Category:Java ME APIs

Java ME API

Java ME has the following APIs
• MIDP 2.0 API
• Wireless Messaging API (JSR-120)
• Wireless Messaging API 2.0 (JSR-205)
• Mobile Media API (JSR-135)
• Advanced Multimedia Supplements (JSR-234
• Bluetooth API (JSR-82)
• Mobile 3D Graphics API (JSR-184)
• Scalable 2D Vector Graphics API (JSR-226)
• Location API (JSR-179)
• FileConnection API (JSR-75, Optional package)
• PIM API (JSR-75, Optional package)
• SIP API (JSR180)
• Security and Trust Services API (JSR-177)
• Web Services API Parser Package (JSR-172)
• Web Services API RPC Package (JSR-172)
• Mobile Internalization API (JSR-238)

'

MIDP 2.0 API

User Interface Package
javax.microedition.lcdui
The UI API provides a set of features for implementation of user interfaces for MIDP applications.
javax.microedition.lcdui.game
The Game API package provides a series of classes that enable the development of rich gaming content for wireless devices.

Persistence Package
javax.microedition.rms
The Mobile Information Device Profile provides a mechanism for MIDlets to persistently store data and later retrieve it.

Application Lifecycle Package
javax.microedition.midlet
The MIDlet package defines Mobile Information Device Profile applications and the interactions between the application and the environment in which the application runs.

Networking Package
javax.microedition.io
This package contains Classes for the Generic Connection framework.

Audio Package
javax.microedition.media
The MIDP 2.0 Media API contains classes for creating player

javax.microedition.media.control
This package defines the specific Control types that can be used with a Player.

Public Key Package
javax.microedition.pki
Certificates are used to authenticate information for secure Connections.

Core Packages
java.io
Provides classes for input and output through data streams.
java.lang
Provides classes that are fundamental to the Java programming language.
java.util
Contains the collection classes, and the date and time facilities.

(C) Copyright Nokia 2013. All rights reserved.

Wireless Messaging API (JSR 120)
Messaging Interfaces
javax.wireless.messaging
This package defines an API which allows applications to send and receive wireless messages.

Networking Package
javax.microedition.io
This package includes the platform networking interfaces which have been modified for use on platforms that support message connections.

Mobile Media API (JSR-135)

javax.microedition.media
Interface Summary
Control : A Control object is used to control some media processing functions.
Controllable : Controllable provides an interface for obtaining the Controls from an object like a Player.
Player : Player controls the rendering of time based media data.
PlayerListener : PlayerListener is the interface for receiving asynchronous events generated by Players.
TimeBase : A TimeBase is a constantly ticking source of time.

Class Summary
Manager : Manager is the access point for obtaining system dependent resources such as Players for multimedia processing.

javax.microedition.media.control
This package defines the specific Control types that can be used with a Player.
Interface Summary
FramePositioningControl
The FramePositioningControl is the interface to control precise positioning to a video frame for Players.
GUIControl
GUIControl extends Control and is defined for controls that provide GUI functionalities.
MetaDataControl
MetaDataControl is used to retrieve metadata information included within the media streams.
MIDIControl
MIDIControl provides access to MIDI rendering and transmitting devices.
PitchControl
PitchControl raises or lowers the playback pitch of audio without changing the playback speed.
RateControl
RateControl controls the playback rate of a Player.
RecordControl
RecordControl controls the recording of media from a Player.
StopTimeControl
StopTimeControl allows one to specify a preset stop time for a Player.
TempoControl
TempoControl controls the tempo, in musical terms, of a song.
ToneControl
ToneControl is the interface to enable playback of a user-defined monotonic tone sequence.
VideoControl
VideoControl controls the display of video.
VolumeControl
VolumeControl is an interface for manipulating the audio volume of a Player.

javax.microedition.media.protocol
This package defines the custom protocol handling interfaces.
Interface Summary
SourceStream
Abstracts a single stream of media data.

Class Summary
ContentDescriptor
A ContentDescriptor identifies media data containers.
DataSource
A DataSource is an abstraction for media protocol-handlers.

Advanced Multimedia Supplements API (JSR-234).
The JSR 234 has the following packages
Packages
javax.microedition.amms
This package introduces four new concepts: GlobalManager, Spectator, Module and MediaProcessor.
javax.microedition.amms.control

(C) Copyright Nokia 2013. All rights reserved.

This package contains Controls for setting formats, for defining the effect processing network structure and for setting the audio panning.
javax.microedition.amms.control.audio3d
This package contains Controls for various 3D audio settings.
javax.microedition.amms.control.audioeffect
This package contains EffectControls for audio and related assisting Controls.
javax.microedition.amms.control.camera
This package contains Controls for various camera settings.
javax.microedition.amms.control.imageeffect
This package contains EffectControls for video and for still images.
javax.microedition.amms.control.tuner
This package contains Controls for various tuner settings.

Bluetooth API (JSR 82)
''''*javax.bluetooth : This package defines classes for Bluetooth communication*
javax.obex : -Defines Object exchange package

' ''''Mobile 3D Graphics API
This specification defines the Mobile 3D Graphics API (M3G) for J2ME.

Packages
javax.microedition.m3g
Defines an API for rendering three-dimensional (3D) graphics at interactive frame rates, including a scene graph structure and a corresponding file format for efficient management and deployment of 3D content.

Class Summary
AnimationController
Controls the position, speed and weight of an animation sequence.
AnimationTrack
Associates a KeyframeSequence with an AnimationController and an animatable property.
Appearance
A set of component objects that define the rendering attributes of a Mesh or Sprite3D.
Background
Defines whether and how to clear the viewport.
Camera
A scene graph node that defines the position of the viewer in the scene and the projection from 3D to 2D.
CompositingMode
An Appearance component encapsulating per-pixel compositing attributes.
Fog
An Appearance component encapsulating attributes for fogging.
Graphics3D
A singleton 3D graphics context that can be bound to a rendering target.
Group
A scene graph node that stores an unordered set of nodes as its children.
Image2D
A two-dimensional image that can be used as a texture, background or sprite image.
IndexBuffer
An abstract class defining how to connect vertices to form a geometric object.
KeyframeSequence
Encapsulates animation data as a sequence of time-stamped, vector-valued keyframes.
Light
A scene graph node that represents different kinds of light sources.
Loader
Downloads and deserializes scene graph nodes and node components, as well as entire scene graphs.
Material
An Appearance component encapsulating material attributes for lighting computations.
Mesh
A scene graph node that represents a 3D object defined as a polygonal surface.
MorphingMesh
A scene graph node that represents a vertex morphing polygon mesh.
Node
An abstract base class for all scene graph nodes.
Object3D
An abstract base class for all objects that can be part of a 3D world.
PolygonMode
An Appearance component encapsulating polygon-level attributes.
RayIntersection
A RayIntersection object is filled in by the pick methods in Group.
SkinnedMesh
A scene graph node that represents a skeletally animated polygon mesh.
Sprite3D
A scene graph node that represents a 2-dimensional image with a 3D position.

(C) Copyright Nokia 2013. All rights reserved.

Texture2D
An Appearance component encapsulating a two-dimensional texture image and a set of attributes specifying how the image is to be applied on submeshes.
Transform
A generic 4x4 floating point matrix, representing a transformation.
Transformable
An abstract base class for Node and Texture2D, defining common methods for manipulating node and texture transformations.
TriangleStripArray
TriangleStripArray defines an array of triangle strips.
VertexArray
An array of integer vectors representing vertex positions, normals, colors, or texture coordinates.
VertexBuffer
VertexBuffer holds references to VertexArrays that contain the positions, colors, normals, and texture coordinates for a set of vertices.
World
A special Group node that is a top-level container for scene graphs.

Scalable 2D Vector Graphics API for J2ME (JSR-226)
Packages
javax.microedition.m2g
This package contains the core rendering part of this proposal.
org.w3c.dom
This package is subsetted DOM Level 3 Core APIs.
org.w3c.dom.events
This package contains the necessary event handling interfaces that are a subset of DOM Events Level 2 APIs.
org.w3c.dom.svg
The SVG Tiny 1.1 DOM API defined in this specification is subset of W3C SVG Tiny 1.2 uDOM.

JSR179 Location API for J2ME
Packages
javax.microedition.location
The javax.microedition.location package contains the basic classes needed to request and get a location result.

Interface Summary
LocationListener
The LocationListener represents a listener that receives events associated with a particular LocationProvider.
ProximityListener
This interface represents a listener to events associated with detecting proximity to some registered coordinates.

Class Summary
AddressInfo
The AddressInfo class holds textual address information about a location.
Coordinates
The Coordinates class represents coordinates as latitude-longitude-altitude values.
Criteria
The criteria used for the selection of the location provider is defined by the values in this class.
Landmark
The Landmark class represents a landmark, i.e. a known location with a name.
LandmarkStore
The LandmarkStore class provides methods to store, delete and retrieve landmarks from a persistent landmark store.
Location
The Location class represents the standard set of basic location information.
LocationProvider
This is the starting point for applications using this API and represents a source of the location information.
Orientation
The Orientation class represents the physical orientation of the terminal.
QualifiedCoordinates
The QualifiedCoordinates class represents coordinates as latitude-longitude-altitude values that are associated with an accuracy value.

Exception Summary
LandmarkException
The LandmarkException is thrown when an error related to handling landmarks has occurred.
LocationException
The LocationException is thrown when a location API specific error has occurred.

FileConnection Optional Package (JSR 75)
Packages
javax.microedition.io.file
This package describes file system access support based on the Generic Connection Framework.

javax.microedition.io.file
This package describes file system access support based on the Generic Connection Framework.

(C) Copyright Nokia 2013. All rights reserved.

Interface Summary
FileConnection
This interface is intended to access files or directories that are located on removeable media and/or file systems on a device.
FileSystemListener
This class is used for receiving status notification when adding or removing a file system root.

Class Summary
FileSystemRegistry
The FileSystemRegistry is a central registry for file system listeners interested in the adding and removing (or mounting and unmounting) of file systems on a device.

PIM Optional Package (JSR 75)

Packages
javax.microedition.pim
The PIM package defines APIs to access Personal Information Management (PIM) data.

Interface Summary
Contact
Represents a single Contact entry in a PIM Contact database.
ContactList
Represents a Contact list containing Contact items.
Event
Represents a single Event entry in a PIM Event database.
EventList
Represents an Event list containing Event items.
PIMItem
Represents the common interfaces of an item for a PIM list.
PIMList
Represents the common functionality of a PIM list.
ToDo
Represents a single To Do item in a PIM To Do database.
ToDoList
Represents a ToDo list containing ToDo items.

Class Summary
PIM
Class for accessing PIM lists on a device and performing PIM wide functions.
RepeatRule
Represents a description for a repeating pattern for an Event item.

Exception Summary
FieldEmptyException
Represents an exception thrown when an attempt is made to access a field that does not have any data values associated with it.
FieldFullException
Represents an exception thrown when an attempt is made to add data to a field but the field already has all available slots for data values assigned.
PIMException
Represents exceptions thrown by the PIM classes.
UnsupportedFieldException
Represents an exception thrown when a field is referenced that is not supported in the particular PIM list that an element belongs to.

JSR180 SIP API for J2ME Version 1.0
Packages
javax.microedition.sip
This specification defines the JSR180, SIP API for J2ME.

Interface Summary
SipClientConnection
SipClientConnection represents SIP client transaction.
SipClientConnectionListener
Listener interface for incoming SIP responses.
SipConnection
SipConnection is the base interface for SIP connections.
SipConnectionNotifier
This interface defines a SIP server connection notifier.
SipDialog
SipDialog represents one SIP Dialog.
SipRefreshListener
Listener interface for RefreshHelper events.
SipServerConnection
SipServerConnection represents SIP server transaction.
SipServerConnectionListener
Listener interface for incoming SIP requests.

Class Summary
SipAddress
SipAddress provides a generic SIP address parser.

(C) Copyright Nokia 2013. All rights reserved.

SipHeader
SipHeader provides generic SIP header parser helper.
SipRefreshHelper
This class implements the functionality that facilitates the handling of refreshing requests on behalf of the application.

Exception Summary
SipException
This is an exception class for SIP specific errors.

Security and Trust Services API (SATSA)
The Security and Trust Services API for J2ME defines four optional packages to support smart card communication, generation of digital signatures, and low-level cryptography operations.
Exception classes from java.lang required by the SATSA-APDU, SATSA-JCRMI, and SATSA-CRYPTO optional
java.lang
Provides classes that are fundamental to the Java programming language.

Optional Package: SATSA-APDU Communication
javax.microedition.apdu
This package defines the APDU protocol handler for ISO7816-4 communication to a smart card device.

Optional Package: SATSA-JCRMI Communication
java.rmi
A subset of the java.rmi package in the Java 2 Standard Edition.
javacard.framework
Provides Java Card API exceptions that may be thrown by a remote method.
javacard.framework.service
Provides a Java Card API exception that may be thrown by a remote method.
javacard.security
Provides a Java Card API exception that may be thrown by a remote method.
javax.microedition.jcrmi
Provides classes and interfaces for Java Card RMI connection.

Optional Package: SATSA-PKI Signature Service
javax.microedition.pki
This package defines classes to support basic user certificate management.
javax.microedition.securityservice
This package defines classes to generate application-level digital signatures that conform to the Cryptographic Message Syntax (CMS) format.

Optional Package: SATSA-CRYPTO Cryptographic Interfaces
java.security
Provides the classes and interfaces for the security framework.
java.security.spec
Provides classes and interfaces for key specifications and algorithm parameter specifications.
javax.crypto
Provides the classes and interfaces for cryptographic operations.
javax.crypto.spec
Provides classes and interfaces for key specifications and algorithm parameter specifications.

JavaME Web Services
J2ME XML Parser Package
javax.xml.parsers

org.xml.sax

org.xml.sax.helpers

Package javax.xml.parsers
Class Summary
SAXParser
Defines the API that represents a simple SAX parser.
SAXParserFactory
Defines a factory API that enables applications to configure and obtain a SAX based parser to parse XML documents.

Exception Summary
ParserConfigurationException
Indicates a serious configuration error.

Error Summary
FactoryConfigurationError
Thrown when a problem with configuration with the Parser Factories exists.

(C) Copyright Nokia 2013. All rights reserved.

org.xml.sax
Interface Summary
Attributes
Interface for a list of XML attributes.
Locator
Interface for associating a SAX event with a document location.

Class Summary
InputSource
A single input source for an XML entity.

Exception Summary
SAXException
Encapsulate a general SAX error or warning.
SAXNotRecognizedException
Exception class for an unrecognized identifier.
SAXNotSupportedException
Exception class for an unsupported operation.
SAXParseException
Encapsulate an XML parse error or warning.

JSR-238 Mobile Internationalization API.

Packages
javax.microedition.global
Contains classes that implement the JSR-238, "Mobile Internationalization API".

Package javax.microedition.global
Contains classes that implement the JSR-238, "Mobile Internationalization API".
Class Summary
Formatter
Creates locale-specific renditions of data items.
ResourceManager
Resource manager for applications.
StringComparator
Compares two strings using locale-specific rules.

Exception Summary
ResourceException
Thrown when a resource manager operation fails.
UnsupportedLocaleException
Thrown when an operation is attempted that is not supported for the specified locale.

(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.