

Lecture Notes And Lab Manual

On

Mobile Application Development (MAD)

MCA III year I semester

Topic: **Introduction to J2ME & J2ME Profiles**

By

HARIKA Y

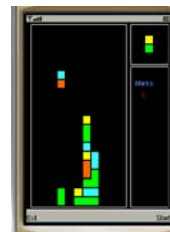
Assistant Professor

MCA Department

JB INSTITUTE OF ENGG & TECHNOLOGY

Hyderabad.

Introduction to J2ME



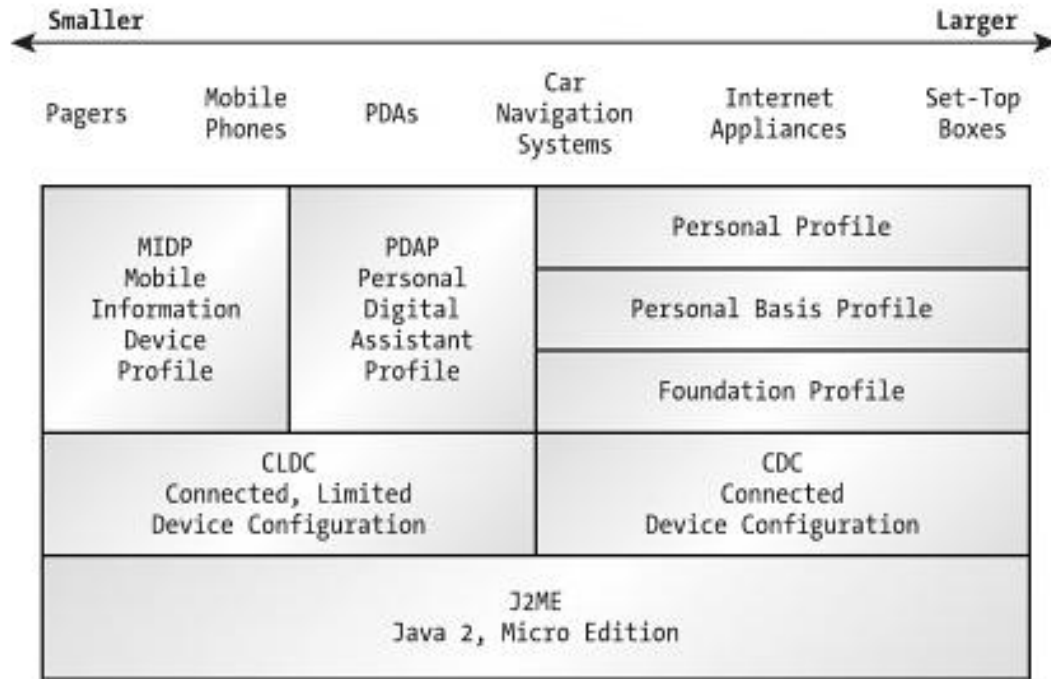
Or

- J2ME provides a robust, flexible environment for applications running on mobile and other embedded devices such as mobile phones, personal digital assistants (PDAs), TV set-top boxes, and printers.
- J2ME includes flexible user interfaces, robust, security, built-in network protocols, and support for networked and offline applications that can be downloaded dynamically.
- Applications based on J2ME are portable across many devices, yet leverage each device's native capabilities.

- **Configurations**

MOBILE APPLICATION DEVELOPMENT NOTES

- Profiles &
- Optional API's

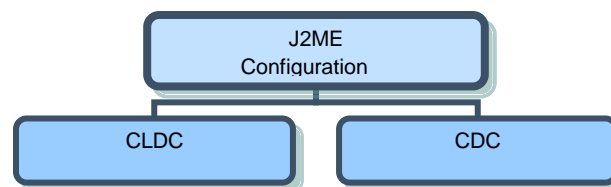


J2ME Configuration :

- **A configuration is a complete Java runtime environment, consisting of:**
 - Java virtual machine (VM) to execute Java bytecode
 - Native code to interface to the underlying system
 - Set of core Java runtime classes

currently J2ME supports two standard configurations:

- Connected Limited Device Configuration (CLDC)
- Connected Device Configuration (CDC)



MOBILE APPLICATION DEVELOPMENT NOTES

- **CDC**
 - Full Java implementation
 - 32 bit
 - Device must have \geq 2MB memory
- **CLDC**
 - Subset of Java
 - 16 or 32 bit
 - Device with 160 – 512 kb memory
- **3 sizes of Virtual Machines are specified**
 - 1MB – 10MB Classic VM (CVM)
 - 10kb – 500kb Kilo VM (kVM)
 - 8bit Card VM

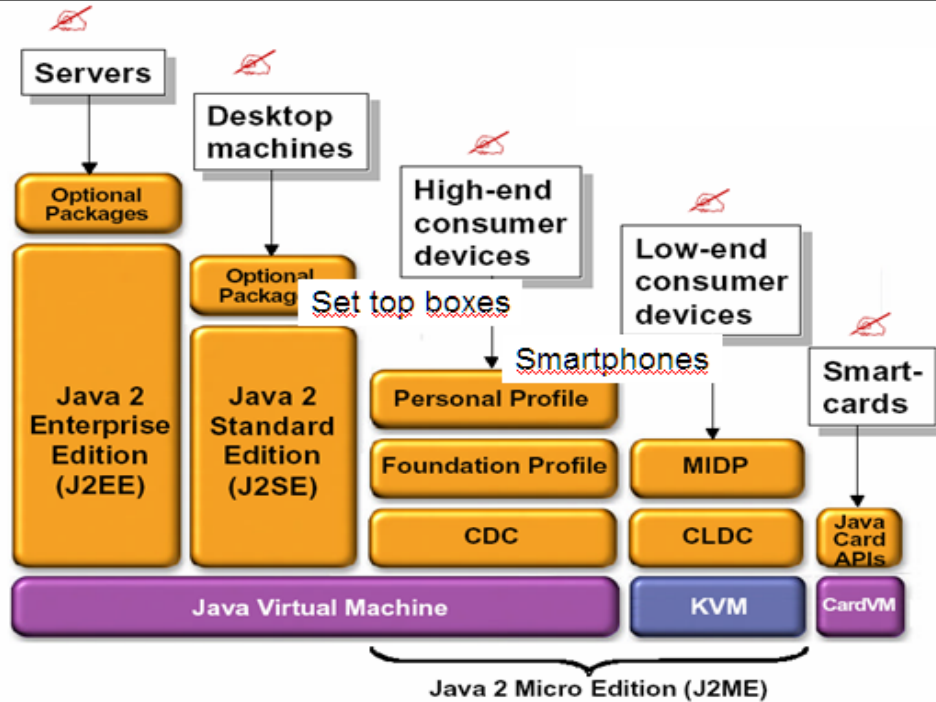
J2ME Profiles :

- **A profile complements a configuration by adding additional classes that provide features appropriate to a particular type of device or to a specific vertical market segment. Both J2ME configurations have one or more associated profiles, some of which may themselves rely on other profiles.**
 - It Guarantees interoperability within device verticals
 - Industry groups specify profiles for each of the configurations (Nokia, AOL, Palm, Oracle, etc)

The Profiles supported by J2ME are

- MIDP Mobile Information Device Profile
- Foundation Profile (non gui networked device)
- Personal basis, Personal and RMI Profiles
- Game Profile

MOBILE APPLICATION DEVELOPMENT NOTES



Mobile Information Device Profile (MIDP)

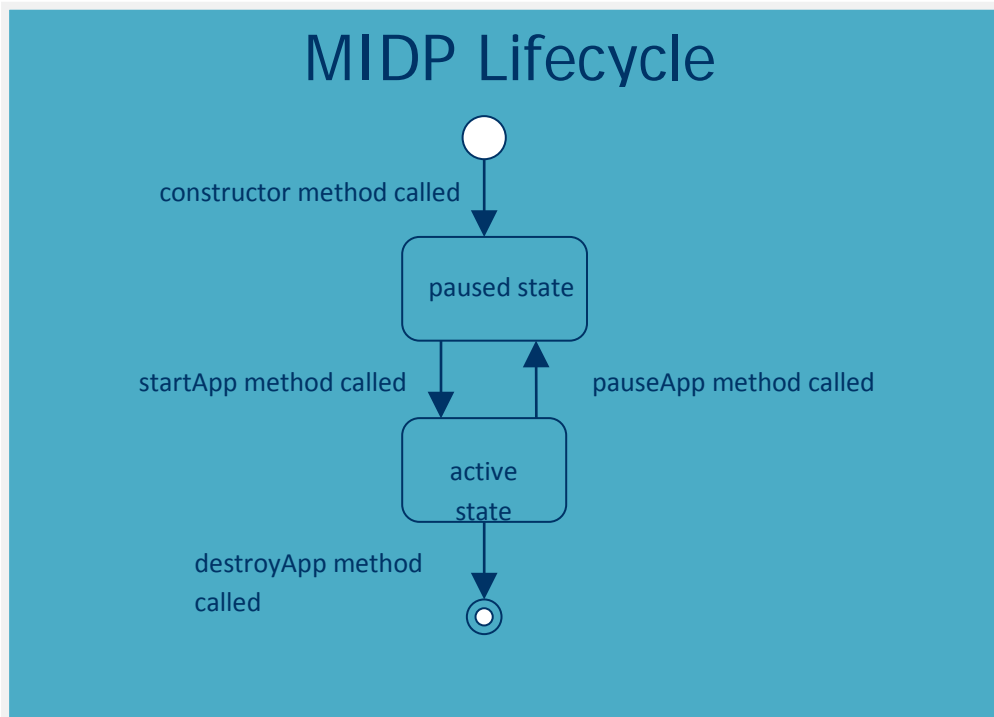
This profile Addresses

- Persistence
- Networking
- Application life-cycle management
- Event handling
- Hardware characteristics
 - Screen size 96 x 54
 - Touch screen and/or 1/2 handed key input
 - 128k non-volatile memory plus 8k for persistence, 32k for Java Runtime
 - Support for 2-way wireless networking

MOBILE APPLICATION DEVELOPMENT NOTES

MIDP Applications are called MIDlets

- MIDP device contains a program called the **Application Management Software** (AMS) which downloads the MIDlet suite from the server, opens the MIDlet suite, then launches the user-specified MIDlet on the MIDP device
- High level API's
 - Textfields, lists, forms and images for programs such as e-commerce applications & basic user interfaces
- Low level API's
 - Incorporate graphics & shapes at precise pixel locations, provides animation for games applications



PDA Profile (PDAP)

MOBILE APPLICATION DEVELOPMENT NOTES

The PDA Profile is similar to MIDP, but it is aimed at PDAs that have better screens and more memory than cell phones.

Foundation Profile

The Foundation Profile extends the CDC to include almost all of the core Java 2 Version 1.3 core libraries. As its name suggests, it is intended to be used as the basis for most of the other CDC profiles.

Personal Basis and Personal Profiles

The Personal Basis Profile adds basic user interface functionality to the Foundation Profile. It is intended to be used on devices that have an unsophisticated user interface capability, and it therefore does not allow more than one window to be active at any time. Platforms that can support a more complex user interface will use the Personal Profile instead.

RMI Profile

The RMI Profile adds the J2SE Remote Method Invocation libraries to the Foundation Profile. Only the client side of this API is supported.

Game Profile This will provide a platform for writing games software on CDC devices.

Book's to be referred:

1) The Complete Reference J2ME

By James Keogh

2) J2ME in a Nutshell

By Kim Topley

O 'REILLY publisher

MOBILE APPLICATIONS DEVELOPMENT
LAB MANUAL

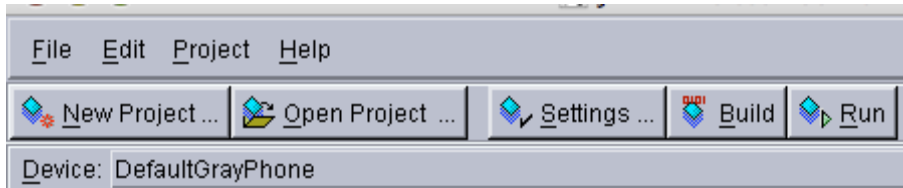
MOBILE APPLICATION DEVELOPMENT NOTES

S.NO	DESCRIPTION OF THE PROGRAM
1	Write programs to implement HelloWorld midlet
2	Write programs to implement multiple midlets
3	Write programs to implement command class
4	Write programs to implement check color
5	Write programs to implement MenuCreation
6	Create a MIDP application,which draws a bargraph to display.Data values can be given integer
7	Create a MIDP application which Examine ,that a phone number,which a user entered in given format.
8	Create a MIDP Application,which draws a Pie Graph to the display.Data Values can be given at int[] array.You can enter four data(integer)values to the input text field.
9	Write a program for quiz midlet
10	Create a slideshow which has three slides.which includes only text.Program should change to the newslide after 5 seconds.After the third slide program returns to the First Slide
11	Create a slideshow which has three slides,which includes pictures at PNG format.Program should change to the new slide other 5 seconds.
12	Write program for creating datagram
13	Write program for creating login file

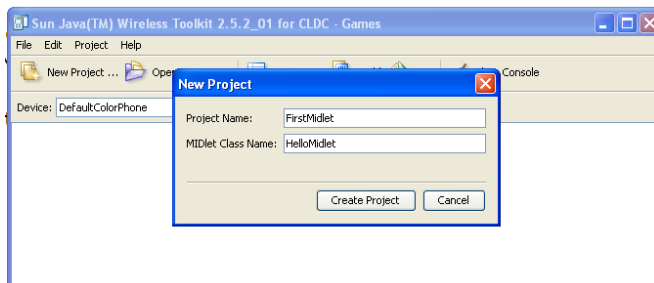
MOBILE APPLICATION DEVELOPMENT NOTES

Prg-1:-Printing HelloWorld program

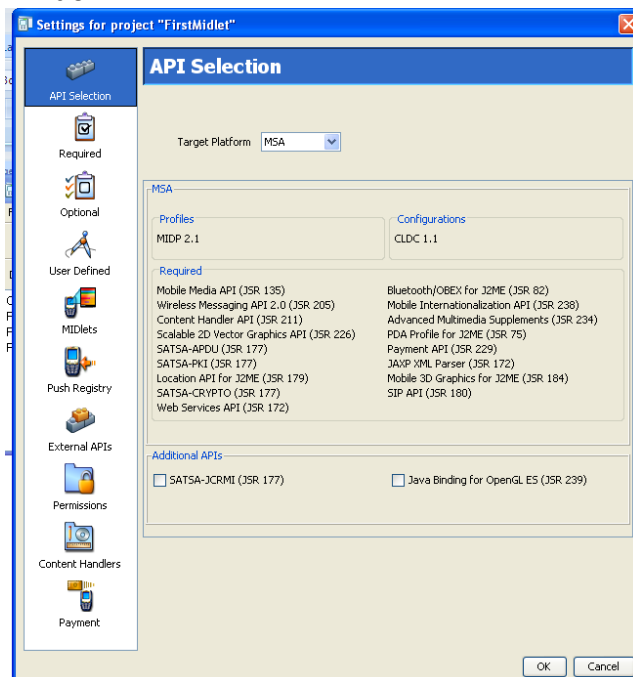
Step-1:-Start ->AllPrograms->Sun Java Tool Kit->Wireless Tool Kit



Step-2:-Click New Project –Enter project Name as FirstMidlet -> Enter ClassName as HelloMidlet->click on Create Project



Step-3:- A setting window will open up. Accept the defaults by clicking ok in that window.



MOBILE APPLICATION DEVELOPMENT NOTES

Step-4:-Write Following Code in Notepad and save it as HelloMidlet.java

```
import javax.microedition.midlet.*;
import javax.microedition.lcdui.*;
public class HelloMidlet extends MIDlet {
    public HelloMidlet() {
    }
    public void startApp() {
        Form form = new Form( "First Program" );
        form.append( "Hello World" );
        Display.getDisplay(this).setCurrent( form );
    }
    public void pauseApp() {
    }
    public void destroyApp( boolean unconditional ) {
    }
}
```

Step-5:-Place HelloMidlet.java in C:\Documents and settings\ADMIN\j2mewtk\2.5.2\apps\FirstMidlet\src\

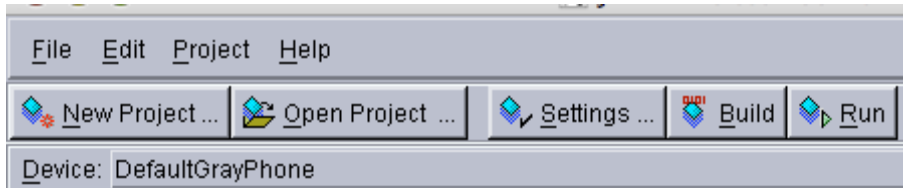
Step-6 :In the ktoolbar main window click on the “Build” button. When the build compiles successfully then click on the “Run” button.



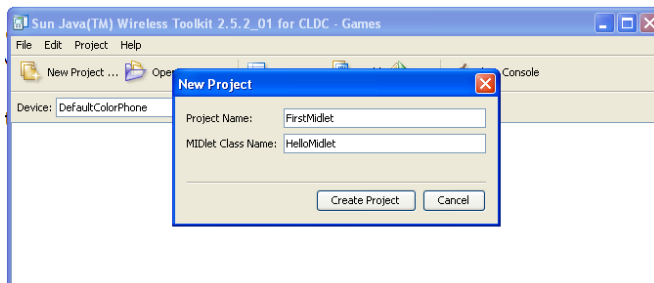
MOBILE APPLICATION DEVELOPMENT NOTES

Program 2: creating multiple midlets in single project

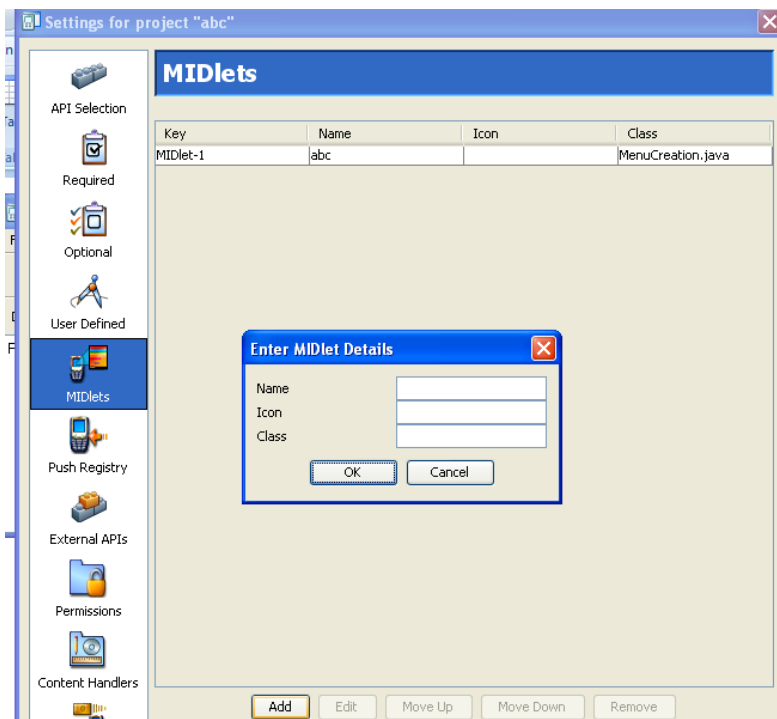
Step-1:-Start ->AllPrograms->Sun Java Tool Kit->Wireless Tool Kit



Step-2:-Click New Project –Enter project Name as FirstMidlet -> Enter ClassName as HelloMidlet->click on Create Project



Step 3: click settings button and select midlet- >add give midlet name , class name



MOBILE APPLICATION DEVELOPMENT NOTES

Step 4: write a following code in source folder

Midlet 1:

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

import javax.microedition.lcdui.*;

public class HelloMidlet extends MIDlet {

    public ByeMidlet() {

    }

    public void startApp() {

        Form form = new Form( "First Program" );

        form.append( "Hello World" );

        Display.getDisplay(this).setCurrent( form );

    }

    public void pauseApp() {

    }

    public void destroyApp( boolean unconditional ) {

    }

}
```

Step 5: write a following code with different class name in source folder

Midlet 2:

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

import javax.microedition.lcdui.*;

public class HelloMidlet extends MIDlet {
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
public HelloMidlet() {  
  
}  
  
public void startApp() {  
  
    Form form = new Form( "First Program" );  
  
    form.append( "good bye" );  
  
    Display.getDisplay(this).setCurrent( form );  
  
}  
  
public void pauseApp() {  
  
}  
  
public void destroyApp( boolean unconditional ) {  
  
}  
  
}
```

Step 6: click BUILD button

Step 7: if build successfully completed then click RUN button



MOBILE APPLICATION DEVELOPMENT NOTES

Program 3: write a program for implementing command class

Step 1,2,3 same as above program

Step 4: write down the following code in the source folder

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

import javax.microedition.lcdui.*;

public class OnlineHelp extends MIDlet implements CommandListener
{
    private Display dis;

    private Command back;

    private Command exit;

    private Command help;

    private Form form;

    private TextBox hm;

    public OnlineHelp()
    {
        dis=Display.getDisplay(this);

        back=new Command("BACK",Command.BACK,2);

        exit=new Command("EXIT",Command.EXIT,1);

        help=new Command("HELP",Command.HELP,3);

        form=new Form("online help");

        hm=new TextBox("online help","press back to return to the previous screen or press
        exit to close this program",81,0);

        hm.addCommand(back);
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
form.addCommand(help);

form.addCommand(exit);

form.setCommandListener(this);

hm.setCommandListener(this);

}

public void startApp()

{

dis.setCurrent(form);

}

public void pauseApp()

{

}

public void destroyApp(boolean unconditional)

{

}

public void commandAction(Command c , Displayable d)

{

if(c==back)

{

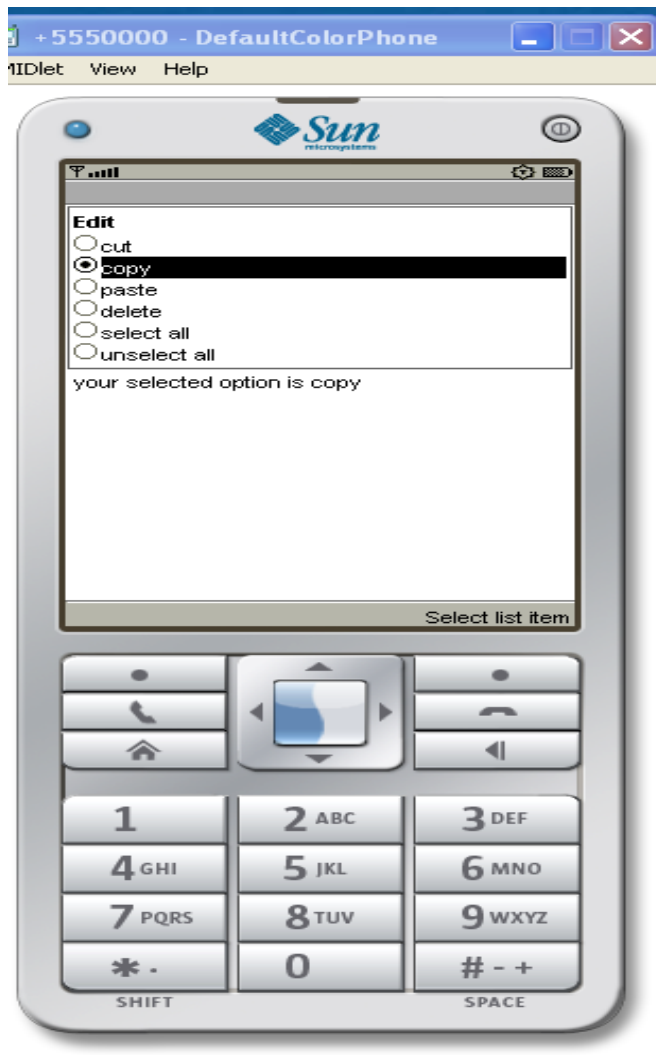
dis.setCurrent(form);

}

}
```


MOBILE APPLICATION DEVELOPMENT NOTES

```
else if(c==exit)
{
destroyApp(false);
notifyDestroyed();
}
else if(c==help)
{dis.setCurrent(hm);
}}}
```



Step 5: click BUILD button

Step 6: if build successfully completed then click RUN button

MOBILE APPLICATION DEVELOPMENT NOTES

Program 4: write a program for check color

Step 1,2,3 same as above program

Step 4: write the following code in the source folder

```
package greeting;

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

public class Checkcolor123 extends MIDlet implements CommandListener
{
    private Display display;
    private Form form;
    private TextBox textbox;
    private Command exit;
    public Checkcolor123()
    {
        display=Display.getDisplay(this);
        exit=new Command("Exit",Command.SCREEN,1);
        String message=null;
        if(display.isColor())
        {
            message="Color display";
        }
        else
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
{  
message="No color display";  
}  
textbox=new TextBox("Check Colors",message,17,0);  
textbox.addCommand(exit);  
textbox.setCommandListener(this);  
}  
public void startApp()  
{  
display.setCurrent(textbox);  
}  
public void pauseApp()  
{  
}  
public void destroyApp(boolean unconditional)  
{  
}  
public void commandAction(Command command,Displayable diasplayable)  
{  
if(command==exit)  
{  
destroyApp(true);  

```

MOBILE APPLICATION DEVELOPMENT NOTES

```
notifyDestroyed();
```

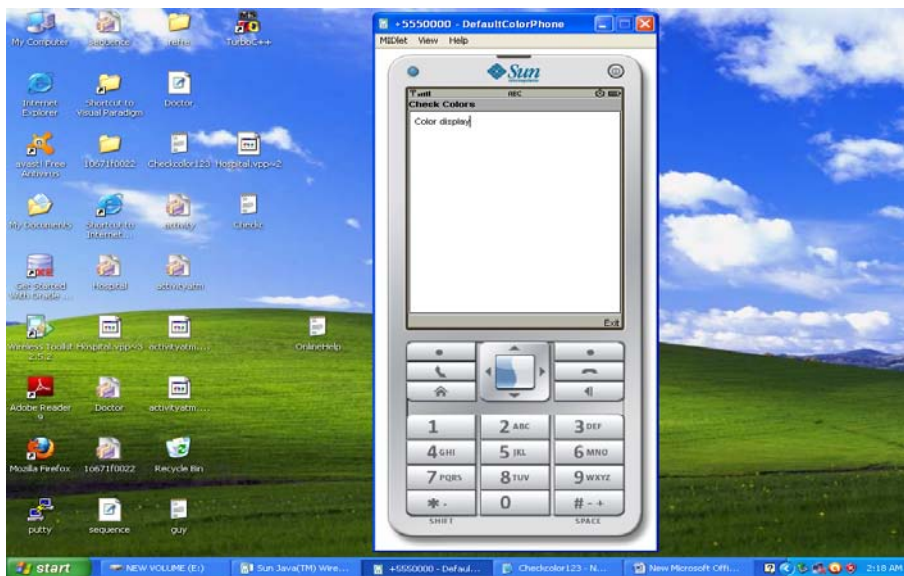
```
}
```

```
}
```

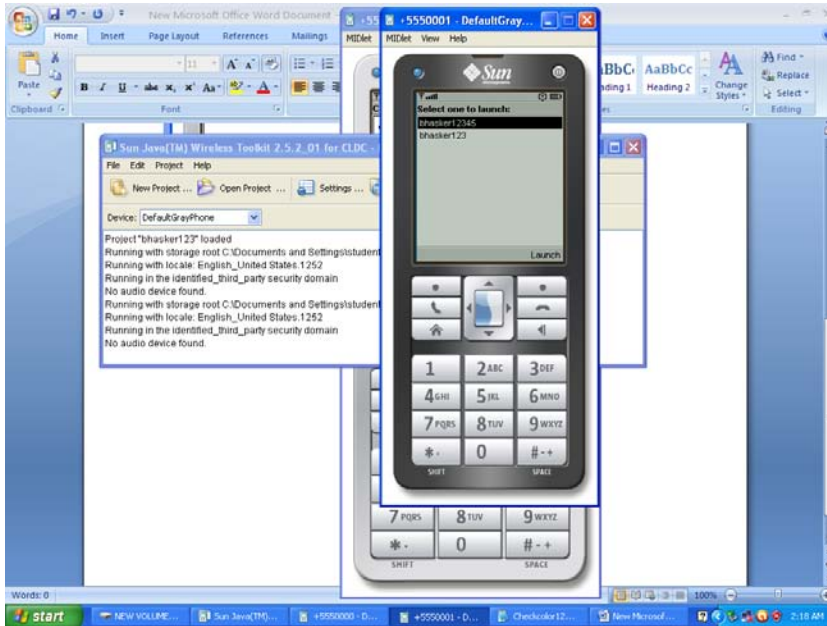
```
}
```

Step 5: click BUILD button

Step 6: if build successfully completed then click RUN button



MOBILE APPLICATION DEVELOPMENT NOTES



Program 5: program for creating menu events

Step 1,2,3 same as above program

Step 4: write the following code in the source folder

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

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

```
/**
```

```
 * @author ADMIN
```

```
 */
```

```
public class MenuEvents extends MIDlet implements  
CommandListener,ItemStateListener {
```

```
    public ChoiceGroup ch;
```

```
    public ChoiceGroup ch1;
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
public Form form;

public Form form1;

public Display display;

public Command View;

public Command Exit;

public Command Back;

public StringItem options;

public Item item;

public MenuEvents()
{
    display=Display.getDisplay(this);

    form=new Form("");

    form1=new Form("Selcted Options are");

    ch=new ChoiceGroup("Preferences",Choice.MULTIPLE);

    ch.append("cut",null);

    ch.append("copy",null);

    ch.append("paste",null);

    ch.append("delete",null);

    ch.setSelectedIndex(1, true);

    form.append(ch);

    ch1=new ChoiceGroup("",Choice.EXCLUSIVE);

    ch1.append("select all",null);
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
ch1.append("unselect all",null);

ch1.setSelectedIndex(1, true);

form.append(ch1);

View=new Command("View",Command.OK,1);

Exit =new Command("Exit",Command.EXIT,1);

Back=new Command("Back",Command.BACK,1);

form.addCommand(View);

form.addCommand(Exit);

form1.addCommand(Back);

form.setCommandListener(this);

form1.setCommandListener(this);

form.setItemStateListener(this);

}

public void startApp() {

    display.setCurrent(form);

}

public void pauseApp() {

}

public void destroyApp(boolean unconditional) {

}
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
public void commandAction(Command command, Displayable displayable)
{
    if(displayable==form)
    {
        if(command==View)
        {
            boolean opt[]=new boolean[ch.size()];
            options=new StringItem("", "");
            String values="";
            ch.getSelectedFlags(opt);
            options.setText("");

            for(int i=0;i<opt.length;i++)
            {
                if(opt[i])
                {
                    values+=ch.getString(i)+"\n";
                }
            }
            options.setText(values);
            form1.append(options);
        }
    }
}
```


MOBILE APPLICATION DEVELOPMENT NOTES

```
        display.setCurrent(form1);
    }
else if(command==Exit)
    {

        destroyApp(true);
        notifyDestroyed();
    }
    }
else if(displayable==form1)
    {
        if(command==Back)
        {
            display.setCurrent(form);
            options.setText("");
        }
    }

    }

    }

    public void itemStateChanged(Item item)
    {
```

MOBILE APPLICATION DEVELOPMENT NOTES

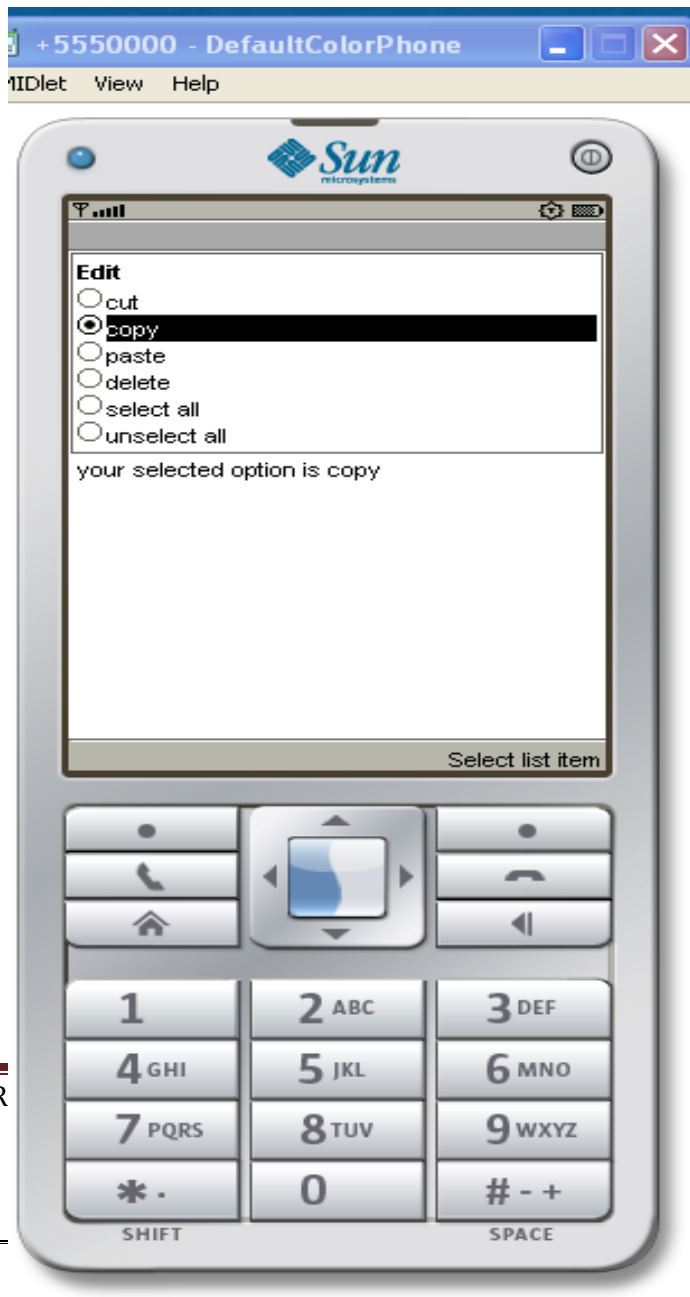
```
if(item==ch1)
{
    int i=0;
    int size=ch.size();
    while(i<size)
    {
        if(ch1.getSelectedIndex()==0)
            true);
        ch.setSelectedIndex(i,
        else
        ch.setSelectedIndex(i,
        i++;
    }
}
}
```

Step

Step

RUN

PREPAR



5: click BUILD button

6: if build successfully completed then click button

MOBILE APPLICATION DEVELOPMENT NOTES

Program6:-

Create a MIDP application, which draws a bargraph to display. Data values can be given at int[] array?

Step 1,2,3 same as above program

Step4: write down following code

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

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

```
public class BarGraphMIDlet extends MIDlet implements CommandListener{
```

```
    public Form form;
```

```
    public Command exitCommand;
```

```
    public Command OkCommand;
```

```
    public Command backCommand;
```

```
    public Displayable d;
```

```
    public Display display;
```

```
    public TextField textfield1;
```

```
    public TextField textfield2;
```

```
    public TextField textfield3;
```

```
    public TextField textfield4;
```

```
    public TextField textfield5;
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
public BarGraphMIDlet()
{
    display=Display.getDisplay(this);

    form=new Form("BarGraph");

    textfield1=new TextField("Value1:-"," ",30,TextField.ANY);
    textfield2=new TextField("Value2:-"," ",30,TextField.ANY);
    textfield3=new TextField("Value3:-"," ",30,TextField.ANY);
    textfield4=new TextField("Value4:-"," ",30,TextField.ANY);
    textfield5=new TextField("Value5:-"," ",30,TextField.ANY);

    form.append(textfield1);
    form.append(textfield2);
    form.append(textfield3);
    form.append(textfield4);
    form.append(textfield5);

    OkCommand=new Command("Ok",Command.OK,1);
    exitCommand=new Command("Exit",Command.EXIT,1);
    backCommand=new Command("Back",Command.BACK,1);

    form.addCommand(OkCommand);
    form.addCommand(exitCommand);
    form.setCommandListener(this);

}
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
public void startApp() {  
    display.setCurrent(form);  
}
```

```
public void pauseApp() {  
}
```

```
public void destroyApp(boolean unconditional) {  
}
```

```
public void commandAction(Command command, Displayable displayable)  
{  
    if(displayable==form)  
    {  
        if(command==OkCommand)  
        {  
            int[] data=new int[5];  
            data[0]=Integer.parseInt(textfield1.getString());  
            data[1]=Integer.parseInt(textfield2.getString());  
            data[2]=Integer.parseInt(textfield3.getString());  
            data[3]=Integer.parseInt(textfield4.getString());  
            data[4]=Integer.parseInt(textfield5.getString());  
            d=new BarCanvas(data);
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
        d.addCommand(backCommand);

        d.setCommandListener(this);

        display.setCurrent(d);
    }

    else if(command==exitCommand)

        notifyDestroyed();

    }

    else if(displayable==d)

    {

        if(command==backCommand)

            display.setCurrent(form);

        }

    }

}
```

```
class BarCanvas extends Canvas{
```

```
    int[] data;

    public int x;

    public int y;

    public int y1;
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
public int h;

public BarCanvas(int[] data)
{
    this.data=data;

    x=10;
}

public void paint(Graphics g)
{
    g.setColor(255, 255, 255);
    g.fillRect(0, 0, this.getWidth(), this.getHeight());
    g.setColor(255, 125, 100);

    int i=0;
    y1=data[0];
    h=200;
    while(i<data.length)
    {
        y=data[i];
        h=200+y1-y;
        g.fillRect(x, y, 25, h);
        x+=30;
        i++;
    }
}
```

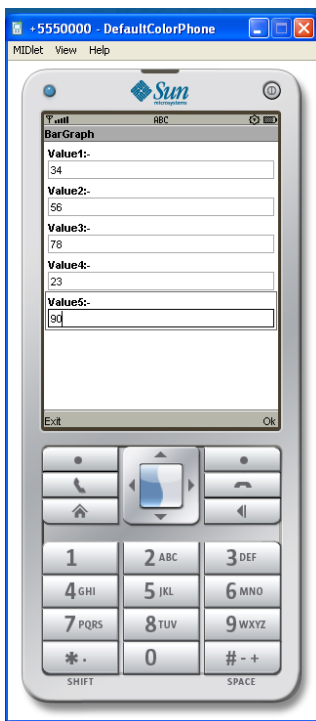
MOBILE APPLICATION DEVELOPMENT NOTES

}

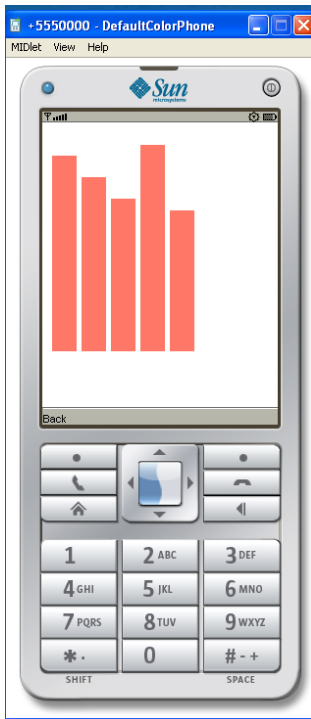
}

Step 5: click BUILD button

Step 6: if build successfully completed then click RUN button



MOBILE APPLICATION DEVELOPMENT NOTES



Program 7 :- Create a MIDP application which Examine ,that a phone number,which a user entered in given format.

*Area code should be one of the following :040,041,050,0400,044

Step 1,2,3 same as above program

Step 4: write down the following code in the source folder

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

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

MOBILE APPLICATION DEVELOPMENT NOTES

```
public class InputChecking extends MIDlet implements CommandListener {
```

```
    public Form form1;
```

```
    public TextField textfield1;
```

```
    public Command exitCommand;
```

```
    public Command okCommand;
```

```
    public StringItem st;
```

```
    public Display display;
```

```
    public InputChecking()
```

```
    {
```

```
        display=Display.getDisplay(this);
```

```
        form1=new Form("Insert the Phone number");
```

```
        exitCommand=new Command("Exit",Command.EXIT,1);
```

```
        okCommand=new Command("Ok",Command.OK,1);
```

```
        st=new StringItem("Phone Number is ","");
```

```
        textfield1=new TextField("Phone;","",30,TextField.ANY);
```

```
        form1.append(textfield1);
```

```
        form1.addCommand(okCommand);
```

```
        form1.addCommand(exitCommand);
```

```
        form1.setCommandListener(this);
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
}
```

```
public void startApp() {  
    display.setCurrent(form1);  
}
```

```
public void pauseApp() {  
}
```

```
public void destroyApp(boolean unconditional) {  
}
```

```
public void commandAction(Command cmd, Displayable displayable)  
{  
    if(cmd==exitCommand)  
        notifyDestroyed();  
    else if(cmd==okCommand)  
    {  
  
        String s=textfield1.getString();  
  
        s=s.replace(' ', '.');
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
int len=s.length();

int i=0;

int c=0;


String s1="";

while(i<len)

{

    if(s.charAt(i)=='.')

    {

        if(c==0)

        {

            if(s1.equals("040") || s1.equals("041") || s1.equals("050") ||
s1.equals("0400") || s1.equals("044"))

            {

                c++;

                s1="";

            }

        }

    }

}
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
        if(c==1)
        {
            if(s1.length()-1==3)
            {
                c++;
                s1="";
            }
            s1=s1+s.charAt(i);
            i++;
        }

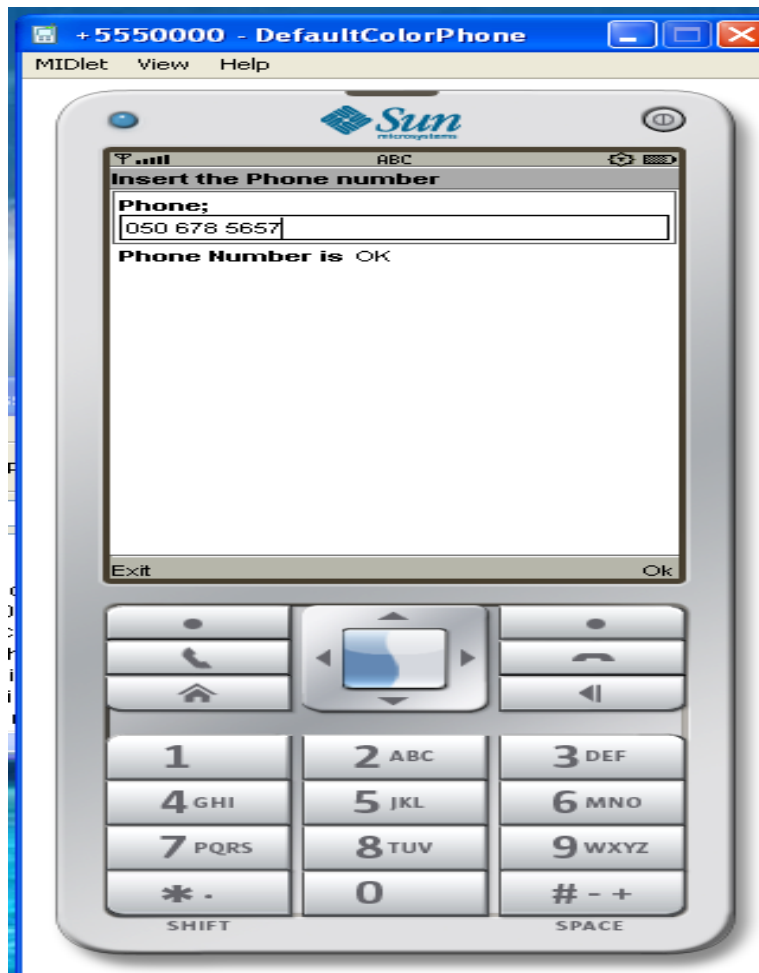
        if(s1.length()-1==3 || s1.length()-1==4 || s1.length()-1==5)
        {
            c++;
            if(c==3)
            {
                st.setText("OK");
            }
            else
            {
                st.setText("wrong\n Phone Number Format is xxx xxx xxxx\nArea code must be
040|050|041|0400|044");
            }
            form1.append(st);
        }
    }
}
```

Step 5: click build

MOBILE APPLICATION DEVELOPMENT NOTES

Step 6: click run

OutPUT:



- Program 8:-

Create a MIDP Application, which draws a Pie Graph to the display. Data Values can be given at `int[]` array. You can enter four data(integer) values to the input text field.

Step 1,2,3 same as above program

Step 4: write down the following code in the source folder

MOBILE APPLICATION DEVELOPMENT NOTES

```
import javax.microedition.midlet.MIDlet;

import javax.microedition.lcdui.*;

public class PieChartMIDlet extends MIDlet implements CommandListener {

    public Form form;

    public Command exitCommand;

    public Command OkCommand;

    public Display display;

    public TextField textfield1;

    public TextField textfield2;

    public TextField textfield3;

    public TextField textfield4;

    public TextField textfield5;

    public Displayable d;

    public void startApp() {

        display = Display.getDisplay(this);

        form=new Form("Draw Pie");

        textfield1=new TextField("Value1:-"," ",30,TextField.ANY);

        textfield2=new TextField("Value2:-"," ",30,TextField.ANY);

        textfield3=new TextField("Value3:-"," ",30,TextField.ANY);

        textfield4=new TextField("Value4:-"," ",30,TextField.ANY);

        textfield5=new TextField("Value5:-"," ",30,TextField.ANY);

        form.append(textfield1);
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
form.append(textfield2);

form.append(textfield3);

form.append(textfield4);

form.append(textfield5);

exitCommand = new Command("exit", Command.EXIT, 1);

OkCommand=new Command("Ok",Command.OK,1);

form.addCommand(OkCommand);

form.addCommand(exitCommand);

form.setCommandListener(this);

display.setCurrent(form);

}

public void pauseApp() {

}

public void destroyApp(boolean unconditional) {

}

public void commandAction(Command c, Displayable s) {

    if(s==form)

    {

        if(c==exitCommand)

        notifyDestroyed();

        else if(c==OkCommand)

        {
```


MOBILE APPLICATION DEVELOPMENT NOTES

```
int[] data = new int[5];

data[0]=Integer.parseInt(textfield1.getString());
data[1]=Integer.parseInt(textfield2.getString());
data[2]=Integer.parseInt(textfield3.getString());
data[3]=Integer.parseInt(textfield4.getString());
data[4]=Integer.parseInt(textfield5.getString());

d = new PieChartCanvas(data);
d.addCommand(exitCommand);
d.setCommandListener(this);

    display.setCurrent(d);
}
}

else if(s==d)
{
    if(c==exitCommand)
        display.setCurrent(form);
}
}
}
```

```
class PieChartCanvas extends Canvas {
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
int[] data;

int colors[] = { 0xFF0000, 0xA9E969, 0x00FFFF, 0xC675EC, 0x008800, 0x00C400 };

public PieChartCanvas(int[] data) {

    this.data = data;

}

public void paint(Graphics g) {

    int width = this.getWidth();

    int height = this.getHeight();

    g.setColor(255, 255, 255);

    g.fillRect(0, 0, width, height);

    int sum = 0;

    for (int i = 0; i < data.length; i++) {

        sum += data[i];

    }

    int deltaAngle = 360 * 100 / sum / 100;

    int x = 4;

    int y = 4;

    int diameter;

    if (width > height)

        diameter = height - y * 2;

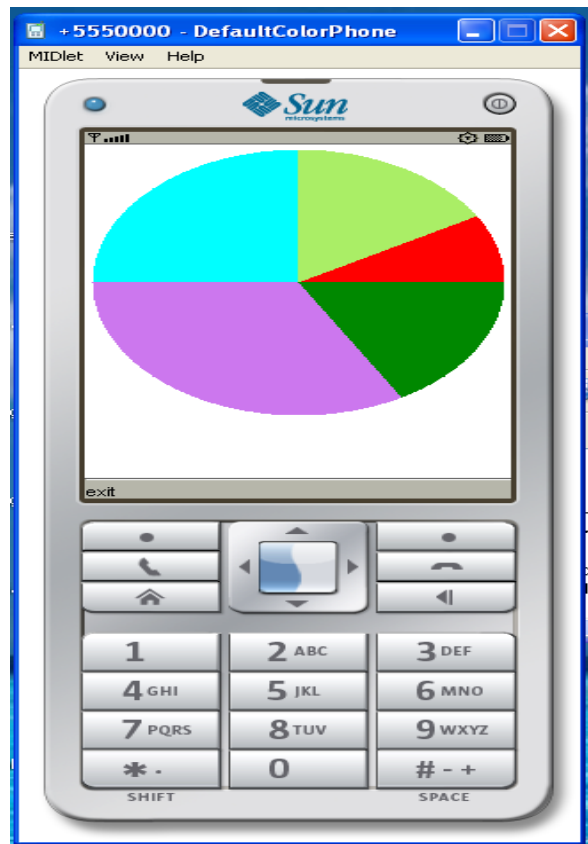
    else
```

MOBILE APPLICATION DEVELOPMENT NOTES

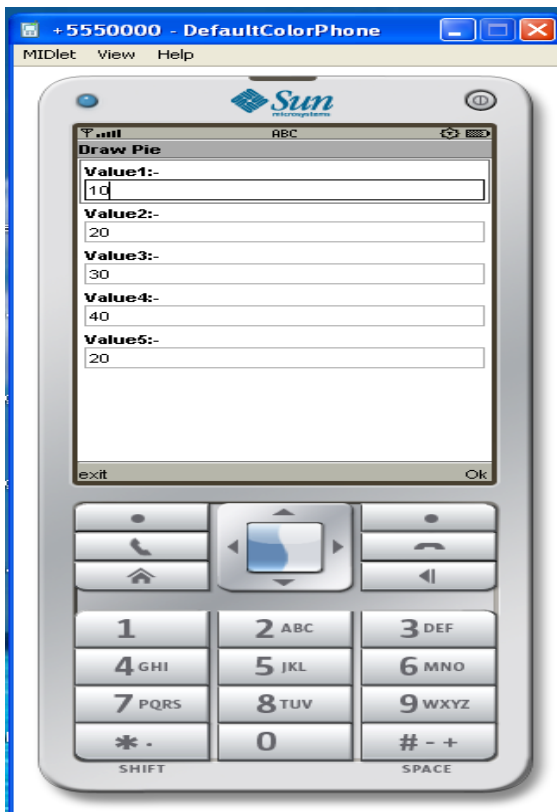
```
diameter = width - x * 2;  
  
int startAngle = 0;  
  
for (int i = 0; i < data.length; i++) {  
  
    g.setColor(colors[i]);  
  
    g.fillArc(x, y, diameter, diameter, startAngle, deltaAngle * data[i]);  
  
    startAngle += deltaAngle * data[i];  
  
    }  
}
```

Step 5: click build

Step6: click run



MOBILE APPLICATION DEVELOPMENT NOTES



program 9: program for quiz midlet

Step 1,2,3 same as above program

Step 4: write down the following code in the source folder

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

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

```
import javax.microedition.rms.*;
```

```
import java.io.*;
```

```
/**
```

```
* @author ADMIN
```

```
*/
```

```
public class QuizMidlet extends MIDlet implements CommandListener {
```

```
    public Form form1;
```

```
    public Form form2;
```

```
    public Form form3;
```

```
    public Form form4;
```

```
    public Form form5;
```

```
    public Form form6;
```

```
    public Form form7;
```

```
    public ChoiceGroup ch1;
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
public ChoiceGroup ch2;

public ChoiceGroup ch3;;

public ChoiceGroup ch4;;

public ChoiceGroup ch5;;

public Command nextCommand;

public Command backCommand;

public Command MenuCommand;

public Command OkCommand;

public Command ExitCommand;

public Command sCommand;

public Display display;

public StringItem st;

public TextField textfield;

public int count;

public RecordStore recordstore=null;

public RecordEnumeration re=null;

public Alert alert;

public Compare comp;

public StringItem st1;

public QuizMidlet()

{

    count=0;
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
display=Display.getDisplay(this);

nextCommand=new Command("Next",Command.OK,1);

backCommand=new Command("Back",Command.BACK,1);

OkCommand=new Command("Save",Command.SCREEN,1);

ExitCommand=new Command("Exit",Command.SCREEN,1);

sCommand=new Command("TopScores",Command.SCREEN,1);

st=new StringItem("TotalPoints","0");

st1=new StringItem("", "");

textfield=new TextField("EnterName","",20,TextField.ANY);

form1=new Form("J2ME Stands for");

form2=new Form("a+b=");

form3=new Form("5*5");

form4=new Form("Who is AP CM");

form5=new Form("How many Districts in AP");

form6=new Form("Score");

form7=new Form("Top Scoreers");

ch1=new ChoiceGroup("",Choice.EXCLUSIVE);

ch1.append("Java 2 Mobile Edition", null);

ch1.append("Java 2 Macro Edition", null);

ch1.append("Java 2 Micro Edition", null);

ch1.append("Java 2 Music Edition", null);

form1.append(ch1);
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
form1.addCommand(nextCommand);

form1.setCommandListener(this);

ch2=new ChoiceGroup("",Choice.EXCLUSIVE);

ch2.append("b+a", null);

ch2.append("b*a", null);

ch2.append("b/a", null);

ch2.append("b-a", null);

form2.append(ch2);

form2.addCommand(nextCommand);

form2.addCommand(backCommand);

form2.setCommandListener(this);

ch3=new ChoiceGroup("",Choice.EXCLUSIVE);

ch3.append("20", null);

ch3.append("30", null);

ch3.append("10", null);

ch3.append("25", null);

form3.append(ch3);

form3.addCommand(nextCommand);

form3.addCommand(backCommand);

form3.setCommandListener(this);


ch4=new ChoiceGroup("",Choice.EXCLUSIVE);
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
ch4.append("Rosiah", null);

ch4.append("Jagan", null);

ch4.append("ChandaBabu", null);

ch4.append("Kiran", null);

form4.append(ch4);

form4.addCommand(nextCommand);

form4.addCommand(backCommand);

form4.setCommandListener(this);

ch5=new ChoiceGroup("",Choice.EXCLUSIVE);

ch5.append("8", null);

ch5.append("4", null);

ch5.append("11", null);

ch5.append("23", null);

form5.append(ch5);

form5.addCommand(backCommand);

form5.addCommand(nextCommand);

form5.setCommandListener(this);

form6.addCommand(OkCommand);

form6.addCommand(ExitCommand);

    form6.addCommand(sCommand);

form6.setCommandListener(this);

form7.addCommand(backCommand);
```


MOBILE APPLICATION DEVELOPMENT NOTES

```
form7.setCommandListener(this);

try
{
    recordstore=RecordStore.openRecordStore("Quiz", true);
}
catch(Exception ex)
{

}

}

public void startApp() {
    display.setCurrent(form1);
}

public void pauseApp() {
}

public void destroyApp(boolean unconditional) {
}

public void commandAction(Command cmd,Displayable displayable)
{
}
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
    if(displayable==form1)
    {
        if(cmd==nextCommand)
            display.setCurrent(form2);

    }
else if(displayable==form2)
    {

        if(cmd==nextCommand)
            display.setCurrent(form3);
        else if(cmd==backCommand)
            display.setCurrent(form1);
    }
    else if(displayable==form3)
    {
        if(cmd==nextCommand)
            display.setCurrent(form4);
        else if(cmd==backCommand)
            display.setCurrent(form2);
    }
    else if(displayable==form4)
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
{  
if(cmd==nextCommand)  
    display.setCurrent(form5);  
else if(cmd==backCommand)  
    display.setCurrent(form3);  
}  
  
else if(displayable==form5)  
{  
if(cmd==nextCommand)  
{  
  
    if(ch1.getSelectedIndex()==2)  
        count++;  
    if(ch2.getSelectedIndex()==0)  
        count++;  
    if(ch3.getSelectedIndex()==3)  
        count++;  
    if(ch4.getSelectedIndex()==3)  
        count++;  
    if(ch5.getSelectedIndex()==3)  
        count++;  
}
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
st.setText(String.valueOf(count));

form6.append(st);

form6.append(textfield);

display.setCurrent(form6);
}

else if(cmd==backCommand)

display.setCurrent(form4);
}

else if(displayable==form6)

{

if(cmd==OkCommand)

{

try

{

String Pname=textfield.getString();

int Points=Integer.parseInt(st.getText());

byte[] bytes;

ByteArrayOutputStream ostream=new ByteArrayOutputStream();

DataOutputStream dstream =new DataOutputStream(ostream);

dstream.writeUTF(Pname);

dstream.writeInt(Points);
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
        dstream.flush();

        bytes=ostream.toByteArray();

        recordstore.addRecord(bytes, 0,bytes.length );

        ostream.reset();

        ostream.close();

        dstream.close();

        alert =new Alert("Message","Saved",null,AlertType.INFO);

        alert.setTimeout(Alert.FOREVER);

        display.setCurrent(alert);
    }

    catch(Exception ex)

    {

        alert =new Alert("Message",ex.toString(),null,AlertType.INFO);

        alert.setTimeout(Alert.FOREVER);

        display.setCurrent(alert);

    }

}

else if(cmd==ExitCommand)

{

    try

    {

        recordstore.closeRecordStore();
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
notifyDestroyed();

    }

    catch(Exception ex)

    {

    }

    }

    else if(cmd==sCommand)

    {

        try

        {

            byte[] bytes=new byte[300];

            ByteArrayInputStream bstream=new ByteArrayInputStream(bytes);

            DataInputStream dstream=new DataInputStream(bstream);

            StringBuffer sb=new StringBuffer();

            comp=new Compare();

            re=recordstore.enumerateRecords(null, comp,false);

st1.setText("");

            while(re.hasNextElement())

            {

                recordstore.getRecord(re.nextRecordId(), bytes, 0);

                sb.append(dstream.readUTF()+"|"+dstream.readInt());

                sb.append("\n");

            }

        }

    }

}
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
        dstream.reset();
    }

    bstream.close();
    dstream.close();

    st1.setText(sb.toString());
    form7.append(st1);
}
catch(Exception ex)
{
    alert =new Alert("Msg",ex.toString(),null,AlertType.INFO);
    alert.setTimeout(Alert.FOREVER);
    display.setCurrent(alert);
}    display.setCurrent(form7);
}

}

else if(displayable==form7)
{
    if(cmd==backCommand)
        display.setCurrent(form6);
}
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
    }  
}  
  
class Compare implements RecordComparator  
{  
    public byte[] bytedata=new byte[300];  
    public ByteArrayInputStream bstream=null;  
    public DataInputStream dstream=null;  
    public int compare(byte[] r1,byte[] r2)  
    {  
        int r1int,r2int;  
        int or=0;  
        try  
        {  
            int maxlen=Math.max(r1.length, r2.length);  
            if(maxlen>bytedata.length)  
            {  
                bytedata=new byte[maxlen];  
            }  
            bstream=new ByteArrayInputStream(r1);  
            dstream=new DataInputStream(bstream);  
            dstream.readUTF();  
            r1int=dstream.readInt();
```


MOBILE APPLICATION DEVELOPMENT NOTES

```
bstream=new ByteArrayInputStream(r2);  
dstream=new DataInputStream(bstream);  
dstream.readUTF();  
r2int=dstream.readInt();  
if(r1int==r2int)  
{  
    or= RecordComparator.EQUIVALENT;  
}  
else if(r1int>r2int)  
{  
    or= RecordComparator.PRECEDES;  
}  
else if(r1int<r2int)  
{  
    or= RecordComparator.FOLLOWS;  
}  
return or;  
}  
catch(Exception ex)  
{  
    return RecordComparator.EQUIVALENT;  
}
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
}

public void compareClose()
{
    try
    {
        if(bstream!=null)
        {
            bstream.close();
        }
        if(dstream!=null)
        {
            dstream.close();
        }
    }
    catch(Exception ex)
    {

    }
}
}
```

Step 5: click build

MOBILE APPLICATION DEVELOPMENT NOTES

Step 6: click run

Output:

Program 10:-

Create a slideshow which has three slides.which includes only text.Program should change to the newslide after 5 seconds.After the third slide program returns to the First Slide

Step 1,2,3 same as above program

Step 4: write down the following code in the source folder

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

import javax.microedition.lcdui.*;

public class SlideShow extends MIDlet implements CommandListener {

    public Form slide1;

    public Form slide2;

    public Form slide3;

    public Command Exit;

    public Display display;

    public SlideShow()

    {

        display=Display.getDisplay(this);
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
Exit=new Command("Exit",Command.EXIT,1);

slide1=new Form("Slide1");

slide1.append("This is Slide number 1");

slide1.addCommand(Exit);

slide2=new Form("Slide2");

slide2.append("This is Slide number 2");

slide2.addCommand(Exit);

slide3=new Form("Slide3");

slide3.append("This is Slide number 3");

slide3.addCommand(Exit);

slide1.setCommandListener(this);

slide2.setCommandListener(this);

slide3.setCommandListener(this);

}

public void startApp() {

    Thread runner = new Thread(new ThreadRunner(display,slide1,slide2,slide3));

    runner.start();

}

public void pauseApp() {

}

public void destroyApp(boolean unconditional) {

}
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
public void commandAction(Command command, Displayable displayable)
{
    if(displayable==slide1)
    {
        if(command==Exit)
            notifyDestroyed();
    }
else if(displayable==slide2)
    {
        if(command==Exit)
            notifyDestroyed(); }
else if(displayable==slide3)
    {
        if(command==Exit)
            notifyDestroyed();
    }
}

class ThreadRunner implements Runnable {
    Display display;
    public int c=0;
    public Form slide1;
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
public Form slide2;

public Form slide3;

public ThreadRunner(Display display,Form slide1,Form slide2,Form slide3) {

    this.display = display;

    this.slide1=slide1;

    this.slide2=slide2;

    this.slide3=slide3;

}

public void run() {

    while(true)

    {

        c++;

        if(c==1)

            display.setCurrent(slide1);

        else if(c==2)

            display.setCurrent(slide2);

        else if(c==3)

            display.setCurrent(slide3);

        else if(c==4)

            c=0;
```

MOBILE APPLICATION DEVELOPMENT NOTES

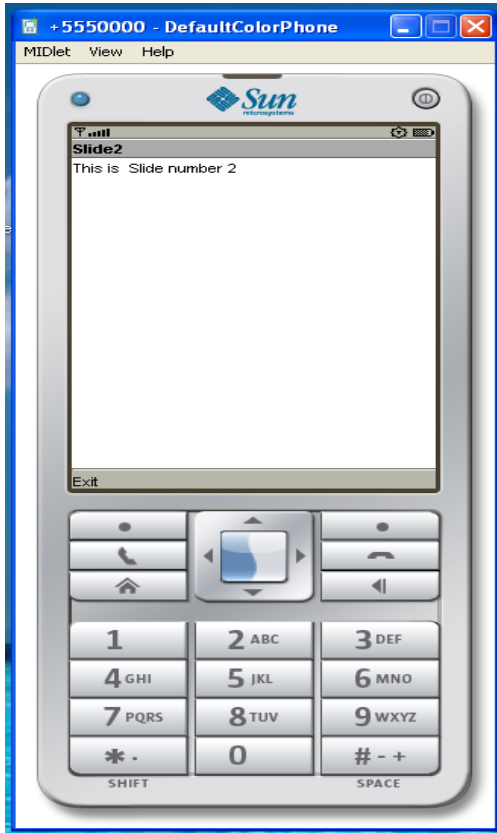
```
try
{
    Thread.sleep(1500);
}
catch(Exception ex)
{
}
}
}
```

Step 5: click build

Step 6: click run

Output:-

MOBILE APPLICATION DEVELOPMENT NOTES



MOBILE APPLICATION DEVELOPMENT NOTES

Program 11:-

Create a slideshow which has three slides, which includes pictures at PNG format. Program should change to the new slide after 5 seconds.

Step 1,2,3 same as above program

Step 4: write down the following code in the source folder

Note:-Put all images in res Folder under ur Main Project.

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

public class imageSlideShow extends MIDlet implements CommandListener {

    public Form slide1;

    public Form slide2;

    public Form slide3;

    public Command Exit;

    public Display display;

    public Image image1;

    public Image image2;

    public Image image3;

    public ImageItem imageitem1;

    public ImageItem imageitem2;

    public ImageItem imageitem3;

    public imageSlideShow()
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
{  
display=Display.getDisplay(this);  
  
    try  
    {  
        image1=Image.createImage("/1.png");  
        image2=Image.createImage("/2.png");  
        image3=Image.createImage("/3.png");  
  
        imageitem1=new  
ImageItem(null,image1,ImageItem.LAYOUT_CENTER,"image1");  
  
        imageitem2=new  
ImageItem(null,image2,ImageItem.LAYOUT_CENTER,"image2");  
  
        imageitem3=new  
ImageItem(null,image3,ImageItem.LAYOUT_CENTER,"image3");  
    }  
  
    catch(Exception ex)  
    {  
  
    }  
  
  
Exit=new Command("Exit",Command.EXIT,1);  
  
slide1=new Form("Slide1");  
  
slide1.append(imageitem1);
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
slide1.addCommand(Exit);

slide2=new Form("Slide2");

slide2.append(imageitem2);

slide2.addCommand(Exit);

slide3=new Form("Slide3");

slide3.append(imageitem3);

slide3.addCommand(Exit);

slide1.setCommandListener(this);

slide2.setCommandListener(this);

slide3.setCommandListener(this);

}

public void startApp() {

    Thread runner = new Thread(new ThreadRunner(display,slide1,slide2,slide3));

    runner.start();

}

public void pauseApp() {

}

public void destroyApp(boolean unconditional) {

}

public void commandAction(Command command,Displayable displayable)
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
{  
    if(displayable==slide1)  
    {  
        if(command==Exit)  
            notifyDestroyed();  
    }  
else if(displayable==slide2)  
    {  
        if(command==Exit)  
            notifyDestroyed();  
    }  
else if(displayable==slide3)  
    {  
        if(command==Exit)  
            notifyDestroyed();  
    }  
}  
  
class ThreadRunner implements Runnable {  
    Display display;  
    public int c=0;
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
public Form slide1;

public Form slide2;

public Form slide3;

public ThreadRunner(Display display,Form slide1,Form slide2,Form slide3) {

    this.display = display;

    this.slide1=slide1;

    this.slide2=slide2;

    this.slide3=slide3;

}

public void run() {

    while(true)

    {

        c++;

        if(c==1)

            display.setCurrent(slide1);

        else if(c==2)

            display.setCurrent(slide2);

        else if(c==3)

            display.setCurrent(slide3);

        else if(c==4)

            c=0;
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
try
{
    Thread.sleep(1500);
}
catch(Exception ex)
{

}
}
}
```

Step 5: click build

Step 6: click run

Output:-

MOBILE APPLICATION DEVELOPMENT NOTES



Program 12: program for creating datagram

Step 1,2,3 same as above program

Step 4: creating for datagram server code placed in source code folder

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

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

```
import javax.microedition.io.*;
```

```
/**
```

```
 * @author ADMIN
```

```
 */
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
public class DatagramServer extends MIDlet implements CommandListener{

    public Form form1;

    public Form form2;

    public Command startCommand;

    public Command refreshCommand;

    public Command exitCommand;

    public Display display;

    public StringItem st;

    public DatagramServer()
    {

        display=Display.getDisplay(this);

        startCommand=new Command("Start",Command.OK,1);

        refreshCommand=new Command("Refresh",Command.OK,1);

        exitCommand=new Command("Exit",Command.EXIT,1);

        st=new StringItem(" ", " ");

        form1 =new Form("DataGramserver");

        form2=new Form("Ready to receive Messages");

        form1.addCommand(startCommand);

        form1.setCommandListener(this);

        form2.addCommand(refreshCommand);

        form2.addCommand(exitCommand);

        form2.setCommandListener(this);
```


MOBILE APPLICATION DEVELOPMENT NOTES

```
}

public void startApp() {
    display.setCurrent(form1);
}

public void pauseApp() {
}

public void destroyApp(boolean unconditional) {
}

public void commandAction(Command cmd, Displayable displayable)
{
    if(displayable==form1)
    {
        if(cmd==startCommand)
        {
            try {
DatagramConnection dgc = (DatagramConnection)
Connector.open("datagram://:9001");
try {
int size = 100;
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
Datagram datagram = dgc.newDatagram(size);  
dgc.receive(datagram);  
  
form2.append(datagram.getData().toString());  
} finally {  
    dgc.close();  
}  
} catch (Exception x){  
    x.printStackTrace();  
}  
  
    display.setCurrent(form2);  
}  
}  
else if(displayable==form2)  
{  
    if(cmd==exitCommand)  
    {  
        notifyDestroyed();  
    }  
else if(cmd==refreshCommand)  
{  
    st.setText(" ");
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
}  
  
    }  
  
    }  
  
}
```

Step 5: following code for datagram client placed source folder

```
/*  
  
 * To change this template, choose Tools | Templates  
 * and open the template in the editor.  
 */  
  
import javax.microedition.midlet.*;  
import javax.microedition.lcdui.*;  
import javax.microedition.io.*;  
/**  
 * @author ADMIN  
 */  
public class DatagramClient extends MIDlet implements CommandListener{  
    public Form form1;  
    public Display display;  
    public TextField textfield;  
    public Command sendCommand;  
    public DatagramClient()
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
{  
    display=Display.getDisplay(this);  
    form1=new Form("Datagram Client");  
    sendCommand=new Command("send",Command.OK,1);  
    textfield=new TextField("Enter Text",null,30,TextField.ANY);  
    form1.append(textfield);  
    form1.addCommand(sendCommand);  
    form1.setCommandListener(this);  
}  
  
public void startApp() {  
    display.setCurrent(form1);  
}  
  
public void pauseApp() {  
}  
  
public void destroyApp(boolean unconditional) {  
}  
  
public void commandAction(Command cmd,Displayable d)  
{  
    if(cmd==sendCommand)  
    {
```

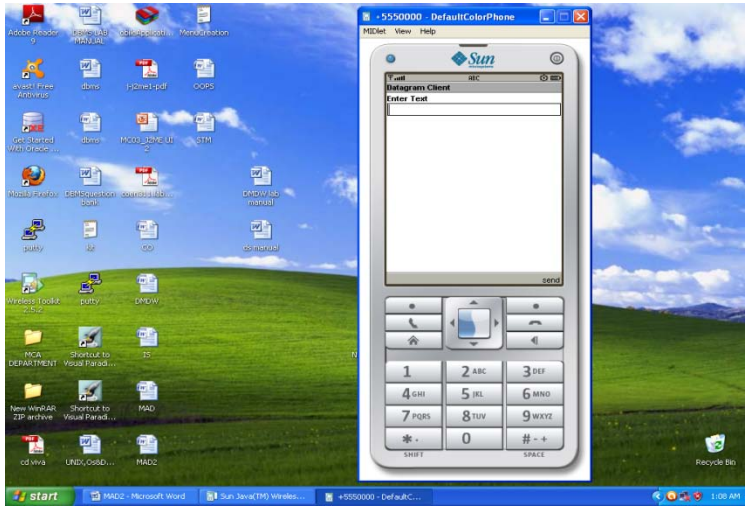
MOBILE APPLICATION DEVELOPMENT NOTES

```
        try    {  
  
        DatagramConnection dgc = (DatagramConnection)  
  
        Connector.open("datagram://localhost:9001");  
  
        try {  
while(true)  
{  
  
        byte[] payload = textfield.getString().getBytes();  
  
        Datagram datagram = dgc.newDatagram(payload, payload.length);  
  
        dgc.send(datagram);  
  
        }  
  
        } finally {  
  
        dgc.close();  
  
        }  
    } catch (Exception x) {  
  
        x.printStackTrace();  
  
    } } }  
  
}
```

Step 6: click build

Step 7: click run

MOBILE APPLICATION DEVELOPMENT NOTES



MOBILE APPLICATION DEVELOPMENT NOTES

Program 13: **program for creating login file**

Step 1,2 3 same as above program

Step 4: write the following code in source folder

```
<%--
```

```
Document : index
```

```
Created on : Aug 17, 2011, 1:35:46 PM
```

```
Author : MCA
```

```
--%>
```

```
<%@page import="java.sql.*" %>
```

```
<%
```

```
String Htno=request.getParameter("t1");
```

```
String Sname=request.getParameter("t2");
```

```
String sql="select * from stu where sno='"+Htno+"' and sname='"+Sname+'";
```

```
try
```

```
{
```

```
Class.forName("com.mysql.jdbc.Driver");
```

```
Connection
```

```
cn=DriverManager.getConnection("jdbc:mysql://localhost:3306/std","root","anurag");
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
Statement st=cn.createStatement();

ResultSet rs=st.executeQuery(sql);

if(rs.next())

{

    out.println("ok");

    out.close();

}

else

{

    out.println("0");

out.close();

}

}

catch(Exception ex)

{

    out.println(ex.toString());

    out.close();

}
```


MOBILE APPLICATION DEVELOPMENT NOTES

```
out.close();
```

```
%>
```

```
<html>
```

```
  <head>
```

```
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

```
    <title>JSP Page</title>
```

```
  </head>
```

```
  <body>
```

```
  </body>
```

```
</html>
```

```
/*
```

```
 * To change this template, choose Tools | Templates
```

```
 * and open the template in the editor.
```

```
*/
```

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

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

```
import javax.microedition.io.*;
```

```
import java.io.*;
```

```
/**
```

MOBILE APPLICATION DEVELOPMENT NOTES

* @author MCA

*/

```
public class login extends MIDlet implements CommandListener {

    public Form form1;

    public Command okCommand;

    public Display display;

    public HttpURLConnection ht=null;

    public InputStream ist=null;

    public StringItem st;

    public TextField t1;

    public TextField t2;

    public Alert alert;

    public Form form2;

    public login()

    {

        display=Display.getDisplay(this);

        st=new StringItem(" "," Welcome");

        alert =new Alert(" ","Wrong UserName or Password",null,AlertType.INFO);


        t1=new TextField("UserName"," ",30,TextField.ANY);

        t2=new TextField("Password"," ",30,TextField.PASSWORD);

        form1=new Form("Login Here");
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
form2=new Form("Welcome");

okCommand=new Command("Login",Command.OK,1);

form1.addCommand(okCommand);

form1.setCommandListener(this);


form1.append(t1);

form1.append(t2);

form2.append(st);

}


public void startApp() {

    display.setCurrent(form1);

}


public void pauseApp() {

}


public void destroyApp(boolean unconditional) {

    notifyDestroyed();

}

public void commandAction(Command cmd,Displayable d)

{
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
if(cmd==okCommand)

{

    try

    {

        //      String
url="http://192.168.5.19:8080/WebApplication7/index.jsp?t1=101&t2=aaa";

        String
url="http://192.168.5.19:8080/WebApplication7/index.jsp?t1="+t1.getString().trim()+"
&t2="+t2.getString().trim();

//ht=(HttpConnection)Connector.open("http://192.168.5.19:8080/WebApplication7/ind
ex.jsp");

        ht=(HttpConnection)Connector.open(url);

        ist=ht.openInputStream();

        byte[] b=new byte[900];

        ist.read(b);

        String s=new String(b);

        s=s.trim();

        if(s.equals("ok"))

            display.setCurrent(form2);

        else

        {

            alert.setTimeout(Alert.FOREVER);
```

MOBILE APPLICATION DEVELOPMENT NOTES

```
        display.setCurrent(alert);  
    } }  
  
    catch(Exception ex)  
    {  
form1.append(ex.toString());  
    }  
    }  
    }  
}
```

Step 5: click build

Step 6: click run