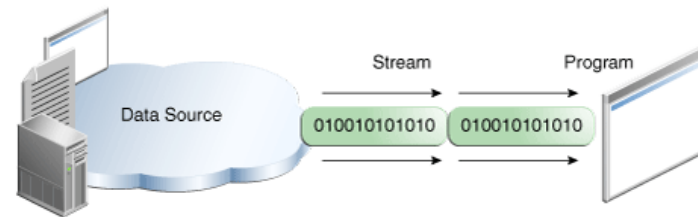


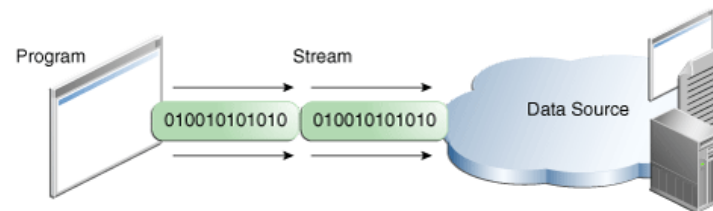
# Java Files (Multiple Windows )

# Stream

- An *I/O Stream* represents an input source or an output destination
- The streams that can handle all kinds of data, from primitive values to advanced objects.
- The data source and data destination pictured above can be anything that holds, generates, or consumes data:
  - disk files
  - another program,
  - a peripheral device,
  - a network socket,
  - or an array



Reading information into a program



Writing information from a program

# Java I.O

- Table listing most Java IO classes divided by input, output, being byte based or character based, and any more specific purpose.

	Byte Based		Character Based	
	Input	Output	Input	Output
Basic	InputStream	OutputStream	InputStreamReader	OutputStreamWriter
Arrays	ByteArrayInputStream	ByteArrayOutputStream	CharArrayReader	CharArrayWriter
Files	<b>FileInputStream</b> <b>RandomAccessFile</b>	<b>FileOutputStream</b> <b>RandomAccessFile</b>	FileReader	FileWriter
Buffering	BufferedInputStream	BufferedOutputStream	<b>BufferedReader</b>	<b>BufferedWriter</b>
Strings			StringReader	StringWriter
Data	DataInputStream	DataOutputStream		
Data - Formatted		PrintStream		PrintWriter
Objects	<b>ObjectInputStream</b>	<b>ObjectOutputStream</b>		

# Buffered(Writer/Reader)

```
FileWriter fw = new FileWriter("your_file.txt");
BufferedWriter bw = new BufferedWriter(fw);
bw.writeData(content);
bw.flush();
bw.close();
```

```
FileReader fr = new FileReader("your_file.txt");
BufferedReader br = new BufferedReader(fr);

String sCurrentLine;
// read until the end of file
while ((sCurrentLine = br.readLine()) != null) {
    System.out.println(sCurrentLine);
}
br.close();
```

# Object(Output/Input)Stream

```
FileOutputStream fo = new FileOutputStream("your_file.data")
ObjectOutputStream out = new ObjectOutputStream(fo);
// make sure the object is serializable
out.writeObject(yourSerializableObject);
out.close();
```

```
public class YourClass implements Serializable{}
```

```
// create file stream and object stream
FileInputStream fi = new FileInputStream("your_file.data");
ObjectInputStream oistream = new ObjectInputStream(fi);

// read until the end of file input stream
while (istream.available() > 0){
    // casting here
    YourClass readObj = (YourClass)oistream.readObject();
}
oistream.close();
```

# 3. Implements

## Add action listener for Buttons

- Declare event handler class and implements the methods in listener interface.

```
public class FileIoDemo extends JFrame implements ActionListener {  
    // Declare fields  
    // Implement methods  
    // ...  
  
    public void actionPerformed(ActionEvent e) {  
        //Handle open button action.  
        if (e.getSource() == mnbtOpen) {  
            // Open file here  
        }else if (e.getSource() == mnbtSave) {  
            // Save file here  
        }else if (e.getSource() == mnbtExit){  
            FileIoDemo.this.dispose();  
        }  
    }  
}
```

- Register an instance of the event handler class as a listener on menu item components.

```
mnbtOpen = new JMenuItem("Open");  
mnbtOpen.addActionListener(this);  
  
mnbtSave = new JMenuItem("Save");  
mnbtSave.addActionListener(this);
```

# 3. Implements

## Event handling

- Open selected file:

```
if (e.getSource() == mnbtOpen) {  
    // create file chooser and show it  
    JFileChooser fc = new JFileChooser();  
    int returnVal = fc.showOpenDialog(FileIoDemo.this);  
  
    // check if user action  
    if (returnVal == JFileChooser.APPROVE_OPTION) {  
        File file = fc.getSelectedFile();  
        Path path = Paths.get(file.getAbsolutePath());  
  
        // put your code to ready from file  
  
    }  
}
```

# 3. Implements

## Event handling

- Save to selected file:

```
else if (e.getSource() == mnbtSave) {  
    // create file chooser and show it  
    JFileChooser fc = new JFileChooser();  
    int returnVal = fc.showSaveDialog(FileIoDemo.this);  
  
    // check if user action  
    if (returnVal == JFileChooser.APPROVE_OPTION) {  
        File file = fc.getSelectedFile();  
        Path path = Paths.get(file.getAbsolutePath());  
  
        // put your code to write to file  
  
    }  
}
```





PRACTICE

Secret Notepad Registration Form

Full Name

Tamer ABUHMED

Gender

☒ Male ☐ Female

User Name

JavaUser

Password

.....

Confirm Password

.....|

Register Cancel

Secret Notepad Registration Form

? Thank you  
You are registered

OK

Secret NotePad Login

User Name

Password

OK Cancel

Secret NotePad

File Edit Help

New

Open

Save

Exit