы (jar-файлы).
- Проект и бандлы должны иметь префикс “фио_”, образованный из начальных (строчных) букв фамилии, имя, отчества студента.

- Бандлы должны без ошибок устанавливаться и активироваться в Felix 6.0.1.

Отчет по практике должен содержать:

1. Информацию о студенте (ФИО, группа,)
2. Описание результатов по каждому этапу
3. Выводы

Описание по каждому этапу должно содержать:

1. Описание всех классов и пакетов
2. Описание бандлов. Для каждого бандла описать сервисы и зависимости на другие бандлы
3. Ссылка на архив с исходным кодом или ссылка на открытый git репозиторий
4. Ссылки на собранные бандлы

В выводах требуется рассказать, какие преимущества/недостатки имеет OSGi. Привести примеры систем, которые целесообразно/нецелесообразно реализовывать на OSGi. Какими свойствами/ограничениями должны обладать такие системы.

Документация/литература

- “OSGi in Action: Creating Modular Applications in Java” by Richard S. Hall, Karl Pauls, Stuart McCulloch, and David Savage; Manning Publications Co
- [Apache Felix Documentation](#)
- <http://www.javaworld.com/article/2077837/java-se/java-se-hello-osgi-part-1-bundles-for-beginners.html>
- <http://www.vogella.com/tutorials/OSGi/article.html>
- [Google](#)