15,05,2018 Коллекции 9

Класс **Properties** — одна из реализаций интерфейса **Map**, которая имеет ряд особенностей и выполняет достаточно интересную роль — с его помощью можно легко создавать конфигурационные файлы и к тому же создавать программы, которые поддерживают многоязычность. Сначала посмотрим вариант с конфигурационным файлом. Для этого класса конфигуарционный файл — это обычный текстовый файл, в котором свойства хранятся в виде ключ/значение через знак «=». Вот так выглядит такой файл:

package edu.javacourse.collection;

import java.awt.BorderLayout;

import java.io.FileReader;

import java.io.IOException;

import java.util.Properties;

import javax.swing.JButton;

import javax.swing.JFrame;

public class PropertiesExample extends JFrame

{

    public PropertiesExample() {

        try {

            // Создаем объект

            Properties pr = new Properties();

            // Загружаем данные из файла - позже узнаем более подробно про файлы

            pr.load(new FileReader("simple.properties"));

            // Получаем свойства по именам - это же по сути Map

            String upText = pr.getProperty("up.button.title");

            String dnText = pr.getProperty("dn.button.title");

            // Создаем кнопки с указанными названиями

            JButton up = new JButton(upText);

            JButton dn = new JButton(dnText);

            add(up, BorderLayout.NORTH);

            add(dn, BorderLayout.SOUTH);

        } catch (IOException io\_ex) {

            io\_ex.printStackTrace(System.out);

        }

        setBounds(200, 200, 500, 200);

        setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

        setVisible(true);

    }

    public static void main(String[] args) {

        PropertiesExample pe = new PropertiesExample();

    }

Конфигурационный файл в голове проекта

# Comments - file "simple.properties"

up.button.title=UP

dn.button.title=DOWN

Как видите никаких сложностей нет — просто загрузили файл и он сам разобрался на составные части. Удобно.