CSCI 132:
Basic Data Structures and Algorithms

Java file I/O

Reese Pearsall

Spring 2024

Announcements

Lab 2 due **tomorrow** at 11:59 PM

Program1 will be posted later this week

Schedule is changing a little bit

Reading in a file in Java

```
FileReader file;
try {
    file = new FileReader("video_games.csv");

    Scanner inFile = new Scanner(file);

    inFile.nextLine();

    while(inFile.hasNext()) {
            String line = inFile.nextLine();
            System.out.println(line);

            String[] splitted_line = line.split(",");
            System.out.println(splitted_line[0]);

} catch (IOException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}
```

Opening a reading a file must be done inside of a try/catch statement

Reading in a file in Java

Open a file with the `FileReader` library

```
FileReader file;
try {
    file = new FileReader("video_games.csv");

    Scanner inFile = new Scanner(file);

    inFile.nextLine();

    while(inFile.hasNext()) {
            String line = inFile.nextLine();
            System.out.println(line);

            String[] splitted_line = line.split(",");
            System.out.println(splitted_line[0]);

} catch (IOException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}
```

Reading in a file in Java

```
FileReader file;
try {
    file = new FileReader("video_games.csv");

    Scanner inFile = new Scanner(file);

    inFile.nextLine();

    while(inFile.hasNext()) {
            String line = inFile.nextLine();
            System.out.println(line);

            String[] splitted_line = line.split(",");
            System.out.println(splitted_line[0]);

} catch (IOException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}
```

Open a file with the `FileReader` library

Read contents of file with Scanner

Reading in a file in Java

```
FileReader file;
try {
    file = new FileReader("video_games.csv");

    Scanner inFile = new Scanner(file);

    inFile.nextLine();

    while(inFile.hasNext()) {
            String line = inFile.nextLine();
            System.out.println(line);

            String[] splitted_line = line.split(",");
            System.out.println(splitted_line[0]);

} catch (IOException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}
```

Open a file with the `FileReader` library

Read contents of file with Scanner

Read headers of csv file (discard them)

Reading in a file in Java

```
FileReader file;
try {
    file = new FileReader("video_games.csv");

    Scanner inFile = new Scanner(file);

    inFile.nextLine();

    while(inFile.hasNext()) {
            String line = inFile.nextLine();
            System.out.println(line);

            String[] splitted_line = line.split(",");
            System.out.println(splitted_line[0]);

} catch (IOException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}
```

Open a file with the `FileReader` library

Read contents of file with Scanner

Read headers of csv file (discard them)

Iterate through each line of file

Reading in a file in Java

```
FileReader file;
try {
    file = new FileReader("video_games.csv");

    Scanner inFile = new Scanner(file);

    inFile.nextLine();

    while(inFile.hasNext()) {
            String line = inFile.nextLine();
            System.out.println(line);

            String[] splitted_line = line.split(",");
            System.out.println(splitted_line[0]);

} catch (IOException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}
```

Open a file with the `FileReader` library

Read contents of file with Scanner

Read headers of csv file (discard them)

Iterate through each line of file

Get each line and print it out

Reading in a file in Java

```
FileReader file;
try {
    file = new FileReader("video_games.csv");

    Scanner inFile = new Scanner(file);

    inFile.nextLine();

    while(inFile.hasNext()) {
            String line = inFile.nextLine();
            System.out.println(line);

            String[] splitted_line = line.split(",");
            System.out.println(splitted_line[0]);
```

Open a file with the `FileReader` library

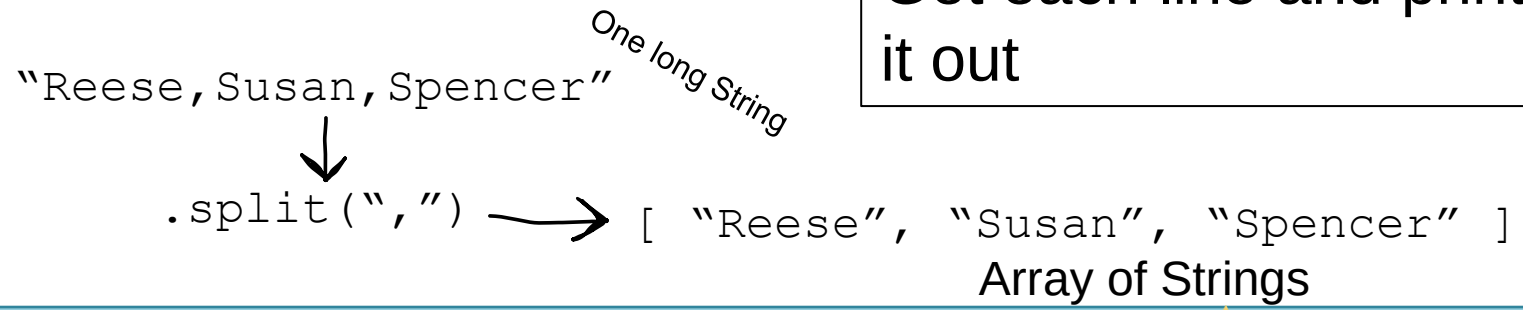
Read contents of file with Scanner

Read headers of csv file (discard them)

Iterate through each line of file

Get each line and print it out

To print out the first column of the file, we can split each line, and print out the first column



Writing to a file in Java

```
try {
    BufferedWriter writer = new BufferedWriter(new FileWriter("output.txt"));
    writer.write("Hello World! \n");
    writer.write("----------------------------\n");
    writer.close();
} catch (IOException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}
```

Writing to a file in Java

```
try {
    BufferedWriter writer = new BufferedWriter(new FileWriter("output.txt"));
    writer.write("Hello World! \n");
    writer.write("----------------------------\n");
    writer.close();
} catch (IOException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}
```

Name of output file

Writing to a file in Java

```
try {
    BufferedWriter writer = new BufferedWriter(new FileWriter("output.txt"));
    writer.write("Hello World! \n");
    writer.write("----------------------------\n");
    writer.close();
} catch (IOException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}
```

We write out contents to file using the `write()` method

Writing to a file in Java

```
try {
    BufferedWriter writer = new BufferedWriter(new FileWriter("output.txt"));
    writer.write("Hello World! \n");
    writer.write("----------------------------\n");
    writer.close();
} catch (IOException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}
```

When writing to a file, it is very important to close the file when you are finished