| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Pack200.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/java/util/jar/Manifest.html)   [**NEXT CLASS**](http://docs.google.com/java/util/jar/Pack200.Packer.html) | [**FRAMES**](http://docs.google.com/index.html?java/util/jar/Pack200.html)    [**NO FRAMES**](http://docs.google.com/Pack200.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | FIELD | CONSTR | [METHOD](#2et92p0) | DETAIL: FIELD | CONSTR | [METHOD](#3dy6vkm) |

## **java.util.jar**

Class Pack200

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 **java.util.jar.Pack200**

public abstract class **Pack200**extends [Object](http://docs.google.com/java/lang/Object.html)

Transforms a JAR file to or from a packed stream in Pack200 format. Please refer to Network Transfer Format JSR 200 Specification at <http://jcp.org/aboutJava/communityprocess/review/jsr200/index.html>

Typically the packer engine is used by application developers to deploy or host JAR files on a website. The unpacker engine is used by deployment applications to transform the byte-stream back to JAR format.

Here is an example using packer and unpacker:

import java.util.jar.Pack200;  
 import java.util.jar.Pack200.\*;  
 ...  
 // Create the Packer object  
 Packer packer = Pack200.newPacker();  
  
 // Initialize the state by setting the desired properties  
 Map p = packer.properties();  
 // take more time choosing codings for better compression  
 p.put(Packer.EFFORT, "7"); // default is "5"  
 // use largest-possible archive segments (>10% better compression).  
 p.put(Packer.SEGMENT\_LIMIT, "-1");  
 // reorder files for better compression.  
 p.put(Packer.KEEP\_FILE\_ORDER, Packer.FALSE);  
 // smear modification times to a single value.  
 p.put(Packer.MODIFICATION\_TIME, Packer.LATEST);  
 // ignore all JAR deflation requests,  
 // transmitting a single request to use "store" mode.  
 p.put(Packer.DEFLATE\_HINT, Packer.FALSE);  
 // discard debug attributes  
 p.put(Packer.CODE\_ATTRIBUTE\_PFX+"LineNumberTable", Packer.STRIP);  
 // throw an error if an attribute is unrecognized  
 p.put(Packer.UNKNOWN\_ATTRIBUTE, Packer.ERROR);  
 // pass one class file uncompressed:  
 p.put(Packer.PASS\_FILE\_PFX+0, "mutants/Rogue.class");  
 try {  
 JarFile jarFile = new JarFile("/tmp/testref.jar");  
 FileOutputStream fos = new FileOutputStream("/tmp/test.pack");  
 // Call the packer  
 packer.pack(jarFile, fos);  
 jarFile.close();  
 fos.close();  
   
 File f = new File("/tmp/test.pack");  
 FileOutputStream fostream = new FileOutputStream("/tmp/test.jar");  
 JarOutputStream jostream = new JarOutputStream(fostream);  
 Unpacker unpacker = Pack200.newUnpacker();  
 // Call the unpacker  
 unpacker.unpack(f, jostream);  
 // Must explicitly close the output.  
 jostream.close();  
 } catch (IOException ioe) {  
 ioe.printStackTrace();  
 }

A Pack200 file compressed with gzip can be hosted on HTTP/1.1 web servers. The deployment applications can use "Accept-Encoding=pack200-gzip". This indicates to the server that the client application desires a version of the file encoded with Pack200 and further compressed with gzip. Please refer to [Java Deployment Guide](http://docs.google.com/technotes/guides/deployment/deployment-guide/pack200.html) for more details and techniques.

Unless otherwise noted, passing a null argument to a constructor or method in this class will cause a [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) to be thrown.

**Since:** 1.5

| **Nested Class Summary** | |
| --- | --- |
| static interface | [**Pack200.Packer**](http://docs.google.com/java/util/jar/Pack200.Packer.html)            The packer engine applies various transformations to the input JAR file, making the pack stream highly compressible by a compressor such as gzip or zip. |
| static interface | [**Pack200.Unpacker**](http://docs.google.com/java/util/jar/Pack200.Unpacker.html)            The unpacker engine converts the packed stream to a JAR file. |

| **Method Summary** | |
| --- | --- |
| static [Pack200.Packer](http://docs.google.com/java/util/jar/Pack200.Packer.html) | [**newPacker**](http://docs.google.com/java/util/jar/Pack200.html#newPacker())()            Obtain new instance of a class that implements Packer. |
| static [Pack200.Unpacker](http://docs.google.com/java/util/jar/Pack200.Unpacker.html) | [**newUnpacker**](http://docs.google.com/java/util/jar/Pack200.html#newUnpacker())()            Obtain new instance of a class that implements Unpacker. |

| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/java/lang/Object.html) |
| --- |
| [clone](http://docs.google.com/java/lang/Object.html#clone()), [equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [hashCode](http://docs.google.com/java/lang/Object.html#hashCode()), [notify](http://docs.google.com/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/java/lang/Object.html#notifyAll()), [toString](http://docs.google.com/java/lang/Object.html#toString()), [wait](http://docs.google.com/java/lang/Object.html#wait()), [wait](http://docs.google.com/java/lang/Object.html#wait(long)), [wait](http://docs.google.com/java/lang/Object.html#wait(long,%20int)) |

| **Method Detail** |
| --- |

### newPacker

public static [Pack200.Packer](http://docs.google.com/java/util/jar/Pack200.Packer.html) **newPacker**()

Obtain new instance of a class that implements Packer.

* If the system property java.util.jar.Pack200.Packer is defined, then the value is taken to be the fully-qualified name of a concrete implementation class, which must implement Packer. This class is loaded and instantiated. If this process fails then an unspecified error is thrown.
* If an implementation has not been specified with the system property, then the system-default implementation class is instantiated, and the result is returned.

Note: The returned object is not guaranteed to operate correctly if multiple threads use it at the same time. A multi-threaded application should either allocate multiple packer engines, or else serialize use of one engine with a lock.

**Returns:**A newly allocated Packer engine.

### newUnpacker

public static [Pack200.Unpacker](http://docs.google.com/java/util/jar/Pack200.Unpacker.html) **newUnpacker**()

Obtain new instance of a class that implements Unpacker.

If the system property java.util.jar.Pack200.Unpacker is defined, then the value is taken to be the fully-qualified name of a concrete implementation class, which must implement Unpacker. The class is loaded and instantiated. If this process fails then an unspecified error is thrown.

If an implementation has not been specified with the system property, then the system-default implementation class is instantiated, and the result is returned.

Note: The returned object is not guaranteed to operate correctly if multiple threads use it at the same time. A multi-threaded application should either allocate multiple unpacker engines, or else serialize use of one engine with a lock.

**Returns:**A newly allocated Unpacker engine.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Pack200.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/java/util/jar/Manifest.html)   [**NEXT CLASS**](http://docs.google.com/java/util/jar/Pack200.Packer.html) | [**FRAMES**](http://docs.google.com/index.html?java/util/jar/Pack200.html)    [**NO FRAMES**](http://docs.google.com/Pack200.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | FIELD | CONSTR | [METHOD](#2et92p0) | DETAIL: FIELD | CONSTR | [METHOD](#3dy6vkm) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).