| | [**Overview**](http://docs.google.com/overview-summary.html) | **Package** | Class | [**Use**](http://docs.google.com/package-use.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 PACKAGE**](http://docs.google.com/javax/annotation/processing/package-summary.html)   [**NEXT PACKAGE**](http://docs.google.com/javax/crypto/interfaces/package-summary.html) | [**FRAMES**](http://docs.google.com/index.html?javax/crypto/package-summary.html)    [**NO FRAMES**](http://docs.google.com/package-summary.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

## Package javax.crypto

Provides the classes and interfaces for cryptographic operations.

**See:**

[**Description**](#3znysh7)

| **Interface Summary** | |
| --- | --- |
| [**SecretKey**](http://docs.google.com/javax/crypto/SecretKey.html) | A secret (symmetric) key. |

| **Class Summary** | |
| --- | --- |
| [**Cipher**](http://docs.google.com/javax/crypto/Cipher.html) | This class provides the functionality of a cryptographic cipher for encryption and decryption. |
| [**CipherInputStream**](http://docs.google.com/javax/crypto/CipherInputStream.html) | A CipherInputStream is composed of an InputStream and a Cipher so that read() methods return data that are read in from the underlying InputStream but have been additionally processed by the Cipher. |
| [**CipherOutputStream**](http://docs.google.com/javax/crypto/CipherOutputStream.html) | A CipherOutputStream is composed of an OutputStream and a Cipher so that write() methods first process the data before writing them out to the underlying OutputStream. |
| [**CipherSpi**](http://docs.google.com/javax/crypto/CipherSpi.html) | This class defines the *Service Provider Interface* (**SPI**) for the Cipher class. |
| [**EncryptedPrivateKeyInfo**](http://docs.google.com/javax/crypto/EncryptedPrivateKeyInfo.html) | This class implements the EncryptedPrivateKeyInfo type as defined in PKCS #8. |
| [**ExemptionMechanism**](http://docs.google.com/javax/crypto/ExemptionMechanism.html) | This class provides the functionality of an exemption mechanism, examples of which are *key recovery*, *key weakening*, and *key escrow*. |
| [**ExemptionMechanismSpi**](http://docs.google.com/javax/crypto/ExemptionMechanismSpi.html) | This class defines the *Service Provider Interface* (**SPI**) for the ExemptionMechanism class. |
| [**KeyAgreement**](http://docs.google.com/javax/crypto/KeyAgreement.html) | This class provides the functionality of a key agreement (or key exchange) protocol. |
| [**KeyAgreementSpi**](http://docs.google.com/javax/crypto/KeyAgreementSpi.html) | This class defines the *Service Provider Interface* (**SPI**) for the KeyAgreement class. |
| [**KeyGenerator**](http://docs.google.com/javax/crypto/KeyGenerator.html) | This class provides the functionality of a secret (symmetric) key generator. |
| [**KeyGeneratorSpi**](http://docs.google.com/javax/crypto/KeyGeneratorSpi.html) | This class defines the *Service Provider Interface* (**SPI**) for the KeyGenerator class. |
| [**Mac**](http://docs.google.com/javax/crypto/Mac.html) | This class provides the functionality of a "Message Authentication Code" (MAC) algorithm. |
| [**MacSpi**](http://docs.google.com/javax/crypto/MacSpi.html) | This class defines the *Service Provider Interface* (**SPI**) for the Mac class. |
| [**NullCipher**](http://docs.google.com/javax/crypto/NullCipher.html) | The NullCipher class is a class that provides an "identity cipher" -- one that does not tranform the plaintext. |
| [**SealedObject**](http://docs.google.com/javax/crypto/SealedObject.html) | This class enables a programmer to create an object and protect its confidentiality with a cryptographic algorithm. |
| [**SecretKeyFactory**](http://docs.google.com/javax/crypto/SecretKeyFactory.html) | This class represents a factory for secret keys. |
| [**SecretKeyFactorySpi**](http://docs.google.com/javax/crypto/SecretKeyFactorySpi.html) | This class defines the *Service Provider Interface* (**SPI**) for the SecretKeyFactory class. |

| **Exception Summary** | |
| --- | --- |
| [**BadPaddingException**](http://docs.google.com/javax/crypto/BadPaddingException.html) | This exception is thrown when a particular padding mechanism is expected for the input data but the data is not padded properly. |
| [**ExemptionMechanismException**](http://docs.google.com/javax/crypto/ExemptionMechanismException.html) | This is the generic ExemptionMechanism exception. |
| [**IllegalBlockSizeException**](http://docs.google.com/javax/crypto/IllegalBlockSizeException.html) | This exception is thrown when the length of data provided to a block cipher is incorrect, i.e., does not match the block size of the cipher. |
| [**NoSuchPaddingException**](http://docs.google.com/javax/crypto/NoSuchPaddingException.html) | This exception is thrown when a particular padding mechanism is requested but is not available in the environment. |
| [**ShortBufferException**](http://docs.google.com/javax/crypto/ShortBufferException.html) | This exception is thrown when an output buffer provided by the user is too short to hold the operation result. |

## Package javax.crypto Description

Provides the classes and interfaces for cryptographic operations. The cryptographic operations defined in this package include encryption, key generation and key agreement, and Message Authentication Code (MAC) generation.

Support for encryption includes symmetric, asymmetric, block, and stream ciphers. This package also supports secure streams and sealed objects.

Many of the classes provided in this package are provider-based. The class itself defines a programming interface to which applications may write. The implementations themselves may then be written by independent third-party vendors and plugged in seamlessly as needed. Therefore application developers may take advantage of any number of provider-based implementations without having to add or rewrite code.

## Related Documentation

For further documentation, please see:

* [**JavaTM Cryptography Architecture API Specification and Reference**](http://docs.google.com/technotes/guides/security/crypto/CryptoSpec.html)
* [**How to Implement a Provider for the JavaTM Cryptography Architecture**](http://docs.google.com/technotes/guides/security/crypto/HowToImplAProvider.html)

**Since:** JCE1.2

| | [**Overview**](http://docs.google.com/overview-summary.html) | **Package** | Class | [**Use**](http://docs.google.com/package-use.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 PACKAGE**](http://docs.google.com/javax/annotation/processing/package-summary.html)   [**NEXT PACKAGE**](http://docs.google.com/javax/crypto/interfaces/package-summary.html) | [**FRAMES**](http://docs.google.com/index.html?javax/crypto/package-summary.html)    [**NO FRAMES**](http://docs.google.com/package-summary.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

[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).