

Flujos de salida con ficheros binarios: la clase FileOutputStream

```
import java.io.*;    //importar las clases
```

1. Abrir flujo de salida: FileOutputStream(String nombreArchivo)

```
FileOutputStream archivo = new FileOutputStream("C:\\programas\\prueba.dat"); //destruye fichero y comienza desde principio
```

```
FileOutputStream archivo = new FileOutputStream("C:\\programas\\prueba.dat", true); //añade a fichero
```

2. Crear objeto con métodos de escritura envolviendo al flujo: ObjectOutputStream

```
ObjectOutputStream out = new ObjectOutputStream(archivo);
```

3. Escribir (serializar) un valor boolean: void writeBoolean(boolean b)

```
out.writeBoolean(b);
```

4. Escribir un char en el fichero: void writeChar(int c)

```
out.writeChar(c);
```

5. Escribir un entero en el fichero: void writeInt(int n)

```
out.writeInt(n);
```

6. Escribir un entero largo en el fichero: void writeLong(long n)

```
out.writeLong(n);
```

7. Escribir un double en el fichero: void writeDouble(double d)

```
out.writeDouble(d);
```

8. Escribir un objeto serializable en el fichero: void writeObject(Object o)

```
out.writeObject(o); // string, tablas y colecciones son objetos
```

```
class miClase implements Serializable { // Ahora miClase es serializable
```

```
// cuerpo de la clase
```

```
}
```

9. Cierra el flujo y vacía el búfer de salida: void close():

```
out.close();
```

Flujos de entrada con ficheros binarios: la clase FileInputStream

```
import java.io.*;    //importar las clases
```

1. Abrir flujo de entrada: FileInputStream(String nombreArchivo)

```
FileInputStream archivo = new FileInputStream("C:\\programas\\prueba.dat");
```

2. Crear objeto con métodos de lectura envolviendo al flujo: ObjectInputStream

```
ObjectInputStream in = new ObjectInputStream(archivo);
```

3. Leer (deserializar) un valor boolean: boolean readBoolean()

```
b = in.readBoolean();
```

4. Leer un char en el fichero: char readChar()

```
c = in.readChar();
```

5. Leer un entero en el fichero: int readInt()

```
n = in.readInt();
```

6. Leer un entero largo en el fichero: long readLong()

```
n = long in.readLong();
```

7. Leer un double en el fichero: double readDouble()

```
d = double in.readDouble();
```

8. Leer un objeto serializable en el fichero: Object readObject()

```
o = in.readObject(); // string, tablas y colecciones son objetos
```

9. Cierra el flujo: void close():

```
in.close();
```