

Литература

1 Habrahabr [Электронный ресурс]. – 2015. – Режим доступа: <http://habrahabr.ru/post/231855/>. – Дата доступа: 13.04.2015..

2 Front-end и Back-end разработка [Электронный ресурс]. – 2012. – Режим доступа: <http://konservs.com/it/web/front-and-back-end-61>. – Дата доступа: 13.04.2015.

И.С. Шестаков (УО «ГГУ им. Ф. Скорины», Гомель)

Науч. рук. **В.С. Давыдов**, канд. техн. наук, доцент

ИСПОЛЬЗОВАНИЕ JAVA BEANS ПРИ СОЗДАНИИ WEB-ПРИЛОЖЕНИЯ ПО УЧЕТУ ИСКЛЮЧИТЕЛЬНЫХ СИТУАЦИЙ В ИНФОРМАЦИОННОЙ СИСТЕМЕ

JavaBeans – классы в языке Java, написанные по определённым правилам. Они используются для объединения нескольких объектов в один (англ. Bean – фасоль, кофейное зерно) для удобной передачи данных.

Спецификация Sun Microsystems определяет JavaBeans как повторно используемые программные компоненты, которыми можно управлять, используя графические конструкторы и средства IDE.

JavaBeans обеспечивают основу для многократно используемых, встраиваемых и модульных компонентов ПО. Компоненты JavaBeans могут принимать различные формы, но наиболее широко они применяются в элементах графического пользовательского интерфейса. Одна из целей создания JavaBeans – взаимодействие с похожими компонентными структурами. Например, Windows-программа, при наличии соответствующего моста или объекта-обёртки, может использовать компонент JavaBeans так, будто бы он является компонентом COM или ActiveX.

Чтобы класс мог работать как bean, он должен соответствовать определённым соглашениям об именах методов, конструкторе и поведении. Эти соглашения дают возможность создания инструментов, которые могут использовать, замещать и соединять JavaBeans.

Правила описания гласят:

1. Класс должен иметь конструктор без параметров, с модификатором доступа public. Такой конструктор позволяет инструментам создать объект без дополнительных сложностей с параметрами.

2. Свойства класса должны быть доступны через get, set и другие методы (так называемые методы доступа), которые должны подчиняться стандартному соглашению об именах. Это легко позволяет

инструментам автоматически определять и обновлять содержание bean'ов. Многие инструменты даже имеют специализированные редакторы для различных типов свойств.

3. Класс должен быть сериализуем. Это даёт возможность надёжно сохранять, хранить и восстанавливать состояние bean независимым от платформы и виртуальной машины способом.

Так как требования в основном изложены в виде соглашения, а не интерфейса, некоторые разработчики рассматривают JavaBeans, как Plain Old Java Objects, которые следуют определённым правилам именования.

В приложении по автоматизации учета исключительных ситуаций есть восемь bean-классов (Issue, Priority, Project, Resolution, Role, Status, Type, User). У каждого из этих классов есть собственный конструктор и конструктор по умолчанию. Для обеспечения доступа к данным у каждого класса есть get и set методы.

Литература

1. Блинов, И.Н. Java 2: практическое руководство / И.Н. Блинов, В.С. Романчик. – Мн.: УниверсалПресс, 2005. – 318 с., ил.

2. Онлайн учебник по JSP [Электронный ресурс]. – 2015. – Режим доступа: <http://www.java2ee.ru/jsp/javaBean.html>. – Дата доступа: 13.04.2015.

3. Свободная энциклопедия Википедия [Электронный ресурс]. – 2013. – Режим доступа: <https://ru.wikipedia.org/wiki/JavaBeans>. – Дата доступа: 13.04.2015.

В.А. Шинкарёва (УО «ГГУ им. Ф. Скорины», Гомель)

Науч. рук. **В.Н. Леванцов**, ст. преподаватель

СИСТЕМА УПРАВЛЕНИЯ БАЗАМИ ДАННЫХ CUBRID

CUBRID является полноценной свободной системой управления базами данных с открытым исходным кодом хорошо оптимизированная для веб-приложений, особенно когда сложные бизнес процессы оперируют большими данными и вырабатывают огромное количество параллельных запросов.

Предоставляя уникальные оптимизированные функции, CUBRID позволяет обрабатывать гораздо больше параллельных запросов за более короткое время. С помощью CUBRID компании смогут воспользоваться преимуществами более высокой производительности