

# FileOutputStream

```
java.lang.Object
  java.io.OutputStream
    java.io.FileOutputStream
```

## Kılgıladığı Arayüzler:

Closeable, Flushable, AutoCloseable

## Bildirimi:

```
public class FileOutputStream
    extends OutputStream
```

`FileOutputStream`, bir dosyaya (file) ya da dosya belirleyiciye (file descriptor) yazan bir çıkış akımıdır. Bir dosyanın elde olması ya da yaratılması, alttaki platforma bağlıdır. Bazı platformlar, bir anda dosyayı yalnızca yazmak üzere açar, bazıları hem okumaya hem yazmaya izin verir. Açık olan bir dosya tekrar açılmak istenirse kurucu başarısız olur.

`FileOutputStream`, resim v.b. gibi raw byte yazar. Karakter yazmak için `FileWriter` sınıfı kullanılmalıdır.

## Kurucuları:

**FileOutputStream**(`File` file)

Creates a file output stream to write to the file represented by the specified `File` object.

**FileOutputStream**(`File` file, boolean append)

Creates a file output stream to write to the file represented by the specified `File` object.

**FileOutputStream**(`FileDescriptor` fdObj)

Creates a file output stream to write to the specified file descriptor, which represents an existing connection to an actual file in the file system.

**FileOutputStream**(`String` name)

Creates a file output stream to write to the file with the specified name.

**FileOutputStream**(`String` name, boolean append)

Creates a file output stream to write to the file with the specified name.

## Metotları:

```
void close()
```

Closes this file output stream and releases any system resources associated with this stream.

```
protected void finalize()
```

Cleans up the connection to the file, and ensures that the `close` method of this file output stream is called when there are no more references to this stream.

```
FileChannel getChannel()
```

Returns the unique `FileChannel` object associated with this file output stream.

```
FileDescriptor getFD()
```

Returns the file descriptor associated with this stream.

```
void write(byte[] b)
```

Writes `b.length` bytes from the specified byte array to this file output stream.

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

Writes `len` bytes from the specified byte array starting at offset `off` to this file output stream.

```
void write(int b)
```

Writes the specified byte to this file output stream.

*java.io.OutputStream Sınıfından Kalıtıl Gelenler:*

`flush`

*java.lang.Object Sınıfından Kalıtıl Gelenler:*

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

```
/*
  Flush output stream
  This Java example shows how to flush output stream using flush
  method of DataOutputStream class.
*/
import java.io.DataOutputStream;
import java.io.FileOutputStream;
import java.io.IOException;
public class FlushStream {
    public static void main(String[] args) {
        String strFilePath = "C://FileIO//WriteByte.txt";
        try
        {
```

```

//create FileOutputStream object
FileOutputStream fos = new FileOutputStream(strFilePath);
/*
 * To create DataOutputStream object from FileOutputStream
use,
 * DataOutputStream(OutputStream os) constructor.
 *
 */
DataOutputStream dos = new DataOutputStream(fos);
String strContent = "This example shows how to flush output
stream!";
dos.writeBytes(strContent);
/*
 * To flush output stream, use
 * void flush() method of DataOutputStream class.
 *
 * This method internally calls flush method of underlying
OutputStream
 * class which forces any buffered output bytes to be
written in the stream.
 */
dos.flush();
//close the stream
dos.close();
}
catch (IOException e)
{
    System.out.println("IOException : " + e);
}
}
}

```