

I Make \$260 Every Day!

[See How...](#)
[Work From Home & Earn \\$7,397 A Month...](#)

[Home](#) | [Java](#) | [Frameworks](#) | [Database](#) | [Technology](#) | [Web Development](#) | [Build/Test Tools](#) | [Servers](#) | [PHP](#)
[Home](#) ► [J2me](#) ► J2ME RMS Sorting Example

 Questions: [Ask](#) | [Latest](#)

 J2ME RMS
Sorting
Example

J2ME RMS Sorting Example

[Share on Google+](#)

Advertisement

This example simply shows how to use of RMS package.

J2ME RMS Sorting Example

[<-Prior](#) | [Index](#) | [Next->](#)

This example simply shows how to use of RMS package. In this example we are going to sort the specified string by the implementing of RecordComparator interface as given:

```
class Comparator implements RecordComparator{
    public int compare(byte[] rec1, byte[] rec2)
    {
        String str1 = new String(rec1);
        String str2 = new String(rec2);

        int result = str1.compareTo(str2);
        if (result == 0){
            return RecordComparator.EQUIVALENT;
        } else if (result < 0){
            return RecordComparator.PRECEDES;
        } else {
            return RecordComparator.FOLLOWS;
        }
    }
}
```

In the javax.microedition.rms package only one method is as follows:

compare(byte[] rec1, byte[] rec2):- This is the integer type method and it returns RecordComparator.PRECEDES, RecordComparator.FOLLOWS, RecordComparator.EQUIVALENT. The PRECEDES apply if rec1 precedes rec2 in sort order, the FOLLOWS apply if rec1 follows rec2 in sort order and the EQUIVALENT apply if rec1 and rec2 are equivalent in terms of sort order.

Google™ Custom Search

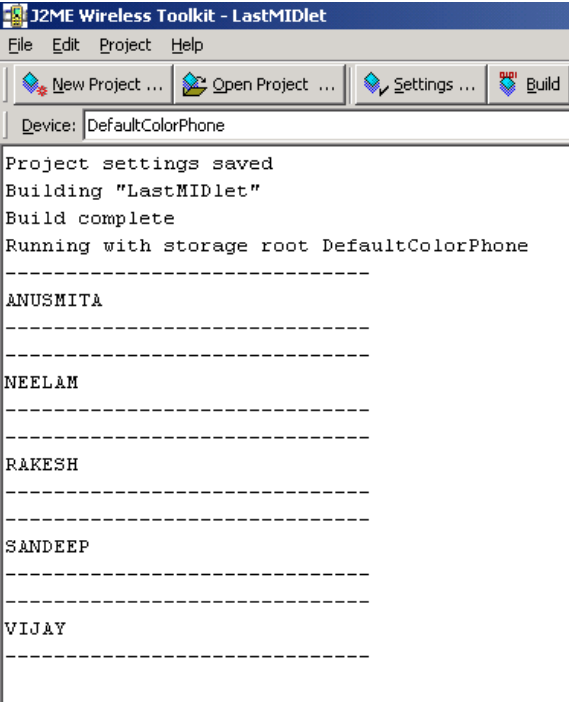
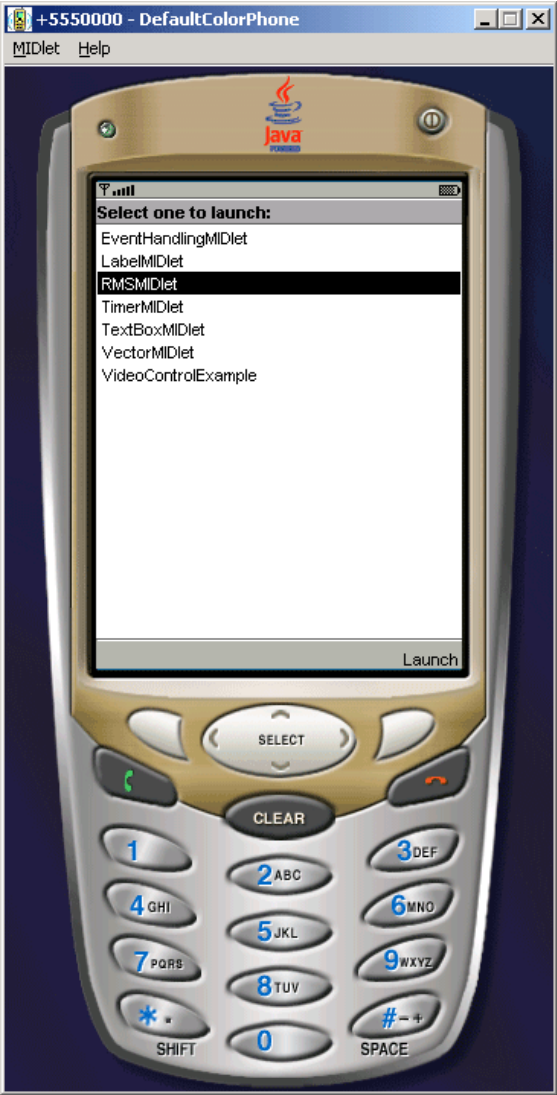
[Search](#)
[Related](#)
[Latest](#)
[Frameworks](#)
[Category](#)
[J2ME Tutorial](#)
[Access URL](#)
[A MIDlet Example to execute Alert Message with an Image](#)
[Appending Image into the J2ME Form](#)
[Appending string in J2ME Form](#)
[Arc MIDlet Example](#)
[Audio MIDlet Example](#)
[Text Example in J2ME](#)
[J2ME Calendar Application](#)
[J2ME Draw Triangle, Rectangle, Arc, Line Round Rectangle Example](#)
[>>>More Tutorials in this section...](#)

I Make \$260 Every Day!

work from home and make \$7,397 a Month

[See How...](#)

The Application is as follows:



RMSMIDlet.java

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



```

import javax.microedition.rms.*;
import javax.microedition.midlet.*;

public class RMSMIDlet extends MIDlet{
    private RecordStore record;
    static final String REC_STORE = "SORT";

    public void startApp(){
        openRecord();
        writeRecord("RAKESH");
        writeRecord("SANDEEP");
        writeRecord("NEELAM");
        writeRecord("ANUSMITA");
        writeRecord("VIJAY");
        readRecord();
        closeRecord();
        deleteRecord();
    }

    public void pauseApp(){}

    public void destroyApp(boolean unconditional){
        notifyDestroyed();
    }

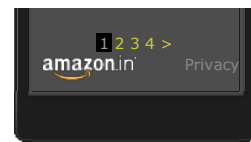
    public void openRecord(){
        try{
            record = RecordStore.openRecordStore(REC_STORE, true );
        }catch (Exception e){
            db(e.toString());
        }
    }

    public void closeRecord(){
        try{
            record.closeRecordStore();
        }catch (Exception e){
            db(e.toString());
        }
    }

    public void deleteRecord(){
        if (RecordStore.listRecordStores() != null){
            try{
                RecordStore.deleteRecordStore(REC_STORE);
            }catch (Exception e){
                db(e.toString());
            }
        }
    }

    public void writeRecord(String str){
        byte[] rec = str.getBytes();
        try{
            record.addRecord(rec, 0, rec.length);
        }catch (Exception e){
            db(e.toString());
        }
    }
}

```



```

public void readRecord(){
try{
if (record.getNumRecords() > 0){
Comparator comp = new Comparator();
RecordEnumeration re = record.enumerateRecords(null, comp, false);

while (re.hasNextElement()){
String str = new String(re.nextRecord());
System.out.println("-----");
System.out.println(str);
System.out.println("-----");
}
}
} catch (Exception e){
db(e.toString());
}
}

private void db(String error){
System.err.println("Exception: " + error);
}
}

class Comparator implements RecordComparator{
public int compare(byte[] rec1, byte[] rec2){
String str1 = new String(rec1);
String str2 = new String(rec2);
int result = str1.compareTo(str2);
if (result == 0){
return RecordComparator.EQUIVALENT;
} else if (result < 0){
return RecordComparator.PRECEDES;
} else {
return RecordComparator.FOLLOWS;
}
}
}
}

```

[Download Source Code](#)


Advertisement



[«Previous](#)

[Index](#)

[Next»](#)

Follow us on [Twitter](#), or add us on [Facebook](#) or [Google Plus](#) to keep you updated with the recent trends of Java and other open source platforms. [Connect Me on Google+](#)

We're on  [Follow](#)

 +1  +735

Recommend the
tutorial

 +1  0

ACCESS DENIED

The
requested

resource is

not available

due to a

temporary

problem

with the

server

configuration

or the

network

connectivity

issues.

We

apologize

for the

inconvenience

and

will

work

on it

immediately.

Thank

you

for

your

patience.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as

soon

as

possible.

We

will

keep

you

updated

as