



(remedial) Java

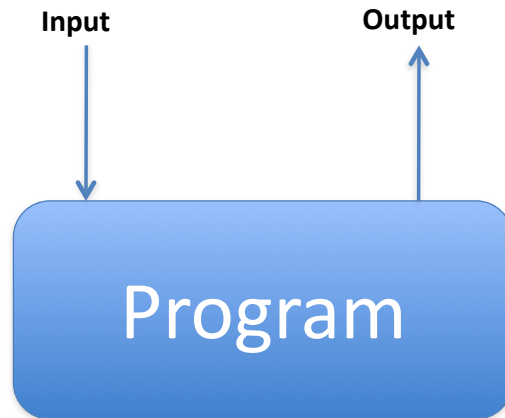
anastasia.bezerianos@lri.fr

ABezerianos - Remedial Java - Session-5.key - 6 September 2019

I/O

ABezerianos - Remedial Java - Session-5.key - 6 September 2019

Input/Output



ABezarianos - Remedial Java - Session-5.key - 6 September 2019

We've seen output

```
System.out.println("some string");
```

ABezarianos - Remedial Java - Session-5.key - 6 September 2019

Standard input/output

- Standard output (console)

```
System.out.println(...)
```

```
System.err.println(...)
```

- Standard input (keyboard)

```
InputStreamReader cin = new  
    InputStreamReader(System.in);
```

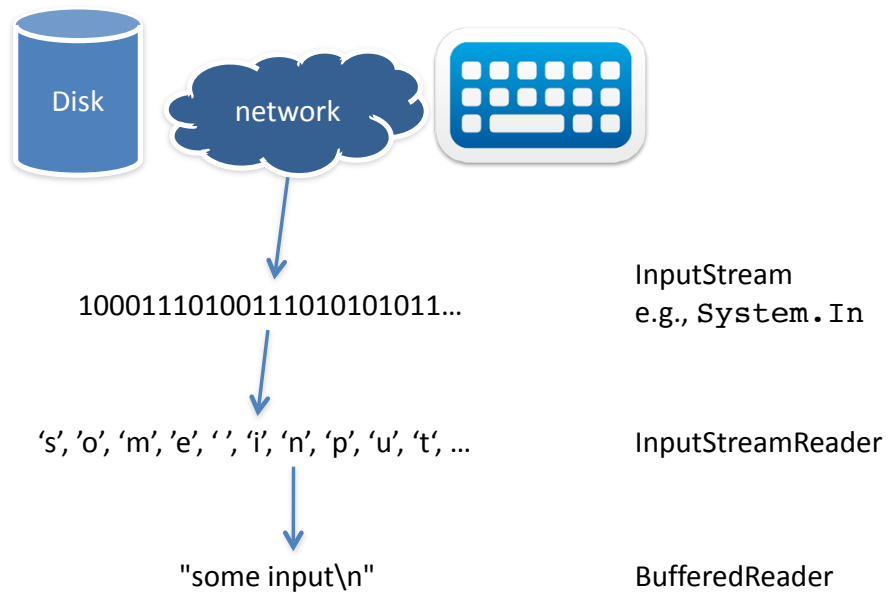
ABezarianos - Remedial Java - Session-5.key - 6 September 2019

An example of input

```
//imports you'll need?  
public class simpleIO {  
  
    public static void main (String[] args){  
  
        System.out.print("Give me some input: ");  
        InputStreamReader stream = new  
            InputStreamReader(System.in);  
        BufferedReader keyboardInput = new BufferedReader (stream);  
  
        //this will "almost" work, deal with i/o exceptions  
        input = keyboardInput.readLine();  
        System.out.println("Your input was: " + input);  
    }  
}
```

ABezarianos - Remedial Java - Session-5.key - 6 September 2019

The full picture



ABezarianos - Remedial Java - Session-5.key - 6 September 2019

InputStream

- `InputStream` is a stream of bytes
 - Read one byte after another using `read()`
- A byte is just a number
 - Data on your hard drive is stored in bytes
 - Bytes can be interpreted as characters, numbers...

```
InputStream stream = System.in;
```

ABezarianos - Remedial Java - Session-5.key - 6 September 2019

InputStreamReader

- *Reader* is a class for reading character streams
 - Read one character after another using `read()`
- `InputStreamReader` takes an `InputStream` and converts bytes to characters
- Still inconvenient
 - Can only read a character at a time..

```
new InputStreamReader(stream);
```

ABezarianos - Remedial Java - Session-5.key - 6 September 2019

BufferedReader

- `BufferedReader` buffers a character stream so that you can read it line by line
 - Read line using `readLine()`

```
new BufferedReader (  
    new InputStreamReader(System.in));
```

ABezarianos - Remedial Java - Session-5.key - 6 September 2019

User input

```
InputStreamReader stream =  
    new InputStreamReader(System.in);  
  
BufferedReader keyboardInput =  
    new BufferedReader (stream);  
  
String answer = keyboardInput.readLine();
```

ABezerianos - Remedial Java - Session-5.key - 6 September 2019

Exercise 3 – part one

www.lri.fr/~anab/teaching/remedialJava/ex-class4.pdf

Reminder:

```
InputStreamReader stream =  
    new InputStreamReader(System.in);  
  
BufferedReader keyboardInput =  
    new BufferedReader (stream);  
  
String answer = keyboardInput.readLine();
```

- You can make a string all lower case by calling on a String object: `str.toLowerCase()`

ABezerianos - Remedial Java - Session-5.key - 6 September 2019

Reading Strings

ABezarianos - Remedial Java - Session-5.key - 6 September 2019

String manipulation

- When reading a line from `StdIn` (or from a file), it is returned as a string. There are useful methods for manipulating strings you receive as input:

- split strings when a character (e.g., `","`) is found, and put the pieces into an array of strings

```
String[] split = str.split(",");
```

- trim empty spaces in beginning/end of string

```
String trimmed = str.trim();
```

ABezarianos - Remedial Java - Session-5.key - 6 September 2019

Aside: string manipulation (cont'd)

- Others, that are usefull when testing the content of the string (self explanatory)

```
startsWith(s1), endsWith(s1), substring(s1)  
toUpperCase(), toLowerCase(),  
equals(s1), isEmpty(), length()
```

ABezarianos - Remedial Java - Session-5.key - 6 September 2019

Aside: string manipulation (cont'd)

- Finally, there are methods to convert a string to another type

```
Integer.parseInt( str ) ;  
// e.g., int i = Integer.parseInt( "100" ) ;  
  
double myDouble = Double.parseDouble(str);  
// e.g., double d = Double.parseDouble("2.5");  
  
boolean myBool = Boolean.parseBoolean(str);  
// e.g., boolean b = Boolean.parseBoolean("true");
```

ABezarianos - Remedial Java - Session-5.key - 6 September 2019

Exercise 3 part two and three

www.lri.fr/~anab/teaching/remedial-java/ex-class4.pdf

Reminder:

- Some conversion methods

```
int i = Integer.parseInt( str ) ;  
double d = Double.parseDouble (str);
```

- Some String manipulation methods

```
String[] split = str.split(",");  
String trimmed = str.trim();
```

ABezarianos - Remedial Java - Session-5.key - 6 September 2019

Reading from / writing to Files

ABezarianos - Remedial Java - Session-5.key - 6 September 2019

FileReader

- **FileReader** takes a text file
 - converts it into a character stream
 - `FileReader("PATH TO FILE");`
- Use this and **BufferedReader** to read files

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

ABezerianos - Remedial Java - Session-5.key - 6 September 2019

An example of reading a file

```
//imports you'll need?  
public class simpleReadFile {  
  
    public static void main (String[] args)  
        throws IOException{  
        //Eclipse quirk: path names are relative to project dir  
        FileReader fr = new FileReader("readme.txt");  
        BufferedReader bf = new BufferedReader (fr);  
  
        String line = null;  
        while( (line = bf.readLine()) != null ) {  
            System.out.println(line);  
        }  
        bf.close(); // always close the stream !!!  
    }  
}
```

ABezerianos - Remedial Java - Session-5.key - 6 September 2019

Exercise 4

www.lri.fr/~anab/teaching/remedialJava/ex-class4.pdf

Reminder:

- Some conversion methods

```
int i = Integer.parseInt( str ) ;  
double d = Double.parseDouble (str);
```

- Some String manipulation methods

```
String[] split = str.split(",");  
String trimmed = str.trim();
```

```
FileReader fr = new FileReader("readme.txt");  
BufferedReader bf = new BufferedReader (fr);  
String line = null;  
while( (line = bf.readLine()) != null )  
    System.out.println(line);  
}  
bf.close();
```

ABezerianos - Remedial Java - Session-5.key - 6 September 2019

Input/Output goals

- The fundamental concept is that of a **stream**
 - the VM connects to, or creates a data stream

ABezerianos - Remedial Java - Session-5.key - 6 September 2019

Input/Output goals

Input/Output (I/O) is the way a program communicates with it's environment:

- Where does the I/O come from?
 - disk file, console, network ...
- How is it formatted?
 - binary, byte, character
 - per line, per word, per character
 - sequentially, random access
 - buffer

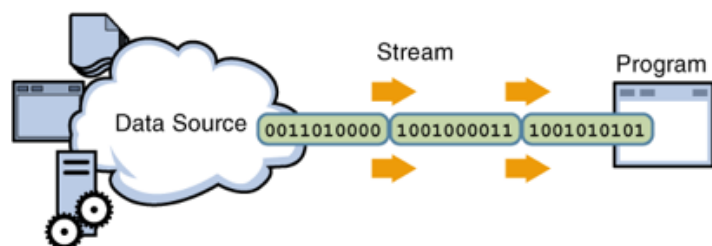
ABezarianos - Remedial Java - Session-5.key - 6 September 2019

Input/Output stream

Input:

e.g. read a file

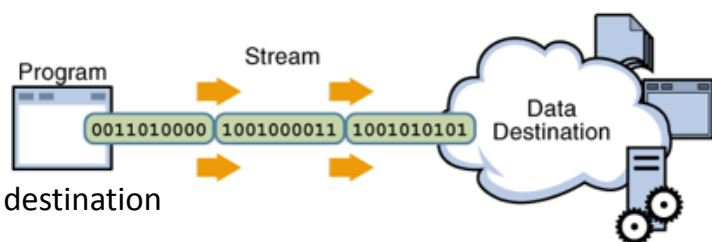
Program is the destination



Output:

e.g. write in a file

Program is the origin, file the destination



ABezarianos - Remedial Java - Session-5.key - 6 September 2019

Input/Output goals

- The package `java.io` regroups classes for I/O
- It is divided in two categories:
 - input and output
 - and for each, 2 abstract classes, one that handles bytes, one that handles characters

ABezarianos - Remedial Java - Session-5.key - 6 September 2019

java.io (input)

The input classes have the methods
`read()` or `read(x)`

- `InputStream` with method:

```
public int read( byte[] b ) throws IOException
```

- `Reader` with method:

```
public int read( char[] c ) throws IOException
```

Always close stream with `close()`

ABezarianos - Remedial Java - Session-5.key - 6 September 2019

java.io (input)

- The `BufferedReader` is a wrapper class that converts an input stream to Strings:

- Reader with method:

```
public String readLine() throws IOException
```

Always close stream with `close()`

java.io (output)

The output classes have the method
`write(x)`

- `OutputStream` with method:

```
public void write( byte[] b ) throws IOException
```

- `Writer` with methods:

```
public void write( char[] c ) throws IOException  
append()
```

Always close stream with `close()`

java.io (output)

- Similarly, `BufferedWriter` is a wrapper class that converts an output stream to `String`:
- `Writer` with method:

```
public String write(String) throws IOException
```

Always close stream with `close()`

ABezarianos - Remedial Java - Session-5.key - 6 September 2019

An example of writing to a file

```
//imports you'll need?
public class simpleWriteFile {

    public static void main (String[] args)
        throws IOException{
        //Eclipse quirk: path names are relative to project dir
        String[] myText = {"this", "is", "my", "text"};

        BufferedWriter bf = new BufferedWriter (
            new FileWriter("myFile.txt"));

        for( int i =0; i < myText.length; ++i) {
            bf.write(myText[i] + "\n");
        }
        bf.close(); // always close the stream !!!
    }
}
```

ABezarianos - Remedial Java - Session-5.key - 6 September 2019

A read/write example

```
public class CopyCharacters {  
    public static void main(String[] args) throws IOException {  
        FileReader input = null; // input stream as characters  
        FileWriter output = null; // output stream as characters  
  
        input = new FileReader("original.txt"); // open file  
        output = new FileWriter("copie.txt"); // open file  
  
        int c; // represents a byte  
        while ((c = input.read()) != -1) { // read one by one  
            // chars until the end  
            output.write(c); // c as int  
        }  
  
        if (input != null) {  
            input.close(); // close the input stream  
        }  
        if (output != null) {  
            output.close(); // close the output stream  
        }  
    }  
}
```

ABezarianos - Remedial Java - Session-5.key - 6 September 2019

Exercise 7

www.lri.fr/~anab/teaching/remedialJava/ex-class4.pdf

Reminder:

```
BufferedWriter bf = new BufferedWriter (  
    new FileWriter("myFile.txt"));  
  
bf.write( str );  
  
bf.close();
```

ABezarianos - Remedial Java - Session-5.key - 6 September 2019

The class File

ABezerianos - Remedial Java - Session-5.key - 6 September 2019

File

- The class `java.io.File` helps with file manipulation independent of platform
- A better name could be “FileName” or “FilePath”, as it represents the name of a file or files in a directory
- Interesting methods
 - `getAbsolutePath()`
 - `exists()`
 - `isDirectory()`
 - `listFiles()`
 - `list (FilenameFilter ff)`

ABezerianos - Remedial Java - Session-5.key - 6 September 2019