PIM API on Java ME

Introduction

Many devices, especially phones, have the ability to maintain lists of phone numbers and names. Some devices also store addresses, e-mails, events, to-do lists, and other personal information. This PIM data is stored in PIM databases. Most devices will have built-in applications that manage the PIM data. Until recently, there was no direct way of accessing this information from your MIDP application. A device vendor may now expose access to its PIM databases through the PIM Optional Package, specified in JSR 75.

Determining If PIM Optional Package Is Available

It is needed to check weathere this package is available or not. We can do it with following code.

```
String currentVersion = System.getProperty("microedition.pim.version " )
```

You should also use the returned version to make sure your code is version compatible. At this time, however, only version “1.0” is available. The call will return null if the optional package is not available.

Obtaining the Singleton PIM Instance

There is PIM class which we need to use. It is a abstract class so we cannot create object of it so we need to use following static method to get the instant of PIM.

```
public static PIM getInstance()
```

Procedure for Opening the PIM Lists

We can access the PIM lists once we have the singleton instance. A PIM list can be obtained using one of these methods.

```
public PIMList openPIMList(char pimListType, int mode) throws PIMException

public PIMList openPIMList(int pimListType, int mode, String name) throws PIMException
```

possible value for pimListType : PIM.CONTACT_LIST, PIM.EVENT_LIST, or PIM.TODO_LIST.

possible value for mode :PIM.READ_ONLY, PIM.READ_WRITE, or PIM.WRITE_ONLY

Functions Used Reading Field Value

The following functions are used to read filed values.

```
public byte [] getBinary(char field, int index)

public long getDate(char field, int index)

public int getInt(char field, int index)

public String getString(char field, int index)

public Boolean getBoolean(char field, int index)

public String [] getStringArray(char field, int index)
```


(C) Copyright Nokia 2014. All rights reserved.

Functions Used Modifying Field Values

```
public void setBinary(int field, int index, int attributes, byte[] value,int offset, int length)

public void setDate(char field, int index, string attributes, boolean value)

public void setInt(char field, int index, string attributes, boolean value)

public void setString(char field, int index, string attributes, boolean value)

public void setBoolean(char field, int index, string attributes, boolean value)

public void setStringArray(char field, int index, string attributes, boolean value)
```


(C) Copyright Nokia 2014. All rights reserved.