**LAB : 4**

**OBJECTIVE :**

Create an

1. WAP to Access URL (a file from url).

2. WAP to Create a List with Images.

**Requrements :**

(a) Windows PC (Windows 7/8/10) / Mac

(b) JDK 1.5

(c) Java Wireless Toolkit 2.5.2

**Implementation :**

1. WAP to Access URL

**UrlAccess.java :**

import java.io.\*;

import javax.microedition.io.\*;

import javax.microedition.lcdui.\*;

import javax.microedition.midlet.\*;

public class UrlAccess extends MIDlet {

private Display display;

String url = "http://unicode-ind.github.io/arvin/Subjects/IT/Sem-2/computer%20prog.html";

public UrlAccess() {

display = Display.getDisplay(this);

}

public void startApp() {

try {

connection(url);

} catch (IOException e) {

System.out.println("IOException " + e);

e.printStackTrace();

}

}

public void pauseApp() {}

public void destroyApp(boolean unconditional) {}

void connection(String url) throws IOException {

StreamConnection sc = null;

InputStream is = null;

StringBuffer buffer = new StringBuffer();

TextBox access;

try {

sc = (StreamConnection)Connector.open(url);

is = sc.openInputStream();

int chars;

while ((chars = is.read()) != -1) {

buffer.append((char) chars);

}

System.out.println(buffer.toString());

access = new TextBox("Access Text", buffer.toString(), 1024, 0);

} finally {

if (is != null) {

is.close();

}

if (sc != null) {

sc.close();

}

}

display.setCurrent(access);

}

}

**Output :**

|  |  |
| --- | --- |
|  |  |

1. WAP to Create a List with Images.

**ListWithImages.java :**

import javax.microedition.lcdui.\*;

import javax.microedition.midlet.\*;

public class ListWithImages extends MIDlet implements CommandListener {

private Display display;

private Form form;

private TextField txtname, txtpassword;

private Command Cmdok, Cmdcancel, ba, select;

private List lst;

public ListWithImages() {

display = Display.getDisplay(this);

}

public void startApp() {

form = new Form("Login");

txtname = new TextField("User id:", "", 250, TextField.ANY);

txtpassword = new TextField("Password:", "", 250, TextField.PASSWORD);

Cmdok = new Command("Ok", Command.OK, 0);

Cmdcancel = new Command("Cancel", Command.CANCEL, 0);

form.append(txtname);

form.append(txtpassword);

form.addCommand(Cmdok);

form.addCommand(Cmdcancel);

form.setCommandListener(this);

display.setCurrent(form);

}

public void pauseApp() {}

public void destroyApp(boolean unconditional) {}

public void commandAction(Command c, Displayable s) {

if (c == Cmdcancel) notifyDestroyed();

//else if (c == Cmdok) showlist();

else if (c == Cmdok) check();

else if (c == ba) display.setCurrent(form);

else if (c == select) showselecteditem();

}

public void check() {

if (txtname.getString().equals("18124004") && txtpassword.getString().equals("123")) {

showlist();

} else {

Alert al = new Alert("Incorrect", "Invalid login!", null, AlertType.ERROR);

display.setCurrent(al);

}

}

public void showlist() {

ba = new Command("Back", Command.BACK, 0);

select = new Command("Select", Command.OK, 0);

lst = new List("Select the item:", List.EXCLUSIVE);

Image m = null;

try {m = Image.createImage("/p2.jpg");} catch (Exception e) {}

lst.append("Item 1", m);

lst.append("Item 2", m);

lst.append("Item 3", m);

lst.append("Item 4", m);

lst.append("Item 5", m);

lst.addCommand(ba);

lst.addCommand(select);

lst.setCommandListener(this);

display.setCurrent(lst);

}

public void showselecteditem() {

String selectedvalue = lst.getString(lst.getSelectedIndex());

System.out.println(selectedvalue);

}

}

**Output :**

|  |  |
| --- | --- |
|  |  |