```
//
//this_java_13장_입출력_소스
//

//File : try-catch Exception
public class Test {
	public static void main(String[] args) {
		try {
			File dir = new File("E:/lecture_src/java_src/inout/Dir");
			File file1 = new File("E:/lecture_src/java_src/inout/file1.txt");
			File file2 = new File("E:/lecture_src/java_src/inout/file2.txt");
			File file3 = new File(new URI("file:///E:/lecture_src/java_src/inout/file3.txt"));

			if (dir.exists() == false) { dir.mkdirs(); }
			if (file1.exists() == false) { file1.createNewFile(); }
			if (file2.exists() == false) { file2.createNewFile(); }
			if (file3.exists() == false) { file3.createNewFile(); }
		} catch (Exception e) {
			System.out.println(e.getMessage());
		}
		System.out.println("결과를 윈도우탐색기로 확인해 보세요.");
	}
}

//File
public class Test {
	public static void main(String[] args) throws Exception {
		File dir = new File("E:/lecture_src/java_src/inout/Dir");
		File file1 = new File("E:/lecture_src/java_src/inout/file1.txt");
		File file2 = new File("E:/lecture_src/java_src/inout/file2.txt");
		File file3 = new File(new URI("file:///E:/lecture_src/java_src/inout/file3.txt"));

		if(dir.exists() == false) {   dir.mkdirs();   }
		if(file1.exists() == false) {   file1.createNewFile();   }
		if(file2.exists() == false) {   file2.createNewFile();   }
		if(file3.exists() == false) {   file3.createNewFile();   }

		System.out.println("결과를 윈도우탐색기로 확인해 보세요.");
	}
}

//FileOutputStream : try-catch Exception
public class Test {
	public static void main(String[] args) {
		try {
			OutputStream fos = new FileOutputStream("test1.txt");
			byte[] data = "ABC".getBytes();

			for(int i=0; i<data.length; i++) {
				fos.write(data[i]);
			}
			fos.flush();
			fos.close();
		} catch (Exception e) {
			System.out.println(e.getMessage());
		}
		System.out.println("결과를 윈도우탐색기로 확인해 보세요.");
	}
```

```
}

//FileInputStream
public class Test {
	public static void main(String[] args) throws Exception {
		InputStream fis = new FileInputStream(args[0]); //args[0]에 test1.txt 입력
		int readByte;
		while(true) {
			readByte = fis.read();
			if(readByte == -1) break;
			System.out.println((char)readByte);
		}
		fis.close();
	}
}

//FileInputStream, FileOutputStream
public class Test {
	public static void main(String[] args) throws Exception {
		String originalFileName = "E:/lecture_src/java_src/inout/smartlead.gif";
		String targetFileName = "E:/lecture_src/java_src/inout/copysmartlead.gif";

		FileInputStream fis = new FileInputStream(originalFileName);
		FileOutputStream fos = new FileOutputStream(targetFileName);

		int readByteNo;
		byte[] readBytes = new byte[100];
		while( (readByteNo = fis.read(readBytes)) != -1 ) {
			fos.write(readBytes, 0, readByteNo);
		}
		fos.flush(); fos.close(); fis.close();
		System.out.println("복사가 잘 되었습니다.");
		System.out.println("결과를 윈도우탐색기로 확인해 보세요.");
	}
}

//FileInputStream, BufferedInputStream
public class Test {
	public static void main(String[] args) throws Exception {
		long start=0, end = 0;

		FileInputStream fis1 = new FileInputStream("smartlead.gif");

		start = System.currentTimeMillis();
		while(fis1.read() != -1) {}
		end = System.currentTimeMillis();

		fis1.close();
		System.out.println("사용하지 않았을 때: " + (end-start) + "ms");

		FileInputStream fis2 = new FileInputStream("smartlead.gif");
		BufferedInputStream bis = new BufferedInputStream(fis2);

		start = System.currentTimeMillis();
		while(bis.read() != -1) {}
		end = System.currentTimeMillis();

		bis.close(); fis2.close();
		System.out.println("사용했을 때: " + (end-start) + "ms");
```

```
	}
}

//DataInputStream, DataOutputStream
public class Test {
	public static void main(String[] args) throws Exception {
		FileOutputStream fos = new FileOutputStream("dos.dat");
		DataOutputStream dos = new DataOutputStream(fos);

		dos.writeUTF("홍길동");
		dos.writeDouble(95.5);
		dos.writeInt(1);

		dos.writeUTF("감자바");
		dos.writeDouble(90.3);
		dos.writeInt(2);

		dos.flush(); dos.close(); fos.close();

		FileInputStream fis = new FileInputStream("dos.dat");
		DataInputStream dis = new DataInputStream(fis);

		for(int i=0; i<2; i++) {
			String name = dis.readUTF();
			double score = dis.readDouble();
			int order = dis.readInt();
			System.out.println(name + " : " + score + " : " + order);
		}
		dis.close(); fis.close();
	}
}

//ObjectOutputStream, ObjectInputStream
public class Test {
	public static void main(String[] args) throws Exception {
		FileOutputStream fos = new FileOutputStream("object.dat");
		ObjectOutputStream oos = new ObjectOutputStream(fos);

		oos.writeObject(new Integer(10));
		oos.writeObject(new Double(3.14));
		oos.writeObject(new int[] { 1, 2, 3 });
		oos.writeObject(new String("홍길동"));

		oos.flush(); oos.close(); fos.close();

		FileInputStream fis = new FileInputStream("object.dat");
		ObjectInputStream ois = new ObjectInputStream(fis);

		Integer obj1 = (Integer) ois.readObject();
		Double obj2 = (Double) ois.readObject();
		int[] obj3 = (int[]) ois.readObject();
		String obj4 = (String) ois.readObject();

		ois.close(); fis.close();

		System.out.println(obj1);
		System.out.println(obj2);
		System.out.println(obj3[0] + "," + obj3[1] + "," + obj3[2]);
		System.out.println(obj4);
```

```
	}
}

//Serializable, ObjectOutputStream, ObjectInputStream
class ClassA implements Serializable {
	int field1;
	ClassB field2 = new ClassB();
	static int field3;
	transient int field4;
}
class ClassB implements Serializable {
	int field1;
}
public class Test {
	public static void main(String[] args) throws Exception {
		FileOutputStream fos = new FileOutputStream("objectAB.dat");
		ObjectOutputStream oos = new ObjectOutputStream(fos);

		ClassA classA = new ClassA();

		classA.field1 = 1;
		classA.field2.field1 = 2;
		classA.field3 = 3;
		classA.field4 = 4;

		oos.writeObject(classA);

		oos.flush(); oos.close(); fos.close();

		FileInputStream fis = new FileInputStream("objectAB.dat");
		ObjectInputStream ois = new ObjectInputStream(fis);

		ClassA v = (ClassA) ois.readObject();

		System.out.println("field1: " + v.field1);
		System.out.println("field2.field1: " + v.field2.field1);
		System.out.println("field3: " + v.field3);
		System.out.println("field4: " + v.field4);
	}
}

//FileWriter
public class Test {
	public static void main(String[] args) throws Exception {
		Writer fw = new FileWriter("E:/lecture_src/java_src/inout/test2.txt");

		char[] data = "홍길동".toCharArray();

		for(int i=0; i<data.length; i++) {
			fw.write(data[i]);
		}
		fw.flush(); fw.close();
		System.out.println("결과를 윈도우탐색기로 확인해 보세요.");
	}
}

//FileReader
public class Test {
	public static void main(String[] args) throws Exception {
```

```
		Reader fr = new FileReader("E:/lecture_src/java_src/inout/test2.txt");
		int readData;
		while( true ) {
			readData = fr.read();
			if(readData == -1) break;
			System.out.print((char)readData);
		}
		fr.close();
	}
}

//FileWriter, 한글
public class Test {
	public static void main(String[] args) throws Exception {
		//File file = new File("E:/lecture_src/java_src/inout/koreanfile.txt");
		//FileWriter fw = new FileWriter(file, true);
		FileWriter fw = new FileWriter("E:/lecture_src/java_src/inout/koreanfile.txt", true);

		fw.write("FileWriter는 한글로된 " + "\r\n");
		fw.write("문자열도 바로 출력할 수 있습니다." + "\r\n");

		fw.flush(); fw.close();
		System.out.println("결과를 윈도우탐색기로 확인해 보세요.");
	}
}

//FileReader, 한글
public class Test {
	public static void main(String[] args) throws Exception {
		FileReader fr = new FileReader("E:/lecture_src/java_src/inout/koreanfile.txt");
		int readCharNo;
		char[] cbuf = new char[100];

		while ((readCharNo=fr.read(cbuf)) != -1) {
			String data = new String(cbuf, 0, readCharNo);
			System.out.print(data);
		}
		fr.close();
	}
}

//inetaddress
public class InetAddressExample {
	public static void main(String[] args) {
		try {
			InetAddress local = InetAddress.getLocalHost();
			System.out.println("내컴퓨터 IP주소: " + local.getHostAddress());

			InetAddress [] nn = InetAddress.getAllByName("www.naver.net");
			InetAddress [] nc = InetAddress.getAllByName("www.naver.com");
			InetAddress [] dn = InetAddress.getAllByName("www.daum.net");

			for(InetAddress remote : nn) System.out.println(remote);
			for(InetAddress remote : nc) System.out.println(remote);
			for(InetAddress remote : dn) System.out.println(remote);

		} catch(UnknownHostException e) { e.printStackTrace(); }
	}
}
```