**Java FilterOutputStream Class**

Java FilterOutputStream class implements the OutputStream class. It provides different sub classes such as BufferedOutputStream and DataOutputStream to provide additional functionality. So it is less used individually.

**Java FilterOutputStream class declaration**

Let's see the declaration for java.io.FilterOutputStream class:

1. **public** **class** FilterOutputStream **extends** OutputStream

**Java FilterOutputStream class Methods**

|  |  |
| --- | --- |
| **Method** | **Description** |
| void write(int b) | It is used to write the specified byte to the output stream. |
| void write(byte[] ary) | It is used to write ary.length byte to the output stream. |
| void write(byte[] b, int off, int len) | It is used to write len bytes from the offset off to the output stream. |
| void flush() | It is used to flushes the output stream. |
| void close() | It is used to close the output stream. |

|  |  |
| --- | --- |
| **package** com.company; **import** java.io.\*; **public class** Main {   **public static void** main(String[] args) **throws** Exception{  **try** {  File file = **new** File(**"D:\\Cat.txt"**);  FileOutputStream fileOutputStream = **new** FileOutputStream(file);  FilterOutputStream filterOutputStream = **new** FilterOutputStream(fileOutputStream);  String s = **"I got the shit here"**;  **byte** byten[] = s.getBytes();  filterOutputStream.write(byten);  filterOutputStream.flush();  filterOutputStream.close();  fileOutputStream.close();  System.***out***.println(**"Success............"**);  } **catch** (Exception ex){  System.***out***.println(**"Message : "** + ex);  }  } } |  |