

**Module** `java.base`

**Package** `java.io`

## Class `IOException`

```
java.lang.Object
  java.lang.Throwable
    java.lang.Exception
      java.io.IOException
```

### All Implemented Interfaces:

`Serializable`

### Direct Known Subclasses:

`AttachOperationFailedException`, `ChangedCharSetException`, `CharacterCodingException`, `CharConversionException`, `ClosedChannelException`, `ClosedConnectionException`, `EOFException`, `FileLockInterruptedException`, `FileNotFoundException`, `FilerException`, `FileSystemException`, `HttpRetryException`, `HttpTimeoutException`, `IIIOException`, `InterruptedByTimeoutException`, `InterruptedIOException`, `InvalidPropertiesFormatException`, `JMXProviderException`, `JMXServerErrorException`, `MalformedURLException`, `ObjectStreamException`, `ProtocolException`, `RemoteException`, `SaslException`, `SocketException`, `SSLException`, `SyncFailedException`, `TransportTimeoutException`, `UnknownHostException`, `UnknownServiceException`, `UnsupportedEncodingException`, `UserPrincipalNotFoundException`, `UTFDataFormatException`, `WebSocketHandshakeException`, `ZipException`

---

```
public class IOException
```

```
extends Exception
```

Signals that an I/O exception of some sort has occurred. This class is the general class of exceptions produced by failed or interrupted I/O operations.

### Since:

1.0

### See Also:

`InputStream`, `OutputStream`, `Serialized Form`

## Constructor Summary

### Constructors

Constructor	Description
<code>IOException()</code>	Constructs an <code>IOException</code> with <code>null</code> as its error detail message.
<code>IOException(String message)</code>	Constructs an <code>IOException</code> with the specified detail message.
<code>IOException(String message, Throwable cause)</code>	Constructs an <code>IOException</code> with the specified detail message and cause.
<code>IOException(Throwable cause)</code>	Constructs an <code>IOException</code> with the specified cause and a detail message of <code>(cause==null ? null : cause.toString())</code> (which typically contains the class and detail message of cause).

## Method Summary

### Methods declared in class `java.lang.Throwable`

`addSuppressed`, `fillInStackTrace`, `getCause`, `getLocalizedMessage`, `getMessage`, `getStackTrace`, `getSuppressed`, `initCause`, `printStackTrace`, `printStackTrace`, `printStackTrace`, `setStackTrace`, `toString`

### Methods declared in class `java.lang.Object`

`clone`, `equals`, `finalize`, `getClass`, `hashCode`, `notify`, `notifyAll`, `wait`, `wait`, `wait`

## Constructor Details

### `IOException`

```
public IOException()
```

Constructs an `IOException` with `null` as its error detail message.

## IOException

```
public IOException(String message)
```

Constructs an `IOException` with the specified detail message.

**Parameters:**

`message` - The detail message (which is saved for later retrieval by the `Throwable.getMessage()` method)

## IOException

```
public IOException(String message,  
                   Throwable cause)
```

Constructs an `IOException` with the specified detail message and cause.

Note that the detail message associated with `cause` is *not* automatically incorporated into this exception's detail message.

**Parameters:**

`message` - The detail message (which is saved for later retrieval by the `Throwable.getMessage()` method)

`cause` - The cause (which is saved for later retrieval by the `Throwable.getCause()` method). (A null value is permitted, and indicates that the cause is nonexistent or unknown.)

**Since:**

1.6

## IOException

```
public IOException(Throwable cause)
```

Constructs an `IOException` with the specified cause and a detail message of `(cause==null ? null : cause.toString())` (which typically contains the class and detail message of `cause`). This constructor is useful for IO exceptions that are little more than

wrappers for other throwables.

**Parameters:**

cause - The cause (which is saved for later retrieval by the `Throwable.getCause()` method). (A null value is permitted, and indicates that the cause is nonexistent or unknown.)

**Since:**

1.6

---

[Report a bug or suggest an enhancement](#)

For further API reference and developer documentation see the [Java SE Documentation](#), which contains more detailed, developer-targeted descriptions with conceptual overviews, definitions of terms, workarounds, and working code examples. [Other versions](#).

Java is a trademark or registered trademark of Oracle and/or its affiliates in the US and other countries.

Copyright © 1993, 2024, Oracle and/or its affiliates, 500 Oracle Parkway, Redwood Shores, CA 94065 USA.

All rights reserved. Use is subject to [license terms](#) and the [documentation redistribution policy](#). Modify [Cookie Preferences](#). Modify [Ad Choices](#).