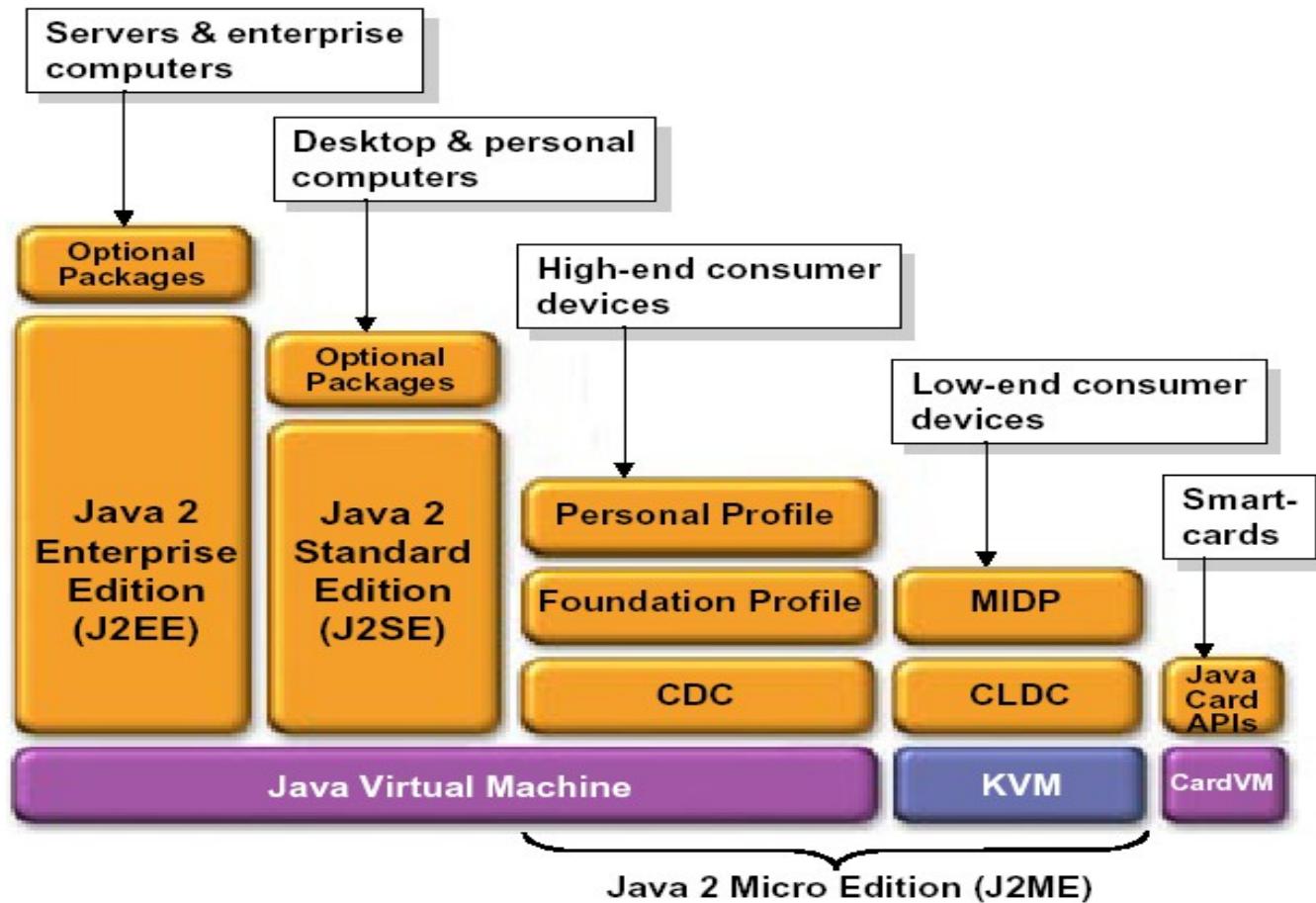
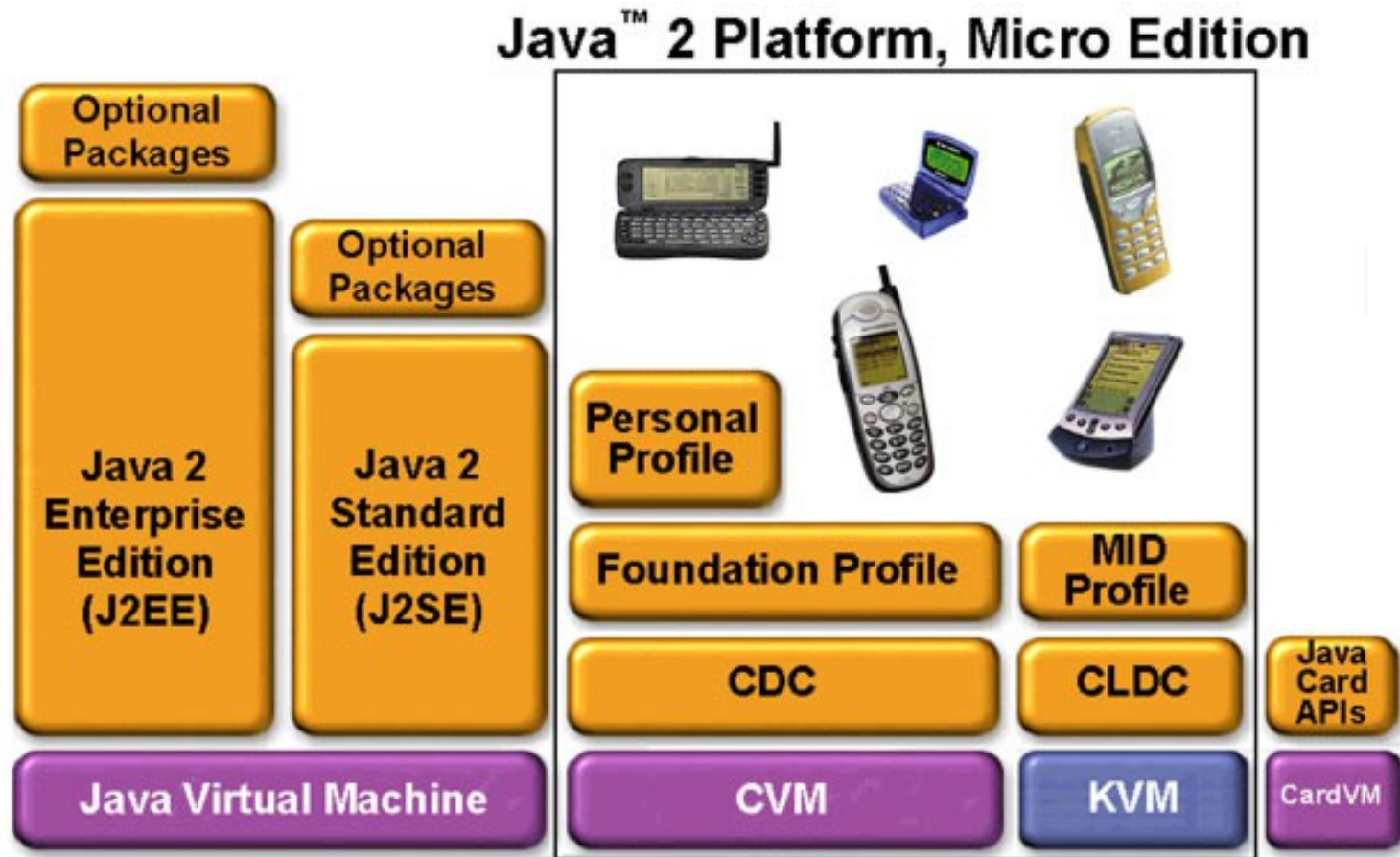


# Uvod v JAVA 2 Micro Edition

# Platforma Java 2



# Platforma Java2



# Osnovni pojmi v J2ME

## ☛ Konfiguracije (CLDC, CDC):

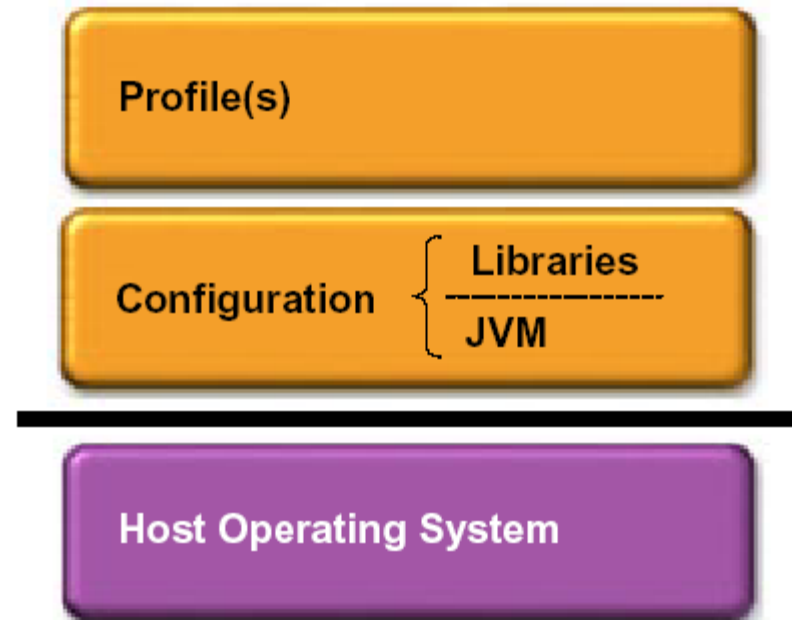
- Definirajo minimalno platformo skupine naprav (horizontalna kategorija)
- Podobne zahteve glede pomnilnika
- Podobne zahteve glede procesorske moči

## ☛ Profili (MIDP, Foundations):

- Zgornja plast, ki “razširi” konfiguracijo
- Naslavlja zahteve, specifične za določen “vertikalni” segment družine naprav

## ☛ JVM (Java Virtual Machine, KVM):

- Vsi uporabljajo “Java Virtual Machine”, ki je del konfiguracije
- Izvaja se na danem gostiteljskem sistemu
- KVM – Kilo Virtual Machine



# Konfiguracije Java 2 ME

☞ Dve veliki kategoriji izdelkov:

- **Potrošniške naprave zgornjega razreda**

**CDC:** Connected Device Configuration

- **Potrošniške naprave spodnjega razreda**

**CLDC:** Connected LIMITED Device Configuration

# Connected Device Configuration (CDC)

- TV set-top box, Internet TV, Internetski zaslonski telefoni, brezžični komunikatorji visokega razreda, zabavni in navigacijski sistemi v avtomobilih
- Široka zmožnost uporabniških vmesnikov
- Velik pomnilnik – Začenši pri 2 do 4 MB
- Trajne in širokopasovne omrežne povezave
- Pogosto uporabljamo TCP/IP

# Connected Limited Device Configuration (CLDC)

- ☛ Mobilni telefoni, pozivniki, osebni organizatorji ipd.
- ☛ Uporabljajo enostavne uporabniške vmesnike (v primerjavi z osebnimi računalniki)
- ☛ Najmanjši pomnilnik od 128 KB naprej, ozka pasovna širina
- ☛ Občasne omrežne povezave
- ☛ Običajno baterijsko napajanje
- ☛ Pogosto ne uporabljajo TCP-IP

# Mobile Information Device Profile (MIDP)

- Definira skupino API (application programming interface), ki doda CDLC minimalno množico zmožnosti
- Minimalne zahteve za pomnilnik**
  - 128 KB stalnega pomnilnika za komponente MIDP
  - 8 MB stalnega pomnilnika za tranje podatke aplikacij
  - 32 KB začasnega pomnilnika ("RAM") za virtualni stroj
- Minimalne zahteve za prikazovalnik**
  - Velikost zaslona: 96 x 54
  - Globina zaslona: 1-bit
  - Velikost piksla: 1:1
- Vhod**
  - Tipkovnica, zaslon, občutljiv na dotik
- Omreženje**
  - Dvosmerno, brezžično, občasne povezave, omejena pasovna širina





# K Virtual Machine (KVM)

- ☛ CLDC in MIDP uporabljajo Sunov “K Virtual Machine”
- ☛ Kompakten, prenosljiv, razvit za male naprave z omejenimi viri
- ☛ Preprost, dobro komentiran
- ☛ Modularen in prilagodljiv
- ☛ Zahteva vsaj 128 KB celotnega pomnilnika
- ☛ Več o tem na: <http://java.sun.com/products/kvm>



# Namen CLDC

## CLDC obravnava

- Jezik Java in Javanski virtualni stroj
- Osnovne ključnice (Core Libraries)
- Vhod/izhod
- Omreženje
- Varnost
- Internacionalizacijo

## CLDC ne obravnava

- Življenjskega cikla aplikacij
- Funkcionalnosti uporabniških vmesnikov
- Rokovanja z dogodki
- Visokonivojskega modela aplikacij

Te stvari obravnava MIDP



# Namen MIDP

## MIDP obravnava

- Pomen in krmiljenje aplikacij MIDP
- Uporabniški vmesnik (vključno s prikazovalnikom in vhodom)
- Shrambo podatkov
- Časovnike (timers)



## MIDP ne obravnava (zaradi različnih aparaturnih infrastruktur)

- Systemske klice API
- Dostavljanje in upravljanje aplikacij (aparaturno odvisno)
- Nizko nivojsko varnost (kar ne nudi že CLDC)
- Varnost na nivoju aplikacij
- Varnost med uporabniki (end to end)



# Varnost v CLDC/MIDP

- ☛ Nizko nivojska varnost (Virtual Machine Security):
  - Zagotavlja, da slabo ali škodljivo zasnovana koda ne poruši naprave
  - Ta nivo zagotavlja “*Class File Verifier*”
- ☛ Varnost na nivoju aplikacije:
  - Zagotavlja, da lahko javanska aplikacija na napravi dostopa le do tistih knjižnic in resursov, ki jih okolje naprave dopušča.
  - Ta nivo zagotavlja model “peskovnika”
- ☛ Varnost med uporabniki:
  - Zagotavlja zaščito transakcije, ki jo je sprožila ena naprava, vse do druge naprave, ki nudi zahtevano storitev, vključno s potjo med obema)
  - Ta nivo je odvisen od implementacije in zato NI podpiran



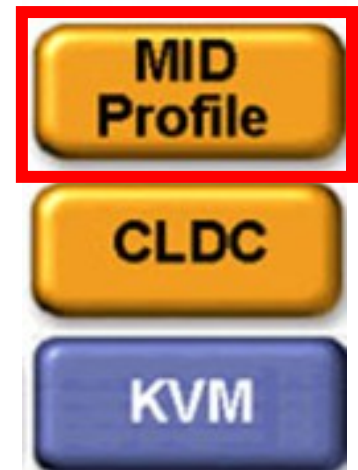
# Podrobnosti CLDC

- ☛ Osnovna tehnologija za enega ali več profilov *J2ME*, na primer MIDP
- ☛ Ni usmerjena v neko specifično vrsto naprav (različne naprave in zmožnosti)
- ☛ Ne podpira upravljanja aplikacij
- ☛ Ne podpira plavajoče vejice
- ☛ Ne pozna finalizacije (`java.lang.Object.finalize`)
- ☛ Omejena zmožnost obravnave napak
- ☛ Zaradi varnosti in omejitev ne podpira JNI (Java Native Interface)
- ☛ Zaradi varnosti ni uporabniško definiranih nalagalnikov razredov
- ☛ Ni skupin niti in demonskih niti



# Podrobnosti MIDP

- ☛ Zaradi varnosti ne uporabljamo običajnega modela za javanske aplikacije
- ☛ Osnovna izvajalna enota je **MIDlet**, ki je izpeljan iz ***javax.microedition.MIDlet***
- ☛ Za souporabo podatkov in resursov uporabljamo **MIDlet suites**
- ☛ MIDlet suite je skupina MIDlets ki so povezani skupaj v datoteko **JAR**.



# MIDleti

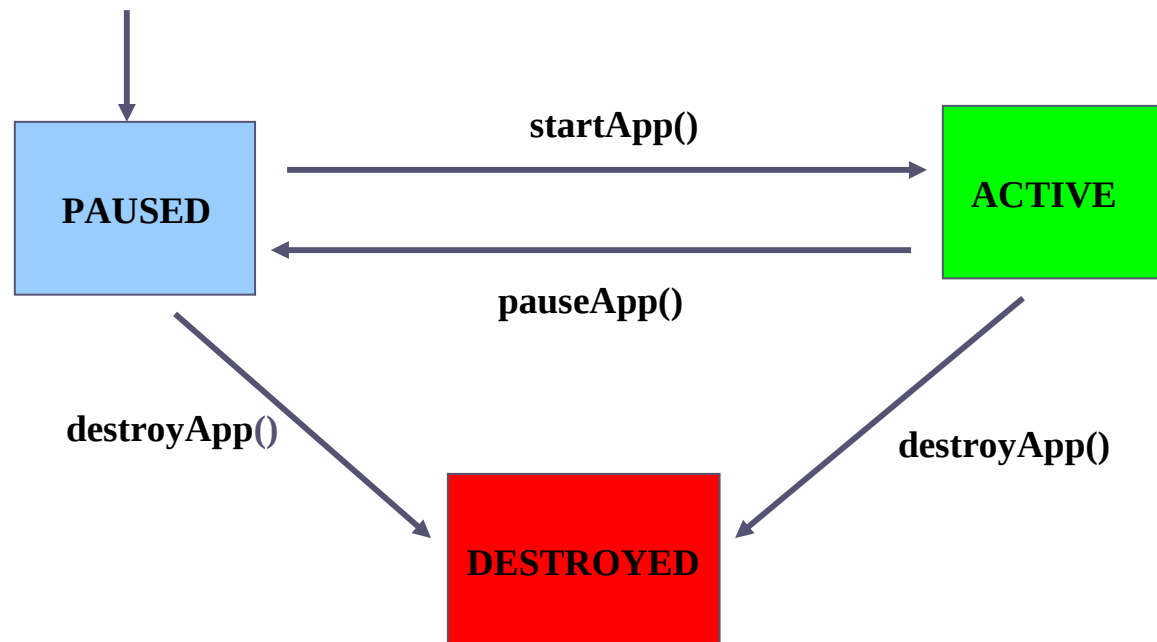
- ☞ Vsak MIDlet je izpeljan iz razreda MIDlet.
- ☞ Vsak Midlet mora implementirati tri abstraktne metode, definirane v *javax.microedition.midlet.MIDlet*
  - **startApp():** Z njo sprožimo MIDlet in pridobimo potrebne resurse
  - **pauseApp():** Uporablja jo sistem, ki zahteva začasno prekinitev MIDleta. Sprosti resurse, ki jih je zasedla metoda *startApp()*.
  - **destroyApp():** I
  - Pokliče jo sistem pred ukinitvijo MIDleta. Lahko jo tudi kliče sam MIDlet. Z njo sprostimo vse zasedene resurse.

# Stanja MIDleta

☛ MIDlet med svojim življenjem prehaja med naslednjimi stanji.

- **Paused:** Preden se MIDlet zažene ali po klicu metod *pauseApp* ali *notifyPaused*. Minimalna poraba CPE. Na primer pri prejemu telefonskega klica.
- **Active:** Po vstopu v metodo *startApp* ali po povratku iz stanja Paused z metodo *resumeRequest*.
- **Destroyed:** Po povratku iz metod *destroyApp* ali *notifyDestroyed*. To je zadnje stanje življenja MIDleta.

**new HelloWorld()**





# Razred MIDlet

```
abstract class MIDlet {  
    // called by the platform  
    abstract void startApp();  
    abstract void pauseApp();  
    abstract void destroyApp();  
    // can be called by the subclass  
    void notifyPaused();  
    void notifyDestroyed();  
    // plus various property accessors  
}
```

# Primer preprostega MIDleta

```
import javax.microedition.midlet.*;
import javax.microedition.lcdui.*;
public class HelloMIDlet extends MIDlet implements CommandListener {
    public HelloMIDlet() {
        display = Display.getDisplay(this);
        exitCommand = new Command("Exit", Command.SCREEN, 2);
    }
    public void startApp() {
        TextBox t = new TextBox("Hello MIDlet", "Wireless Internet", 256, 0);
        t.addCommand(exitCommand);
        t.setCommandListener(this);
        display.setCurrent(t);
    }
    public void pauseApp() { }

    public void destroyApp(boolean unconditional) { }

    public void commandAction( Command c, Displayable s) {
        If (c == exitCommand) {
            destroyApp(false); notifyDestroyed();
        }
    }
}
```



Demo

# CLDC/MIDP “MIDlet”

- ☛ Razred **javax.microedition.midlet.MIDlet** nudi platformo za krmiljenje aplikacije MIDP.
- ☛ Je analogen metodi **public static void main(String [ ] args)** za javanske aplikacije oziroma razredu **java.applet.Applet** za aplete.
- ☛ Ima lahko tudi več “vstopnih” točk.

# Garnitura MIDlet (MIDlet Suite)

- ☞ **JAR datoteka** garniture MIDlet vsebuje
  - Razrede (**class**) , ki implementirajo MIDlete.
  - Vse druge datoteke, ki jih Midleti uporabljajo: ikone, slikovne datoteke itd.
  - **manifest** , ki opisuje vsebino JAR
- ☞ Zaradi varnosti posameznih datotek v garnituri ne moremo spreminjati.
- ☞ Drugače povedano: osnovna enota za namestitev aplikacije, njeno osveževanje ali odstranjevanje je garnitura MIDlet (MIDlet suite).

# Pakiranje MIDletov

- ☞ Enega ali več MIDletov pakiramo v “MIDlet Suite”:
  - Javanske “class” datoteke, druge potrebne datoteke (slike ipd.) in Manifest, vse skupaj v **datoteki JAR** ,
  - **Datoteka JAD** (Java Application Descriptor) opisuje MIDlete v obliki javanskih lastnosti (pari **key: value**)
- ☞ Datoteka JAD omogoča upravniku aplikacij na napravi odločitev, ali MIDlete lahko izvedemo na napravi ali ne.

# Lastnosti datoteke JAD

**MIDlet-Name**

**MIDlet-Version**

**MIDlet-Vendor**

**MIDlet-<n>** for each MIDlet

**MIDlet-Jar-URL**

**MIDlet-Jar-Size**

- ☛ Lastnosti **MIDlet-<n>** določajo ime MIDleta, njegovo ikono in ime razreda v datoteki JAR, ki implementira ta MIDlet

# Primer datoteke JAD (Java Application Descriptor)

MIDlet-Name: HelloMIDlet

MIDlet-Version: 1.0.0

MIDlet-Vendor: Sun Microsystems, Inc.

MIDlet-Description: A Simple Example

MIDlet-Info-URL: <http://java.sun.com/j2me/>

MIDlet-Jar-URL: Hello.jar

MIDlet-Jar-Size: 1063 MicroEdition-Profile:

MIDP-1.0 MicroEdition-Configuration: CLDC-1.0

MIDlet-1: Hello,, HelloMIDlet *(Name, Icon, Class)*

# Nameščanje garnitur MIDlet

- ☞ Garniture MIDlet (MIDlet suites) lahko posredujemo napravam na enega od naslednjih načinov:
  - **Kopiramo** jih direktno v pomnilnik naprave
  - Poiščemo jih z **brkljalnikom** (ki razume MIME tip JAD), ki teče na napravi.
  - Pošljemo jih napravi brezžično z **SMS**

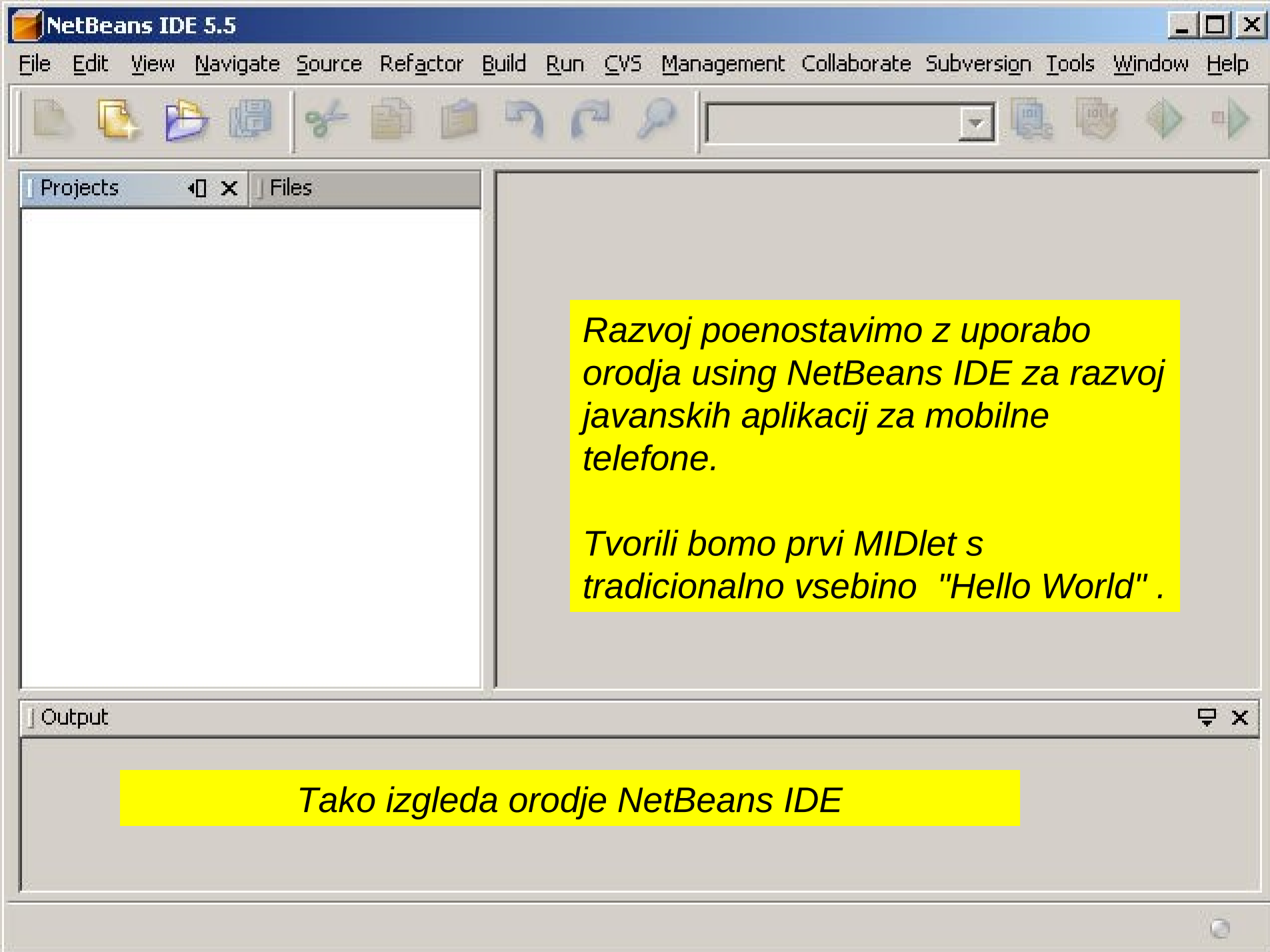


# Moj prvi MIDlet

- ☛ Razvoj poenostavimo z uporabo orodja using NetBeans IDE za razvoj javanskih aplikacij za mobilne telefone.

Tvorili bomo prvi MIDlet s tradicionalno vsebino "Hello World" . "So what's a MIDlet?", you say. Bah! Details...details. Who cares, right? Seriously, in simple terms a MIDlet is a java program you can run on mobile devices such as java enabled cellphones. A MIDlet is to your phone as an exe is to pc (excuse the bad analogy heh).

If you haven't already, open the NetBeans IDE. You should see something similar to this:



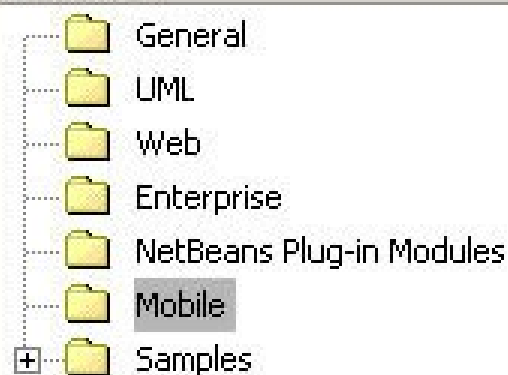
## Steps

1. Choose Project

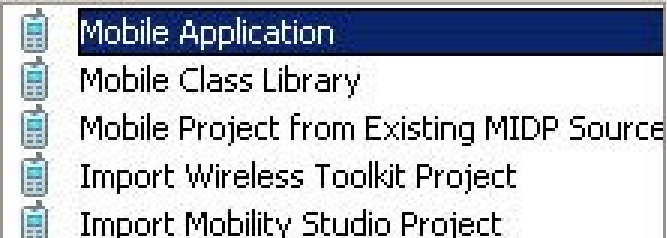
2. ...

## Choose Project

Categories:



Projects:



Description:

ect. You can also  
**generated Ant build**  
preverify, run,  
t.

*V glavnem menuju izberemo "File" in kliknemo na "New Project". Iz seznama kategorij izberemo "Mobile", v seznamu projektov pa "Mobile Application". Nato kliknemo na gumb "Next".*

< Back

Next >

Finish

Cancel

Help

## Steps

1. Choose Project
2. **Name and Location**
3. Default Platform Selection
4. More Configurations Selection

## Name and Location

Project Name: MyMidlet

Project Location: D:\MyDocs\netbeans

Browse...

Project Folder: D:\MyDocs\netbeans\MyMidlet

☒ Set as Main Project

☒ Create Hello MIDlet

*Projektu damo ime in izberemo lokacijo na disku. V našem primeru smo izbrali "MyMidlet". Odkljukajmo tudi opciji "Set as Main Project" in "Create Hello MIDlet". Nato kliknimo na "Next".*

< Back

Next >

Finish

Cancel

Help

New Project

Steps

1. Choose Project

2. Name and Location

3. **Default Platform Selection**

4. More Configurations Selection

Default Platform Selection

Emulator Platform: J2ME Wireless Toolkit 2.2

Device: DefaultColorPhone

Device Configuration: ☒ CLDC-1.0 ☐ CLDC-1.1

Device Profile: ☐ MIDP-1.0 ☒ MIDP-2.0 ☐ MIDP-2.1

*Izberemo emulator.. I zberimo najnovejšega.  
Naprava (device) naj bo "Default ColorPhone". V  
"Device Configuration" izberimo še CLDC-1.0, kot  
profil naprave pa MIDP 2.0. Zaključimo s klikom na  
gumb "Finish" .*

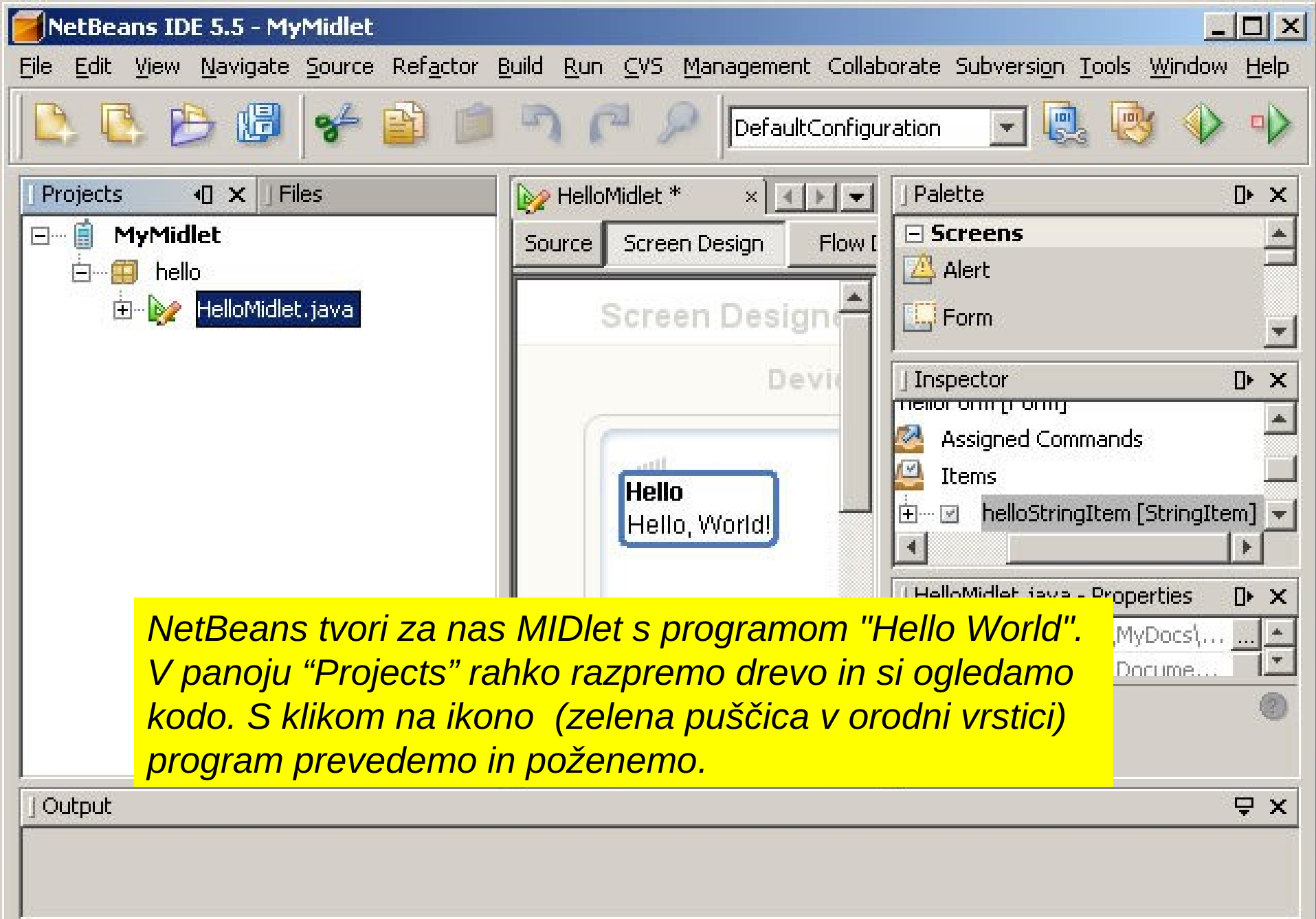
< Back

Next >

Finish

Cancel

Help



*NetBeans tvori za nas MIDlet s programom "Hello World". V panoju "Projects" rahko razpremo drevo in si ogledamo kodo. S klikom na ikono (zelena puščica v orodni vrstici) program prevedemo in poženemo.*

*Po tvorbi in zagonu programa se na zaslonu pojavi emulator. Na zaslonu simuliranega telefona dobimo seznam MIDletov. Kliknemo na telefonski gumb za izbiro in poženemo MIDlet.*





Painless, wasn't it? Now about those "Platform Settings" we chose earlier. It really depends on the phone you want your program to run on. Most phones at the time of writing already supports MIDP 2.0 and this in turn allows us to use new libraries that were added to J2me specifically for game development. From here on, the rest of the tutorials you will find here will be focused on MIDP 2.0 phones.

The reason that I chose CLDC-1.0 is for compatibility. If you're targeting a specific phone model it's best to view the device specification for that phone from the manufacturers website. One of the main changes from CLDC-1.0 and CLDC-1.1 is that in the latter they have added support for real numbers or floats. Unless you really need to use floats in your code I suggest sticking to CLDC-1.0 for the time being. You can also find resources on the internet on how to deal with floats on the CLDC-1.0 platform.



# Sun Java Wireless Toolkit

- ☛ The Sun Java Wireless Toolkit (prej znan pod imenom J2ME Wireless Toolkit) je skupina orodij za tvorbo aplikacij, ki tečejo na napravah, skladnih s specifikacijo Java Technology for the Wireless Industry (JTWI, JSR 185) in specifikacijo Mobile Service Architecture (MSA, JSR 248).
- ☛ Vsebuje razvojna orodja in emulator.

# Koraki razvoja MIDletov

- ☞ Namestitev orodij
  - JDK 1.2 ali kasnejša, CLDC, MIDP
  - JDK 1.2 ali kasnejša, kakšen “toolkit” (**Java2ME**, NetBeans IDE 6.0.1 itd.)
- ☞ Vzpostavitev okolja (ne potrebujemo, če uporabljamo kakšen “toolkit”)
- ☞ Napišemo MIDlet
- ☞ Prevedemo MIDlet
  - `C:/javac -d .\ -bootclasspath c:\MIDPClassPath; C:/MIDletPath HelloMIDlet.java`
- ☞ Preverimo MIDlet
  - `C:/preverify -classpath c:/MIDPClassPath;C:/MIDletPath HelloMIDlet.class`
- ☞ Poženemo MIDlet (na razvojnem računalniku)
  - `C:/Example/Output>midp HelloMIDlet`
- ☞ Pakiramo MIDlet
  - `C:/Jar cf Hello.jar HelloMIDlet.class`
- ☞ Za MIDlet napišemo opisno (Descriptor) datoteko (JAD) (ni potrebno, če uporabljamo “toolkits”)
- ☞ Spet poženemo MIDlet
  - `C:/midp -descriptor Hello.jad`

# Namestitev razvojnih programov



Skopiramo in namestimo

- JDK 1.2 ali kasnejši
- CLDC
- MIDP



Programe namestimo in razpakiramo na naslednji način:

- C:/JDK1.2
- J2me
  - |
  - MIDP
  - |
  - CLDC

# Vzpostavitev okolja

## Vzpostavimo PATH

- `SET PATH=%PATH%;%JAVAHOME%\BIN;`  
`PATH=C:\WINDOWS;C:\WINDOWS\COMMAND;C:\JDK\BIN; C:\j2me\midp\bin;%PATH`  
`%` (vse v eni vrstici)

## Preverimo okolje

- `C:/midp` (pojaviti se mora slika telefona)

## Nastavimo globino zaslona

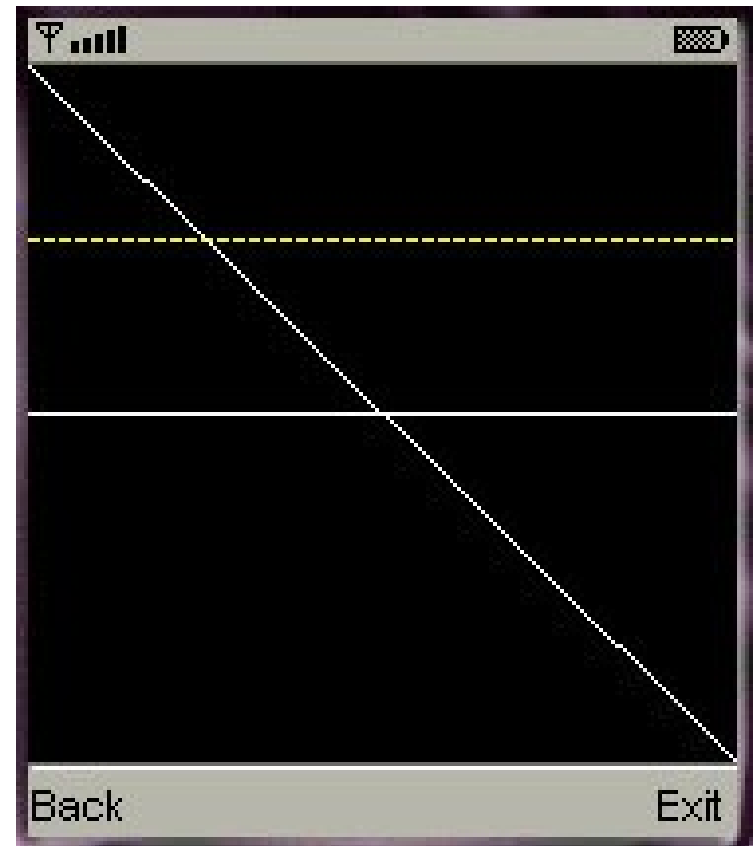
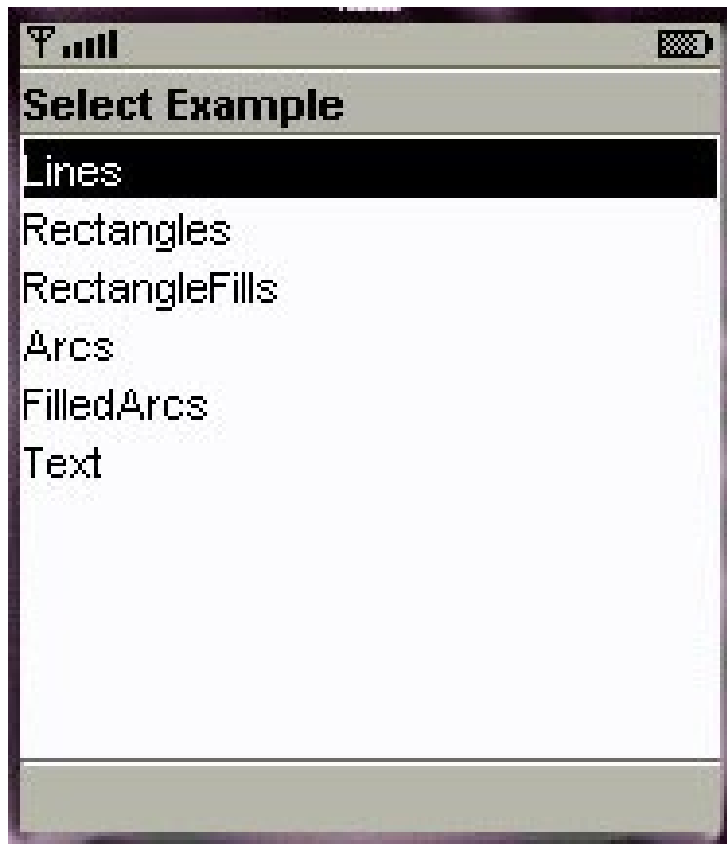
- `C:\set SCREEN DEPTH=8`

## Program preskusimo sz nekim primerom

- `C:\midp -descriptor run.jad` (run.jad je primer datoteke jad)

Še več zanimivih primerov

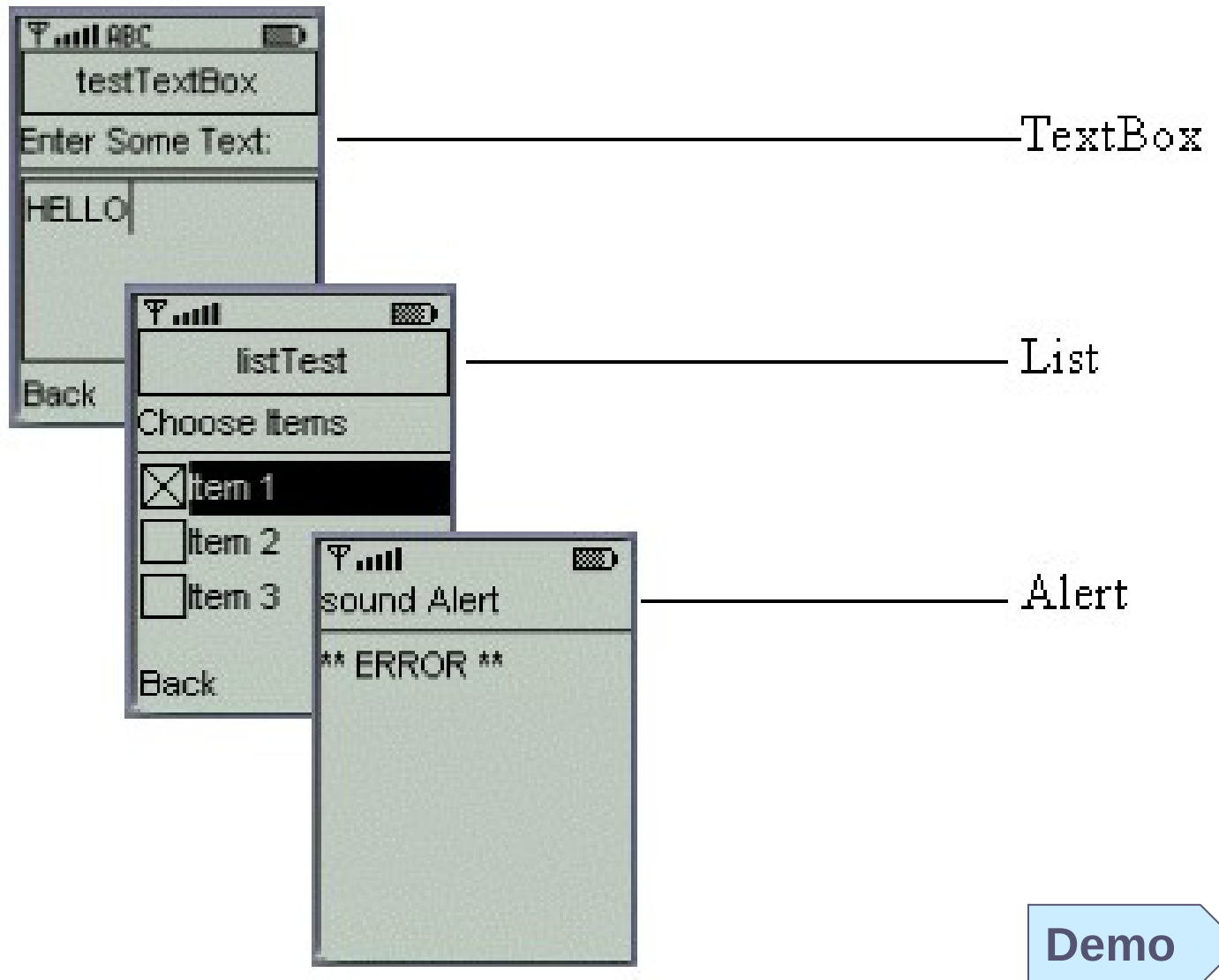
# Primer: GraphicsMidlet (primeri z grafiko)



Še več zgledov

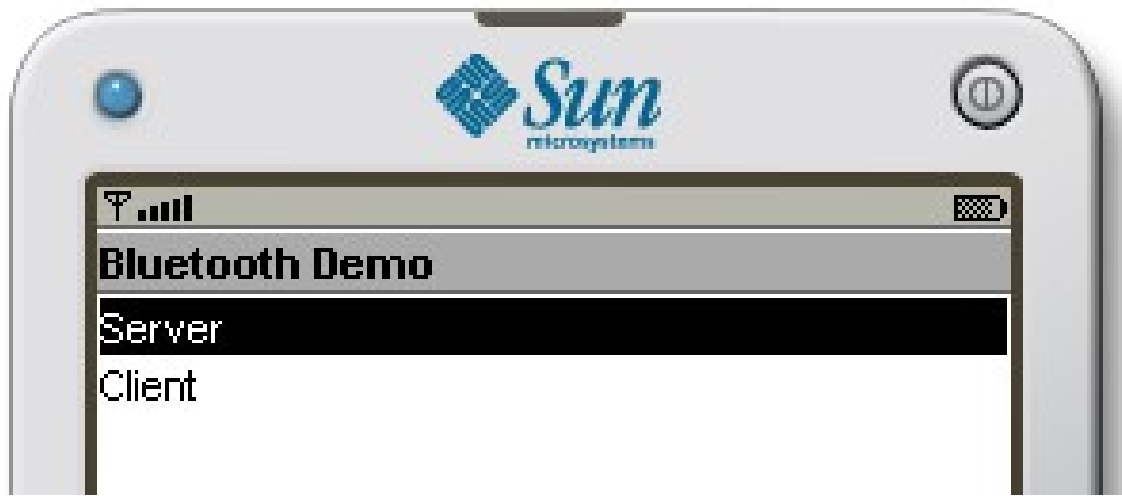
Demo

# Primer: GuiTests



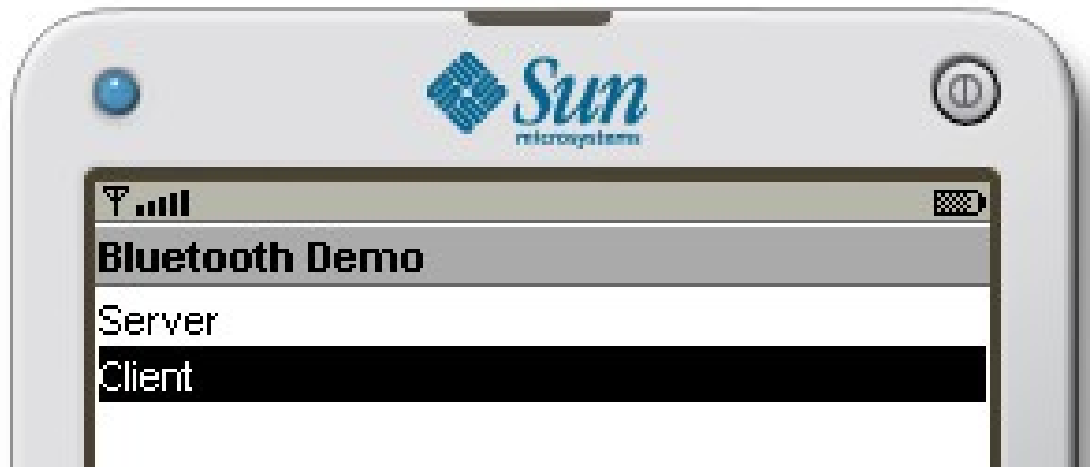
# MIDleti in povezava Bluetooth

## Primer: BluetoothDemo



Bluetooth Demo

Server



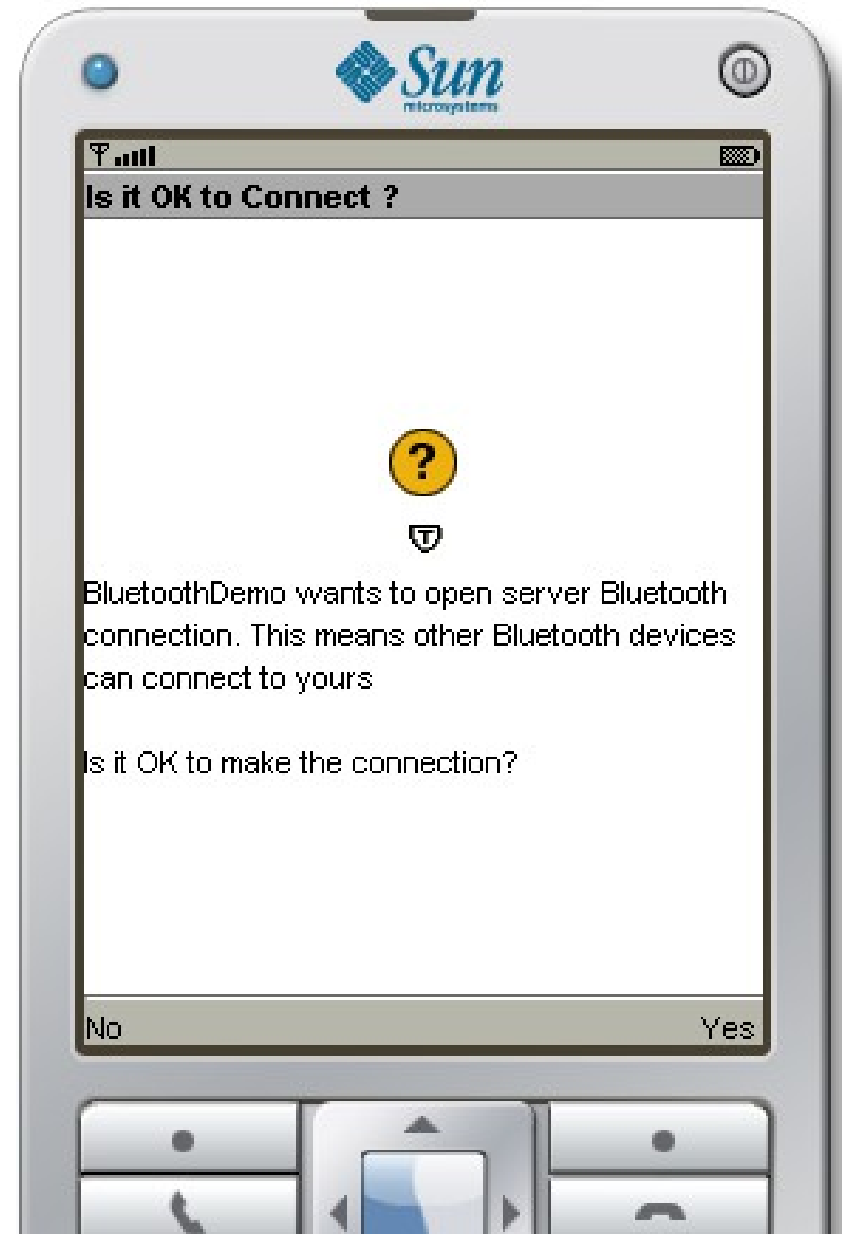
Client



# BluetoothDemo: Server

```
Device Discovered
Major Device Class: 512 Minor Device
Bluetooth Address: 0000000DECAF
Bluetooth Friendly Name: WirelessToo
InquiryCompleted
ServicesDiscovered
SERVICE_SEARCH_COMPLETED

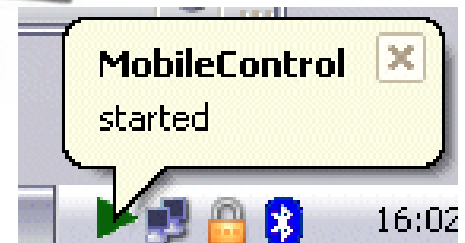
Service URL: btsp://0000000DECAF:1;
Hello Back from Server
```



# MIDleti in uporaba bluetooth

## Primer: Mobile Control

“Mobile Control suite” je neke vrste daljinec za nadzor PCX aplikacij, kot sta na primer PowerPoint in Windows Media Player. Komunikacija med mobilno npravico in PC poteka preko bluetooth.



# Mobile Control Midlet (MobCon)

- Platforma je Win32 s podporo Bluetooth Stack in Serial Profile ter J2ME MIDP2.0 z Bluetooth API (JSR-82).
- Krmiljenje aplikacij omejeno na enočrkovne ukaze, lahko z modifikatorskimi tipkami , kot sta CTRL ali ALT. Konfiguracijo moramo tvoriti na krmilnem računalniku PC vnaprej. Ustreune aplikacije moramo pognati najprej.
- Aplikacije razpozna MobCon s pomočjo regularnih izrazov, definiranih v konfiguraciji. Ti izrazi se morajo ujemati z imeno oken na namizju.
- Daljinsko krmiljenje izvajamo z enotipkovnimi ukazi, podpore grafike nimamo.
- Tako lahko krmilimo predvajalnike mp3, PPT prosojnice, kakšen drug podoben program.

# Mobile Control Midlet (MobCon)

## MobCon Main Menu

Select **Search Device** to start a new device search. If a connection to a Device was stored from a previous session, it is available as item **Connect to *DeviceName***. The mobile itself might prompt you to allow network (bluetooth) access after selection.

## Devices search

The midlet tries to find BT devices within the neighborhood. When search is finished, found devices are displayed and can be selected. After selection, the midlet queries the appropriate Serial Profile service on the device. If successful, the midlet will try to contact the MobileControl Application and query it for available applications. That process can be aborted by pressing **break** on the progress screen.

## Application selection

If the midlet received information from the MobileControl Application, it displays the configured application names. Selecting an application from the list again queries the action-list for that application.

## Actions invocation

After receiving the configured action commands from the MobileControl Application, the midlet tries to get the current window title of the selected application, which then will be displayed as a ticker. If that fails because the selected application was not started or recognized by the MobileControl Application, the ticker will display an error message. In that case the MobileControl Application will reconfigure itself, so you can retry your command. Selecting an action sends that action to the MobileControl Application and will receive the current window title as response, which will update the ticker row.

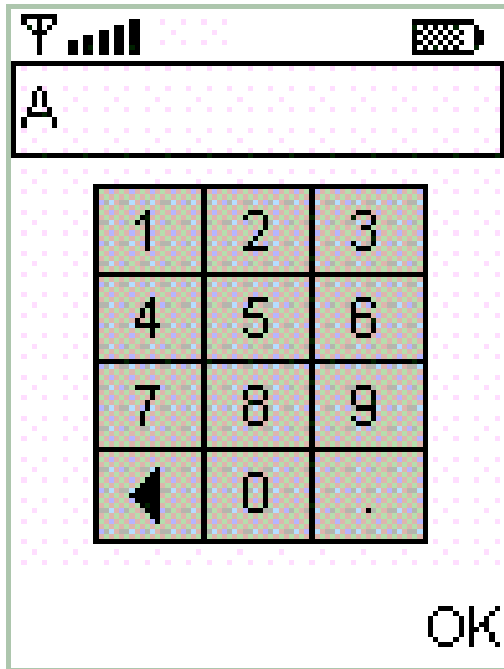
# Podrobnosti

- Aplikacija Mobile Control
- MIDlet:
- Konfiguracijska xml datoteka: config.xml
- Imenovanje ukazov, razred KeyEvent

[www.midlet.org](http://www.midlet.org)

☛ Dodatni zgledi



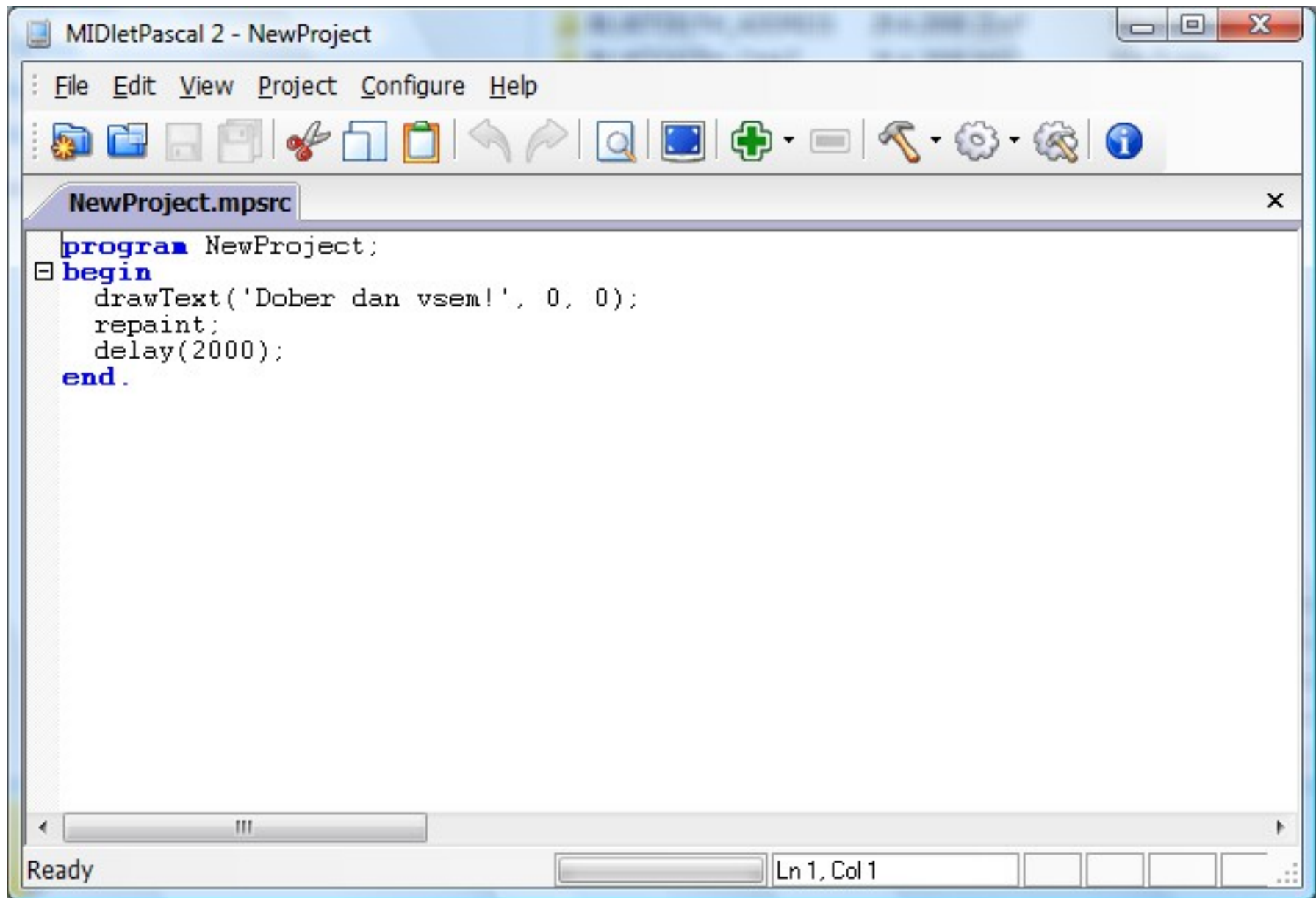


**Software Keyboard 1.0.0**



**UrlReader 0.9.1**

# MIDletPascal





# Prednosti Java 2 ME

- ☛ Dinamično dostavljanje vsebin
- ☛ Varnost(preverjanje razrednih datotek, dobro definirani vmesniki,...)
- ☛ Kompatibilnost med platformami (standardiziran jezik)
- ☛ Interaktivne vsebine, dodatne možnosti uporabniku
- ☛ Off-line dostop
- ☛ Sodoben, objektno usmerjen jezik
- ☛ Veliko razvijalcev
- ☛ Generic Connection Framework ( podpora spremenljivim tehnologijam)

# Slabosti Java 2 ME

- ☛ Programiranje drugačno kot programiranje namiznih aplikacij
- ☛ Najmanjši skupni imenovalec
- ☛ Kompleksen razvoj in posredovanje (v primerjavi z .NET)
- ☛ Ne zagotavlja varnosti med končnimi uporabniki
- ☛ Ni zelo razširjeno oziroma sprejeto