CMSC 132:
Object-Oriented Programming II

Java I/O Overview

Department of Computer Science

University of Maryland, College Park

Input/Output

- **Approaches to store file data**
  - **Text files**
    - **Data represented in human-readable form**
    - **Example: Java source programs**
    - **Use text editor to manipulate the data**
  - **Binary files**
    - **Data represented in binary form**
    - **Designed to be read by programs**
    - **More compact**
    - **More efficient to process (no encoding/decoding required)**
- **Keep in mind all files are stored in binary format**
- **Text I/O provides a level of abstraction to encode/decode characters**

Input/Output

- Several Java I/O classes
- Relying on java.io.*
- I/O classes can be categorized in input/output classes and whether they deal with text or binary data
- Input Object (Input Stream)
  - Object reads a stream of data from a file
- Output Object (Output Stream)
  - Object writes a stream of data to a file

Input/Output

- InputStream class – abstract superclass of all classes representing an input stream of bytes
- OuputStream class – abstract superclass of all classes representing an output stream of bytes
- File class
  - Encapsulates the properties of a file or directory
  - Does not provide methods to read/write from/to a file
  - Example: File f = new File("data.txt");
  - Methods
    - exists() → tests whether file/directory exists
    - delete() → deletes file/directory
    - others

Text Files Input Classes

- **FileReader**
  - read method – returns a character or -1 (end of stream)
  - close method – closes the stream and releases any system resources
  - Example: FileReaderEx.java
- **BufferedReader**
  - Reads text from character-input stream, buffering characters for efficiency
  - readLine method – reads a line of text
  - Example: BufferedReaderEx.java
- **Scanner**
  - Breaks input into tokens delimited by whitespace
  - Methods: hasNext(), nextInt(), nextDouble(), next(), Others
  - Example: ScannerEx.java

Text Files Output Classes

- **FileWriter**
  - **write(int c) method – writes a single character**
  - **close method – closes the stream and releases any system resources**
  - **Example: FileWriterEx.java**
- **BufferedWriter**
  - **Writes text to a character-output stream, buffering characters for efficiency**
  - **Example: BufferedWriterEx.java**
- **PrintWriter**
  - **print method**
  - **println method**
  - **printf method**
  - **Example: PrintWriterEx.java, FileReadWriteEx.java**

Binary Files Input/Output Classes Overview

- **Output**
  - **FileOutputStream – for writing bytes to a file**
  - **BufferedOutputStream – adds a buffer**
  - **DataOutputStream – converts primitive type values or strings into bytes and outputs them to the stream**
  - **Example: BinaryFileWriterEx.java**
- **Input**
  - **FileInputStream – for reading bytes from a file**
  - **BufferedInputStream – adds a buffer**
  - **DataInputStream – reads data from a stream and converts data into appropriate primitive type or strings**
  - **Example: BinaryFileReaderEx.java**

Standard Input/Output

- **Standard I/O**
  - **Provided in System class in java.lang**
  - **System.in**
    - **An instance of InputStream**
  - **System.out**
    - **An instance of PrintStream**
  - **System.err**
    - **An instance of PrintStream**
- **We can use the Scanner class with System.in**