

# Class `java.io.FileOutputStream`

```
java.lang.Object
|
+----java.io.OutputStream
|
+----java.io.FileOutputStream
```

---

public class **FileOutputStream**  
extends [OutputStream](#)

File output stream, can be constructed from a file descriptor or a file name.

**See Also:**

[FileInputStream](#), [File](#)

**Version:**

1.17, 08/10/95

**Author:**

Arthur van Hoff

---

## Constructor Index

- o **[FileOutputStream](#)**(String)  
Creates an output file with the specified system dependent file name.
- o **[FileOutputStream](#)**(int)  
Creates an output file with the specified system dependent file descriptor.
- o **[FileOutputStream](#)**(File)  
Creates an output file with the specified File object.

## Method Index

- o **[close](#)**()  
Closes the stream.
- o **[finalize](#)**()  
Closes the stream when garbage is collected.
- o **[getFD](#)**()  
Returns the file descriptor associated with this stream.
- o **[write](#)**(int)

Writes a byte of data.

o **write**(byte[])

Writes an array of bytes.

o **write**(byte[], int, int)

Writes a sub array of bytes.

## Constructors

### o **FileOutputStream**

```
public FileOutputStream(String name) throws IOException
```

Creates an output file with the specified system dependent file name.

**Parameters:**

name – the system dependent file name

**Throws:** IOException

If the file is not found.

### o **FileOutputStream**

```
public FileOutputStream(int fd) throws IOException
```

Creates an output file with the specified system dependent file descriptor.

**Parameters:**

fd – the system dependent file descriptor

**Throws:** IOException

If an I/O error has occurred.

### o **FileOutputStream**

```
public FileOutputStream(File file) throws IOException
```

Creates an output file with the specified File object.

**Parameters:**

file – the file to be opened for reading

**Throws:** IOException

If the file is not found.

## Methods

### o **write**

```
public void write(int b) throws IOException
```

Writes a byte of data. This method will block until the byte is actually written.

**Parameters:**

b – the byte to be written

**Throws:** IOException

If an I/O error has occurred.

**Overrides:**

write in class OutputStream

**o write**

```
public void write(byte b[]) throws IOException
```

Writes an array of bytes. Will block until the bytes are actually written.

**Parameters:**

b – the data to be written

**Throws:** IOException

If an I/O error has occurred.

**Overrides:**

write in class OutputStream

**o write**

```
public void write(byte b[],  
                  int off,  
                  int len) throws IOException
```

Writes a sub array of bytes.

**Parameters:**

b – the data to be written

off – the start offset in the data

len – the number of bytes that are written

**Throws:** IOException

If an I/O error has occurred.

**Overrides:**

write in class OutputStream

**o close**

```
public void close() throws IOException
```

Closes the stream. This method must be called to release any resources associated with the stream.

**Throws:** IOException

If an I/O error has occurred.

**Overrides:**

close in class OutputStream

**o getFD**

```
public final int getFD()
```

Returns the file descriptor associated with this stream.

**Returns:**

the file descriptor.

**o finalize**

protected void finalize() throws [IOException](#)

Closes the stream when garbage is collected.

---

[All Packages](#)

[This Package](#)

[Previous](#)

[Next](#)