The background features several large, overlapping, semi-transparent swirls in shades of purple, green, and blue. Scattered throughout are numerous small, yellow, triangular shapes, some pointing upwards and others downwards, resembling a sunburst or starburst effect.

# **Компьютерные технологии в метеорологии (Java-технологии)**

**Чукин В. В.**

**<http://www.chukin.ru>**



# Особенности Java

- Синтаксис Java основан на синтаксисе C / C++
- Строгая типизация данных: целые, вещественные, символьные...
- Независимость от операционной системы (Windows, Linux, Mac, Solaris)
- Компиляция текста программы в байт-код
- Интерпретация байт-кода с помощью Java Virtual Machine (JVM)



# Простой

- Java – язык на котором легко писать программы не требующий дополнительного обучения программистов C++
- Java максимально похож на C++
- Из Java исключены редко используемые и малопонятные средства языка C++



# Объектно-ориентированный

- Существуют два способа программирования:
- Процедурный (C, Fortrun, Basic)
- Объектно-ориентированный (C++, Java)

# Поддержка распределенных вычислений

- Поддержка протоколов TCP/IP, HTTP, FTP
- Вызов удаленных методов (web-службы)



# Надежность

- При работе Java-приложения осуществляется автоматическое выделение оперативной памяти и корректная работа с указателями (адресами памяти)



# Безопасность

- Java – язык для работы в сетевой среде.
- Java-приложения имеют средства защиты от вирусов и несанкционированного доступа

# Независимость от архитектуры компьютера


- Из текстового файла компилятор генерирует объектный файл (байт-код), формат которого не зависит от архитектуры компьютера
- Байт-код может выполняться на любых компьютерах
- Для перевода байт-кода в команды конкретного процессора служит Java Virtual Machine (JVM - виртуальная Java-машина)





# Переносимость

- Все элементы Java-приложений не зависят от операционной системы (Windows, Linux, Solaris, Mac)
- Размер основных типов данных и арифметические операции над ними строго определены
- Для символьных данных используется универсальная кодировка Unicode (UTF-8)



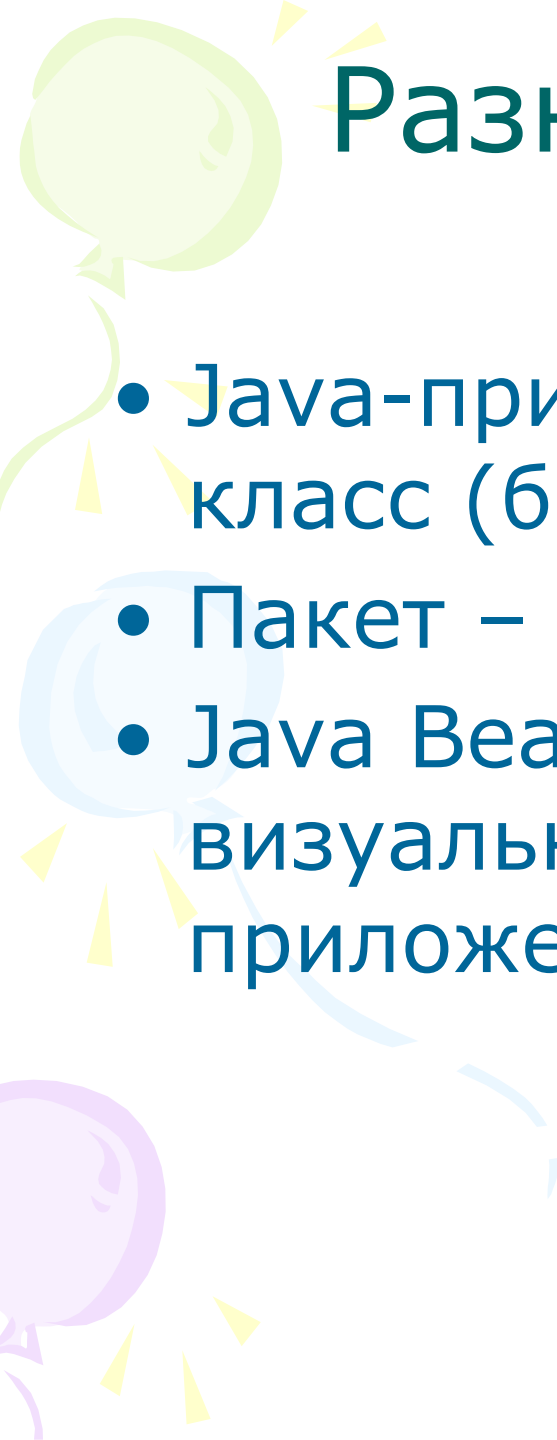
# Использование интерпретатора

- Java-приложение (байт-код) может выполняться на любом компьютере, для которого реализован интерпретатор (JVM)



# Основные понятия Java

- Класс
- Поле класса и метод класса
- Класс, созданный на основе другого класса называется классом-потомком
- Класс, на основе которого создан другой класс является классом-предком
- Пакет – совокупность нескольких классов

A decorative graphic on the left side of the slide features three balloons: a light green one at the top, a light blue one in the middle, and a light purple one at the bottom. Each balloon is attached to a streamer that flows downwards. Small yellow triangular shapes are scattered around the streamers, resembling confetti or streamer tassels.

# Разновидности Java-приложений

- Java-приложение = исполняемый класс (байт-код)
- Пакет – совокупность классов
- Java Bean – компонент для визуального создания Java-приложений в IDE



# Java-технологии для Internet

- Апплеты
- Сервлеты
- Java Server Pages (JSP)
- Web-службы
- Сетевые соединения (сокеты)



# Апплеты

- Апплет – Java-приложение, встроенное в html-страницу
- Для выполнения апплета браузер запускает JVM
- Доступ апплета к ресурсам компьютера пользователя очень ограничен из соображений безопасности



# Сервлеты

- Сервлеты – Java-приложения, выполняющиеся на сервере по запросу удаленных клиентов
- Для работы сервлета необходим запуск JVM
- Для этих целей служит специальный web-сервер – контейнер сервлетов (например, Apache Tomcat)



# Java Server Pages

- JSP – это удобный способ создания динамических html-страниц
- Перед выполнением JSP преобразуются в сервлеты
- JSP – это по сути сервлеты, но с более простыми правилами написания программы





# Web-службы

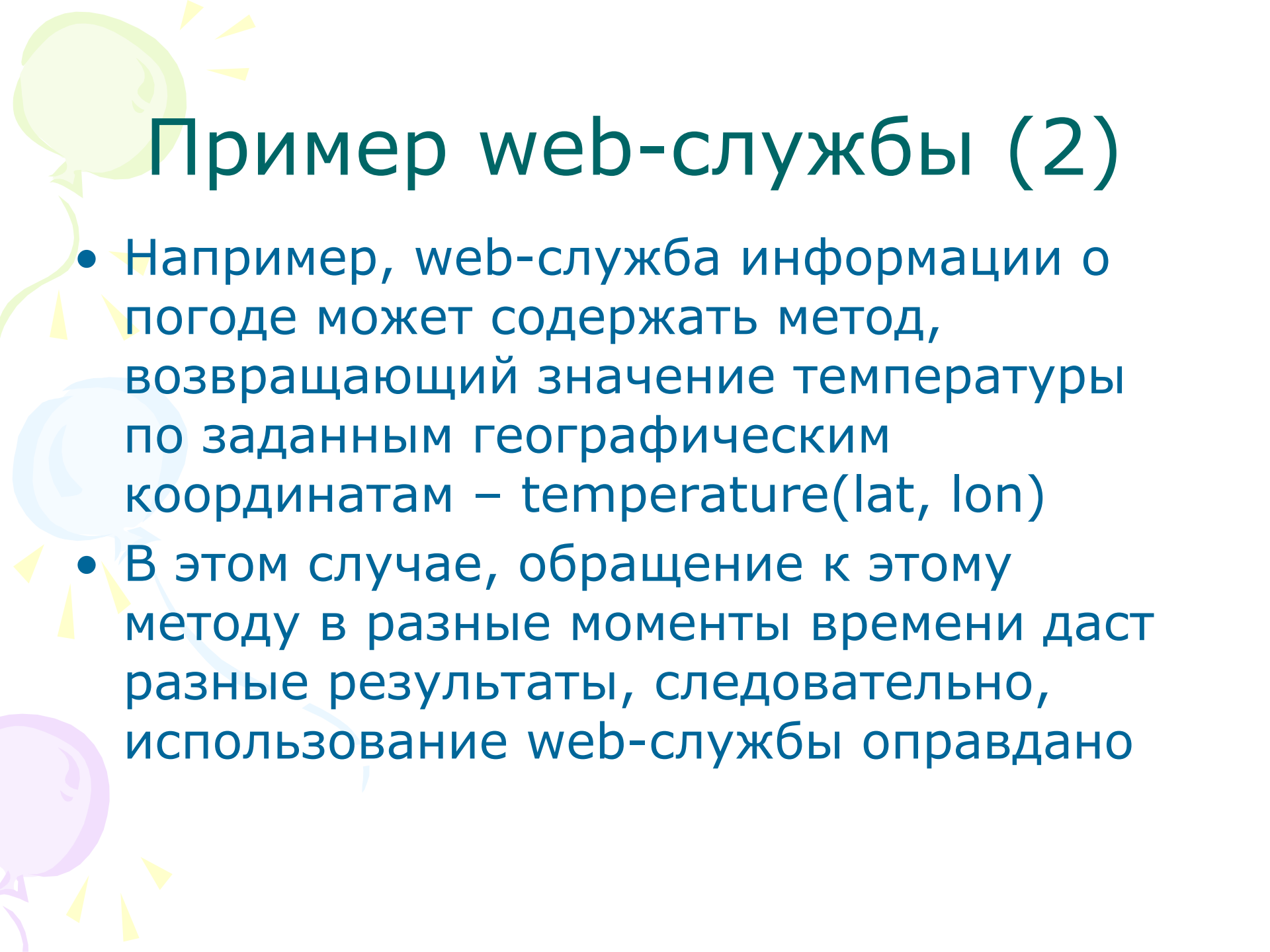
- Web-служба – это Java-приложение, выполняющееся на стороне сервера
- Выполняется не все приложение, а только какой-либо один метод
- Удаленный клиент получает результат выполнения этого метода

# Пример web-службы (1)

- Например, web-служба может в своем составе содержать метод расчета синуса угла –  $\sin(x)$
- В этом случае удаленный пользователь передает серверу имя вызываемого метода "sin" и значение угла "x"
- В ответ сервер возвращает значение синуса

# Пример web-службы (1)

- Поскольку значения функции  $\sin(x)$  известны и неизменны, то легче реализовать эту функцию на стороне клиента
- Web-службы следует использовать в случае изменяющегося со временем содержимого



## Пример web-службы (2)

- Например, web-служба информации о погоде может содержать метод, возвращающий значение температуры по заданным географическим координатам – `temperature(lat, lon)`
- В этом случае, обращение к этому методу в разные моменты времени даст разные результаты, следовательно, использование web-службы оправдано



# Сетевые соединения (сокеты)

- Из Java-приложения возможно подключение к портам удаленного компьютера (установление соединения = сокета) и обмен данными с сервером



# Java-технологии для мобильных устройств

- Мидлеты
- Поддержка Java-мидлетов включена в большинство современных сотовых телефонов



# Java-платформа

- Выделяют три разновидности Java-платформ в зависимости от типа устройств:
- J2ME – Java 2 Mobile Edition  
(для мобильных устройств)
- J2SE – Java 2 Standard Edition  
(для рабочих станций)
- J2EE – Java 2 Enterprise Edition  
(для серверов)



# Интегрированные средства разработки (IDE)

- IntelliJ IDEA
- NetBeans
- Eclipse
- JBuilder



# Пример создания Java-приложения в графической среде NetBeans

The screenshot displays the NetBeans IDE 6.0 interface. The main editor window shows the source code for `Main.java` within a project named `JavaApplication1`. The code defines a `Main` class with a `main` method that prints "Привет, мир!" and "Hello World!".

```
package javaapplication1;

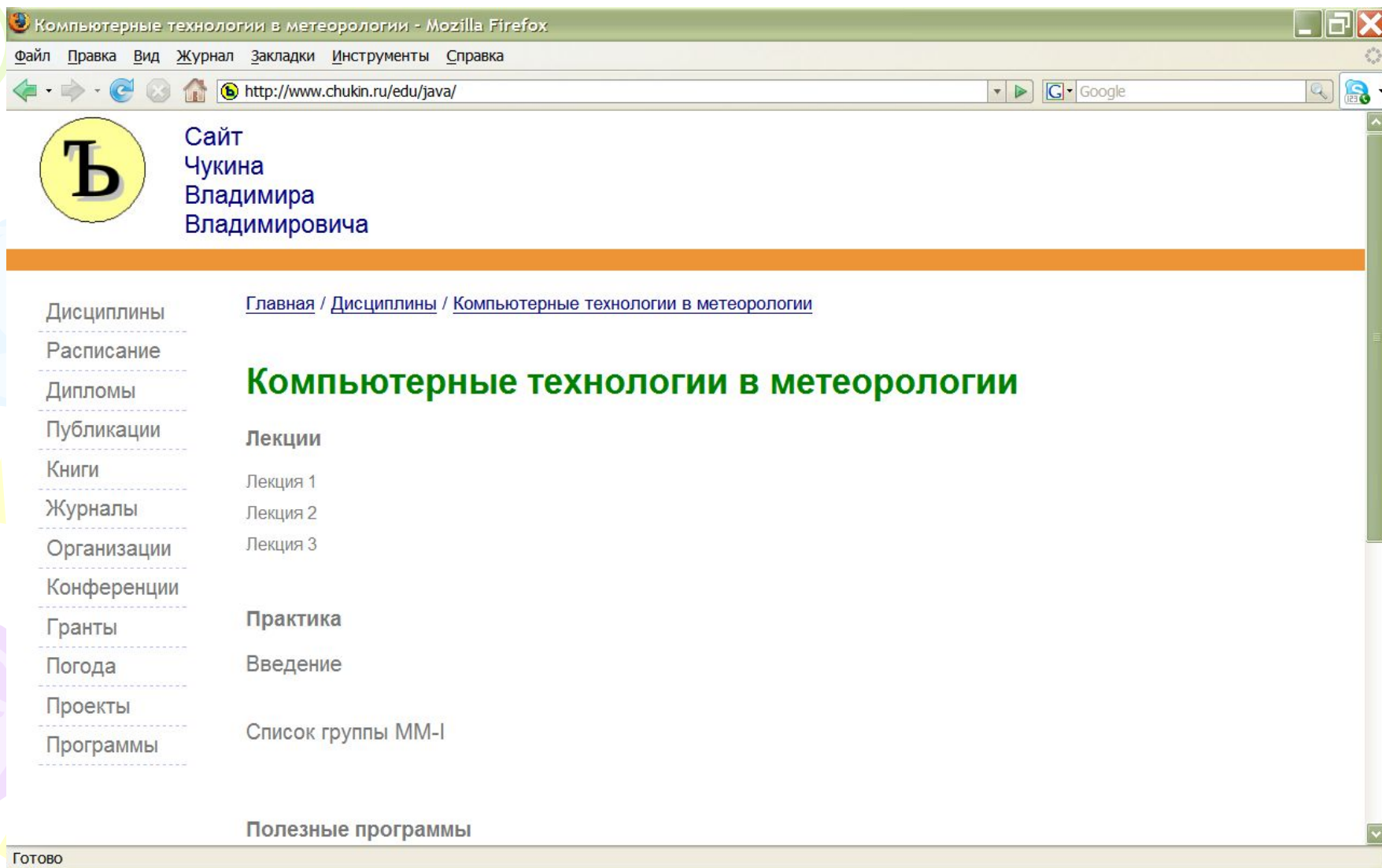
/**
 *
 * @author Vladimir V. Chukin
 */
public class Main {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
        System.out.println("Привет, мир!");
        System.out.println("Hello World!");
    }
}
```

The `Output - JavaApplication1 (run)` window at the bottom shows the execution results:

```
init:
deps-jar:
compile:
run:
Привет, мир!
Hello World!
BUILD SUCCESSFUL (total time: 2 seconds)
```


# Материалы по дисциплине на сайте <http://www.chukin.ru>



Компьютерные технологии в метеорологии - Mozilla Firefox

Файл Правка Вид Журнал Закладки Инструменты Справка

<http://www.chukin.ru/edu/java/> Google

 Сайт Чукина Владимира Владимировича

---

[Дисциплины](#) [Главная](#) / [Дисциплины](#) / [Компьютерные технологии в метеорологии](#)

[Расписание](#)

[Дипломы](#)

[Публикации](#)

[Книги](#)

[Журналы](#)

[Организации](#)

[Конференции](#)

[Гранты](#)

[Погода](#)

[Проекты](#)

[Программы](#)

## Компьютерные технологии в метеорологии

**Лекции**

- Лекция 1
- Лекция 2
- Лекция 3

**Практика**

- Введение
- Список группы ММ-1

**Полезные программы**

Готово