

Crescer 2015/2

Java

Pacote java.io

- ▶ Classes abstratas para leitura / escrita de bytes
 - ▶ InputStream
 - ▶ AudioInputStream, ByteArrayInputStream, FileInputStream
 - ▶ OutputStream
 - ▶ ByteArrayOutputStream, FileOutputStream

Pacote java.io

- ▶ Classes abstratas para leitura / escrita de caracteres
 - ▶ Reader
 - ▶ BufferedReader, CharArrayReader, FilterReader,
 - ▶ Writer
 - ▶ BufferedWriter, CharArrayWriter, FilterWriter, OutputStreamWriter, PipedWriter, PrintWriter, StringWriter

Pacote java.io

▶ File

- ▶ Aponta para um arquivo (não precisa existir)
- ▶ Verifica se é um diretório
 - ▶ Pode retornar uma lista de arquivos
- ▶ Pode criar ou deletar o arquivo ou diretório
- ▶ Renomear
- ▶ Marcar como somente leitura
- ▶ Alterar data de última modificação

Lendo em um arquivo - v1

```
try {  
    BufferedReader buff = new BufferedReader(new FileReader("C:\\temp\\crescer.txt"));  
    String linha = buff.readLine();  
    while (linha != null) {  
        System.out.println(linha);  
        linha = buff.readLine();  
    }  
    buff.close();  
} catch (IOException e){  
    System.err.printf("Falha ao ler arquivo " + e.getMessage());  
}
```

Lendo em um arquivo - v2

► A partir do Java 7 (Interface AutoCloseable)

```
try (BufferedReader buff = new BufferedReader(new FileReader("C:\\temp\\crescer.txt"))) {  
    String linha = buff.readLine();  
    while (linha != null) {  
        System.out.println(linha);  
        linha = buff.readLine();  
    }  
} catch (IOException e) {  
    System.err.printf("Falha ao ler arquivo " + e.getMessage());  
}
```

Lendo em um arquivo - v3

```
try {  
    File file = new File("c:\\temp\\crescer.txt");  
    Scanner scanner = new Scanner(file); // useDelimiter  
    while (scanner.hasNext()) {  
        System.out.println(scanner.next());  
    }  
    scanner.close();  
} catch (FileNotFoundException e) {  
    System.err.printf("Falha ao ler arquivo " + e.getMessage());  
}
```

Lendo em um arquivo v4

```
try {  
    String dados = new String(Files.readAllBytes(new File("c:\\temp\\crescer.txt").toPath()));  
    System.out.println(dados);  
} catch (IOException e) {  
    System.err.printf("Falha ao ler arquivo " + e.getMessage());  
}
```


Pacote java.nio

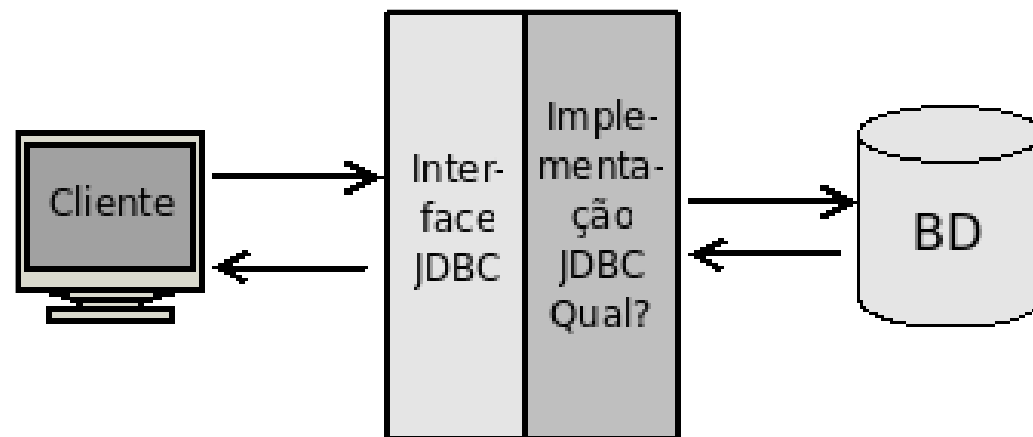
- ▶ Java 7
- ▶ Classes
 - ▶ Path: Estrutura de arquivo (ou diretório)
 - ▶ Files: possui métodos de operação de arquivos

Exercício

- ▶ Inclua no menu do exercício da primeira aula a opção 'Imprimir Lista', que imprime os elementos de uma estrutura (também informada pelo usuário) em um novo arquivo, que também será informado pelo teclado
 - ▶ `java.io.BufferedWriter`

JDBC

► *Java Database Connectivity*



Fonte: Java para Desenvolvimento Web (FJ-21 Caelum)

JDBC

- ▶ Pacote java.sql
 - ▶ Connection
 - ▶ PreparedStatement
 - ▶ ResultSet
- ▶ Implementações
 - ▶ De acordo com o SGBD
 - ▶ Utiliza o protocolo necessário para comunicação com o SGBD (driver)

Criando uma conexão

- ▶ `Connection conexao = DriverManager.getConnection(<String de conexão>)`
- ▶ String de conexão:
 - ▶ `jdbc:mysql://localhost/crescer`
 - ▶ `jdbc:oracle:thin:@10.0.100.16:1521:odev`
- ▶ Incluindo dependência no projeto

Fábrica de Conexões

```
public class ConnectionFactory {  
    public Connection getConnection() {  
        return DriverManager.getConnection("jdbc:mysql://localhost/crescer", "root", "");  
    }  
}
```

Criando entidade e DAO

- ▶ Entidade Cliente

- ▶ ClienteDAO

 - ▶ insert

 - ▶ findAll

Exercício

- ▶ Implementar os métodos insert e findAll na classe ServicoDao