

## Конфигурация CLDC 1.1:

Пакеты:

**java.io** —

базовые интерфейсы и утилитные классы ввода-вывода

Подмножество java.io в Java SE 1.4.2

**java.lang** — фундаментальные классы Java Подмножество java.lang Java SE 1.4.2

**java.util** — коллекции и утилитные классы для хранения дат и времени. Подмножество java.util Java SE 1.4.2

**javax.microedition.io** - Generic Connection framework aka GCF. Ввод-вывод!Этих классов нет в JavaSE 1.4.2 API!!!

**Пакет java.lang**

-Классы - обёртки примитивных типов

(Boolean Byte Character Double Float Integer Long Short)

- System — обёртка вокруг ОС- Math — математика- Thread - нити

- String, StringBuffer — работа со строками

- Исключения (AIOOBE, ILE, RE, NPE, итд)

-Интерфейс Runnable

**Пакет java.io**

- Два интерфейса: DataInput и DataOutput

- Классы:

Потоки - ByteArrayInputStream ByteArrayOutputStream

DataInputStream DataOutputStream InputStream

PrintStream OutputStream

Чтецы и писатели (из потоков):

Reader InputStreamReader Writer OutputStreamWriter

- Исключения - EOFException InterruptedException

IOException UnsupportedEncodingException

UTFDataFormatException

**Пакет java.util**

- Интерфейс Enumeration

- Коллекции: Hashtable Stack Vector

- Время и дата: Calendar Date TimeZone

- Генератор случайных чисел: Random

- Исключения

EmptyStackException NoSuchElementException

**Пакет javax.microedition.io - Generic Connection Framework (GCF)**

- Интерфейсы: Connection ContentConnection Datagram DatagramConnection

InputConnection OutputConnection StreamConnection

StreamConnectionNotifier

- Класс Connector

- Исключение ConnectionNotFoundException

Как открыть соединение? Просто:

```
...
try {
    Connection c = Connector.open(« http://www.oracle.com»);
} catch(ConnectionNotFoundException cnfe) {
    System.out.println(«http is not supported!»);
}
...
HttpConnection httpc = (HttpConnection)c;
...
InputStream is = c.openInputStream();
int byteRead = is.read();
c.close();
```

**Профиль MIDP 2.0/2.1**

Пакеты:

java.lang (довесок к CLDC 1.1, содержит IllegalStateException)

java.util (2 класса Timer и TimerTask)



```

javafx.microedition.io (GCF+субинтерфейсы Connection)
javafx.microedition.lcdui (графика)
javafx.microedition.lcdui.game (спрайты)
javafx.microedition.media (АВВ)
javafx.microedition.media.control (АВВ, контролы)
javafx.microedition.midlet (жизненный цикл приложения)
javafx.microedition.pki (необходимы для SecureConnection)
javafx.microedition.rms (Хранилище)

```

## Безопасность для приложений MIDP

### Спецификация OTA/Provisioning

#### Пакет `javafx.microedition.lcdui`

- Главный класс: **Display**, всё начинается с него
- Главный метод: `Display.setCurrent(Displayable)`
- `Displayable`: «то, что может быть показано на дисплее»
- Наследники `Displayable`:  
  `Alert` `Canvas` `Form` `List` `Screen` и `TextBox`
- В `Form` можно добавить  
  `Image`-и и/или `Item`-ы (`ChoiceGroup`, `CustomItem`, `DateField`, `Gauge`, `ImageItem`, `Spacer`, `StringItem`, `TextField`)

#### Команды:

- Создание  
  `Command(String label, int commandType, int priority)`
- Добавление:  
  `Item.addCommand(Command)`,  
  `Item.removeCommand(Command)`  
  `Displayable.addCommand(Command)`,  
  `Displayable.removeCommand(Command)`
- Получение событий `Displayable.setCommandListener(CommandListener l)`

#### Интерфейс `CommandListener`

- Создайте свою реализацию `CommandListener`:

#### Метод `commandAction(Command c, Displayable d)`

- Добавление:  
  `Displayable.setCommandListener(CommandListener l)`  
  `Alert.setCommandListener(CommandListener l)`
- Узнавайте, из какой команды и `Displayable` пришло событие, через проверку аргументов!

#### Класс `Canvas` — анимация. Пример:

```

class Animation extends Canvas
    implements Runnable {
    // paint the current frame
    void paint(Graphics g) { ... }
    Display display; // the display for the application
    void paint(Graphics g) { ... } // paint the current frame
    void startAnimation() {
        // set up initial frame
        repaint();
        display.callSerially(this);
    }
    // called after previous repaint is finished
    void run() {
        if ( /* there are more frames */ ) {
            // set up the next frame

```



```
        repaint();  
        display.callSerially(this);  
    }  
}  
}
```

### События от пальцев в Canvas :-)

В классе Canvas есть методы

```
keyPressed(int kc)  
keyReleased(int kc)  
keyRepeated(int kc)  
pointerPressed(int x, int y)  
pointerDragged(int x, int y)  
pointerReleased(int x, int y)
```

```
boolean hasPointerEvents()  
boolean hasPointerMotionEvents()  
boolean hasRepeatEvents()
```

### Звук (АВВ), пакет javax.microedition.media

- Точное подмножество JSR 135 MMAPI
- Класс Manager
- Исключение MediaException

- Интерфейсы:

```
Control  
Controllable  
Player  
PlayerListener
```

### Простейший пример:

```
...  
try {  
    Player p = Manager.createPlayer("http://abc.wav");  
    p.start();  
} catch (MediaException pe) {  
} catch (IOException ioe) {  
}  
...
```

Дополнительная функциональность в плеерах:

Пакет javax.microedition.media.control:

Интерфейсы ToneControl и VolumeControl

### Приложения — MIDlet-ы

Пакет javax.microedition.midlet.MIDlet

Класс MIDlet

Исключение MIDletStateChangeException



## Пример

===== HelloWorld.java =====

```
import javax.microedition.midlet.*;
import javax.microedition.lcdui.*;

public class HelloWorld extends MIDlet implements CommandListener {
    private Command exitCommand;
    private TextBox tbox;

    public HelloWorld() {
        exitCommand = new Command("Exit", Command.EXIT, 1);
        tbox = new TextBox("Hello world MIDlet", "Hello World!", 25, 0);
        tbox.addCommand(exitCommand);
        tbox.setCommandListener(this);
    }

    protected void startApp() {
        Display.getDisplay(this).setCurrent(tbox);
    }

    protected void pauseApp() {}
    protected void destroyApp(boolean bool) {}

    public void commandAction(Command cmd, Displayable disp) {
        if (cmd == exitCommand) {
            destroyApp(false);
            notifyDestroyed();
        }
    }
}
```

## Домашняя работа по второй лекции JavaME

### Задача 1

=====

Реализуйте один из алгоритмов сортировки используя лишь CLDC 1.1 API.

Входные параметры:

массив из N элементов типа int, заполненных случайными числами от 0 до N.

### Задача 2 (\*)

=====

- Скачайте JavaME Platform SDK по ссылке <http://download.oracle.com/javame/>

1. Напишите MIDlet, с двумя командами:
  - «About» и «Exit»
2. По активации команды «About» - показывается имя и фамилия в Alert-е,
  - По активации команды «Exit» - ваш MIDlet завершает исполнение

