Xử lý ngoại lệ và Đọc-Ghi file trong Java

CN.TRẦN HẢI LONG

BỘ MÔN CNPM – KHOA CÔNG NGHỆ THÔNG TIN – TRƯỜNG ĐHSPHN

EMAIL: longth@hnue.edu.vn

PHONE: 0966736098

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

Nội dung

I. Xử lý ngoại lệ

II. Đọc-Ghi file

III. Serializable Interface

IV. Đọc-Ghi Object vào file

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

I. Xử lý ngoại lệ

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

I. Xử lý ngoại lệ

1.1 Khái niệm cơ bản

1.2 Khối lệnh try…catch

1.3 Từ khóa throw

1.4 Từ khóa throws

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

I. Xử lý ngoại lệ

1.1 Khái niệm cơ bản

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

I. Xử lý ngoại lệ

1.1 Khái niệm cơ bản

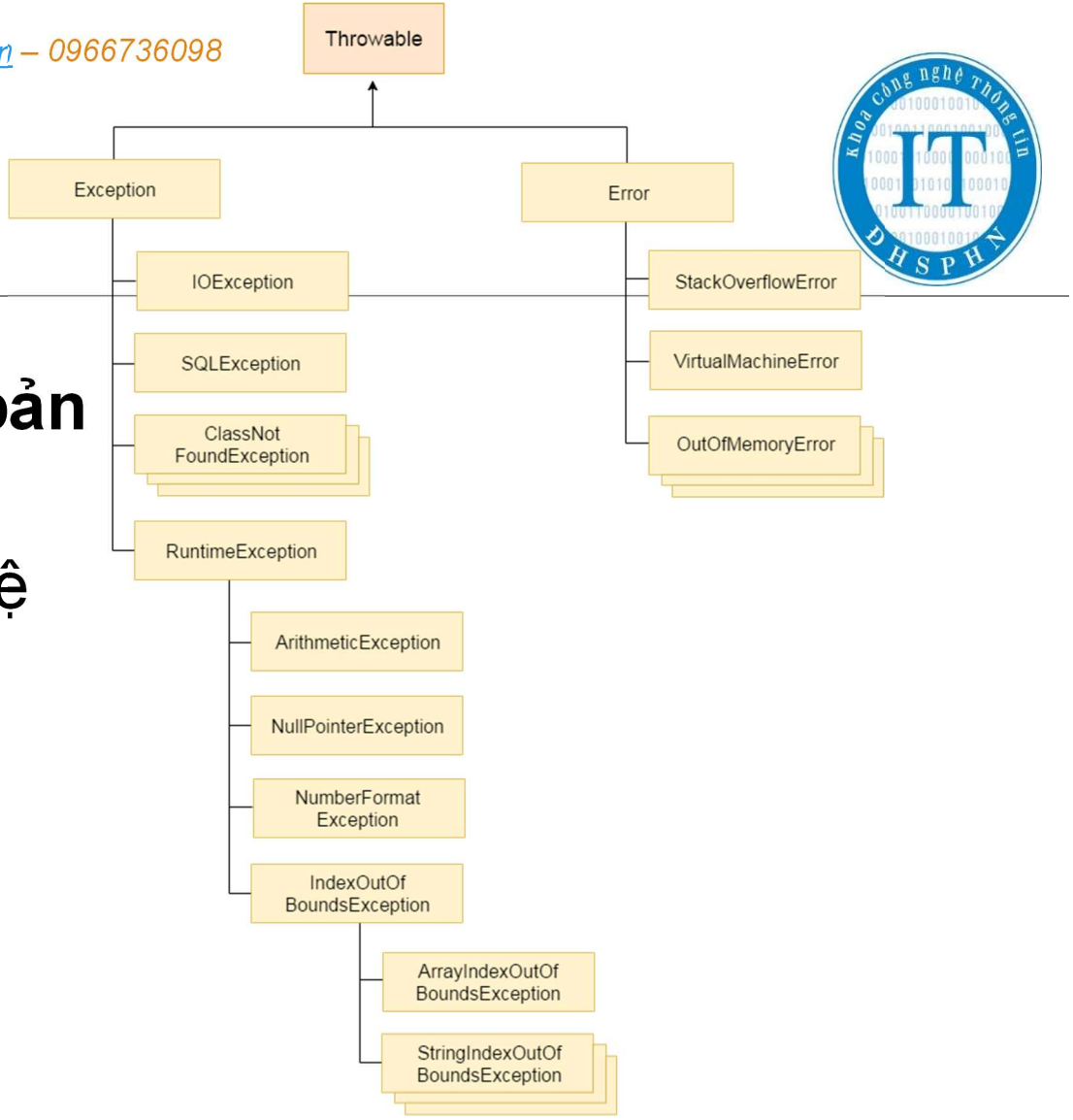
- Exception Handling hay xử lý ngoại lệ là một cơ chế mạnh mẽ giúp xử lý các lỗi runtime để có thể duy trì luồng bình thường của ứng dụng.

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

I. Xử lý ngoại lệ

1.1 Khái niệm cơ bản

- Hệ thống cấp bậc của các lớp ngoại lệ trong Java



CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

I. Xử lý ngoại lệ

1.1 Khái niệm cơ bản

❖Các kiểu của ngoại lệ

- Checked Exception
- Unchecked Exception
- Error

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

I. Xử lý ngoại lệ

1.1 Khái niệm cơ bản

- Các từ khóa xử lý ngoại lệ trong java
  1. try
  2. catch
  3. finally
  4. throw
  5. throws

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

I. Xử lý ngoại lệ

1.2 Khối lệnh try…catch

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

I. Xử lý ngoại lệ

1.2 Khối lệnh try…catch

- Khối lệnh try trong java được sử dụng để chứa một đoạn code có thể xảy ra một ngoại lệ.
- Sau khối lệnh **try** bạn phải khai báo khối lệnh **catch** hoặc **finally** hoặc cả hai.

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

I. Xử lý ngoại lệ

1.2 Khối lệnh try…catch

```
try {
    //
    //Khối lệnh có thể xảy ra ngoại lệ hoặc lỗi
    //
} catch (Exception e) {
    //
    //Khối lệnh xử lý khi xảy ra ngoại lệ hoặc lỗi
    //
}
```

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

I. Xử lý ngoại lệ

1.2 Khối lệnh try…catch

```
try {
    //
    //Khối lệnh có thể xảy ra ngoại lệ hoặc lỗi
    //
} finally {
    //
    //Khối lệnh xử lý khi xảy ra ngoại lệ hoặc lỗi
    //
}
```

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

I. Xử lý ngoại lệ

1.2 Khối lệnh try…catch

```
try {
    //
    //Khối lệnh có thể xảy ra ngoại lệ hoặc lỗi
    //
} catch (Exception e) {
    //
    //Khối lệnh xử lý khi xảy ra ngoại lệ hoặc lỗi
    //
} finally {
    //
    //Khối lệnh xử lý cuối cùng sau đó
    //
}
```

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

I. Xử lý ngoại lệ

1.2 Khối lệnh try…catch

```
try {
    //
    //Khối lệnh có thể xảy ra ngoại lệ hoặc lỗi
    //
} catch (ArithmeticException e) {
    //
    //Khối lệnh xử lý khi xảy ra ngoại lệ hoặc lỗi
    //
} catch (ArrayIndexOutOfBoundsException e) {
    //
    //Khối lệnh xử lý khi xảy ra ngoại lệ hoặc lỗi
    //
} catch (Exception e) {
    //
    //Khối lệnh xử lý khi xảy ra ngoại lệ hoặc lỗi
    //
} finally {
    //
    //Khối lệnh xử lý cuối cùng sau đó
    //
}
```

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

I. Xử lý ngoại lệ

1.2 Khối lệnh try…catch

```
ArrayList<String> listNumber = new ArrayList<>();
int sum = 0;
listNumber.add("1");
listNumber.add("2");
listNumber.add(null);
listNumber.add("3");
listNumber.add("4a");
for (int i = 0; i < listNumber.size(); i++) {
    try {
        sum = sum + Integer.parseInt(listNumber.get(i));
    } catch (Exception e) {
        System.err.println("Bi loi tai phan tu thu " + i + ": ");
        e.printStackTrace();
        listNumber.set(i, "-1");
        sum = sum + Integer.parseInt(listNumber.get(i));
    }
}
System.out.println("Sum = " + sum);
```

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

I. Xử lý ngoại lệ

1.3 Từ khóa throw

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

I. Xử lý ngoại lệ

1.3 Từ khóa throw

- Từ khoá **throw** trong java được sử dụng để ném ra một ngoại lệ cụ thể.

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

I. Xử lý ngoại lệ

1.3 Từ khóa throw

```
int age = 16;
if(age < 18) {
    throw new ArithmeticException("Chua du tuoi tham gia...");
}else {
    System.out.println("Welcome...");
}
```

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

I. Xử lý ngoại lệ

1.4 Từ khóa throws

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

I. Xử lý ngoại lệ

1.4 Từ khóa throws

- Từ khóa **throws** trong java được sử dụng để khai báo một ngoại lệ.

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

I. Xử lý ngoại lệ

1.4 Từ khóa throws

```
public static void main(String[] args) throws IOException{
    FileOutputStream fileOutputStream = null;
    String name = "Nguyen Van Anh";
    try {
        fileOutputStream = new FileOutputStream(new File("D:\\data.txt"));
        byte[] nameByte = name.getBytes();
        fileOutputStream.write(nameByte);
    } catch (Exception e) {
        System.err.println("Something went wrong...");
        e.printStackTrace();
    } finally {
        System.out.println("Write to file successfully...");
        fileOutputStream.close();
    }
}
```

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.1 Lớp FileOutputStream

2.2 Lớp FileInputStream

2.3 Lớp FileWriter

2.4 Lớp FileReader

2.5 Lớp Scanner

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.1 Lớp FileOutputStream

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.1 Lớp FileOutputStream

- Dùng để ghi dữ liệu vào file.
- Dữ liệu được ghi theo định dạng byte.
- Package: java.io.FileOutputStream

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.2 Lớp FileOutputStream

Constructor	Description
FileOutputStream(String filePath)	Creates a new file. It gets file name in string.
FileOutputStream(File file)	Creates a new file. It gets file name in File object.

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.1 Lớp FileOutputStream

Method	Description
protected void finalize()	It is used to clean up the connection with the file output stream.
void write(byte[] ary)	It is used to write **ary.length** bytes from the byte array to the file output stream.
FileChannel getChannel()	It is used to return the file channel object associated with the file output stream.
FileDescriptor getFD()	It is used to return the file descriptor associated with the stream.
void close()	It is used to closes the file output stream.

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.1 Lớp FileOutputStream

```
File file = new File("D:\\data.txt");
FileOutputStream fileOutputStream = new FileOutputStream(file);
String name = "Nguyen Van Anh";
byte byteName[] = name.getBytes();
try {
    fileOutputStream.write(byteName);
    System.out.println("Write to " + file.getName() + " successfully...");
} catch (IOException e) {
    // TODO Auto-generated catch block
    System.err.println("Something went wrong!");
    e.printStackTrace();
} finally {
    fileOutputStream.close();
}
```

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.2 Lớp FileInputStream

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.2 Lớp FileInputStream

- Dùng để đọc dữ liệu từ file.
- Thường dùng để đọc dữ liệu dạng byte như: hình ảnh, video, âm thanh, văn bản, …
- Package: java.io.FileInputStream

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.2 Lớp FileInputStream

Constructor	Description
FileInputStream(String filePath)	It gets filename in string. It opens the given file in read mode. If file doesn't exist, it throws FileNotFoundException.
FileInputStream(File file)	It gets filename in file instance. It opens the given file in read mode. If file doesn't exist, it throws FileNotFoundException.

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.2 Lớp FileInputStream

Method	Description
int available()	It is used to return the estimated number of bytes that can be read from the input stream.
int read()	It is used to read the byte of data from the input stream.
long skip(long x)	It is used to skip over and discards x bytes of data from the input stream.
FileChannel getChannel()	It is used to return the unique FileChannel object associated with the file input stream.
FileDescriptor getFD()	It is used to return the FileDescriptor object.
protected void finalize()	It is used to ensure that the close method is call when there is no more reference to the file input stream.
void close()	It is used to closes the stream.

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.2 Lớp FileInputStream

```
File file = new File("D:\\data.txt");
FileInputStream fileInputStream = new FileInputStream(file);
String name = "";
int key;
try {
    while((key = fileInputStream.read()) != -1) {
        name = name + (char)key;
    }
} catch (IOException e) {
    // TODO Auto-generated catch block
    System.err.println("Something went wrong!");
    e.printStackTrace();
} finally {
    fileInputStream.close();
}
System.out.println("Name = " + name);
```

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.3 Lớp FileWriter

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.3 Lớp FileWriter

- Dùng để ghi dữ liệu vào file.
- Thường dùng để ghi dữ liệu dạng chuỗi hoặc ký tự vào file.
- Package: java.io.FileWriter

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.3 Lớp FileWriter

Constructor	Description
FileWriter(String filePath)	Creates a new file. It gets file name in string.
FileWriter(File file)	Creates a new file. It gets file name in File object.

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.3 Lớp FileWriter

Method	Description
void write(String text)	It is used to write the string into FileWriter.
void write(char c)	It is used to write the char into FileWriter.
void write(char[] c)	It is used to write char array into FileWriter.
void flush()	It is used to flushes the data of FileWriter.
void close()	It is used to close the FileWriter.

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.3 Lớp FileWriter

```
File file = new File("D:\\data.txt");
FileWriter fileWriter = new FileWriter(file);
String name = "Nguyen Van Anh";
try {
    fileWriter.write(name);
    System.out.println("Write to " + file.getAbsolutePath() + " successfully...");
} catch (IOException e) {
    // TODO Auto-generated catch block
    System.err.println("Something went wrong!");
    e.printStackTrace();
} finally {
    fileWriter.close();
}
```

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.4 Lớp FileReader

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.4 Lớp FileReader

- Dùng để đọc dữ liệu từ file.
- Thường dùng để đọc dữ liệu dạng chuỗi hoặc ký tự từ file.
- Package: java.io.FileReader

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.4 Lớp FileReader

Constructor	Description
FileReader(String filePath)	It gets filename in string. It opens the given file in read mode. If file doesn't exist, it throws FileNotFoundException.
FileReader(File file)	It gets filename in file instance. It opens the given file in read mode. If file doesn't exist, it throws FileNotFoundException.

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.4 Lớp FileReader

Method	Description
int read()	It is used to return a character in ASCII form. It returns -1 at the end of file.
void close()	It is used to close the FileReader class.

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.4 Lớp FileReader

```
File file = new File("D:\\data.txt");
FileReader fileReader = new FileReader(file);
String name = "";
int key;
try {
    while((key = fileReader.read()) != -1){
        name = name + (char)key;
    }
} catch (Exception e) {
    // TODO Auto-generated catch block
    System.err.println("Something went wrong!");
    e.printStackTrace();
} finally {
    fileReader.close();
}
System.out.println("Name = " + name);
```

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.5 Lớp Scanner

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.5 Lớp Scanner

- Dùng để đọc dữ liệu từ file.
- Thường dùng để đọc dữ liệu dạng số, chuỗi hoặc ký tự từ file.
- Package: java.util.Scanner

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

II. Đọc-Ghi file

2.5 Lớp Scanner

```
File file = new File("D:\\data.txt");
Scanner sc = new Scanner(file);
String name = "";
try {
    name = sc.nextLine();
} catch (Exception e) {
    // TODO Auto-generated catch block
    System.err.println("Something went wrong!");
    e.printStackTrace();
} finally {
    sc.close();
}
System.out.println("Name = " + name);
```

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

III. Serializable Interface

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

III. Serializable Interface

3.1 Khái niệm cơ bản

3.2 Từ khóa transient

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

III. Serializable Interface

3.1 Khái niệm cơ bản

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

III. Serializable Interface

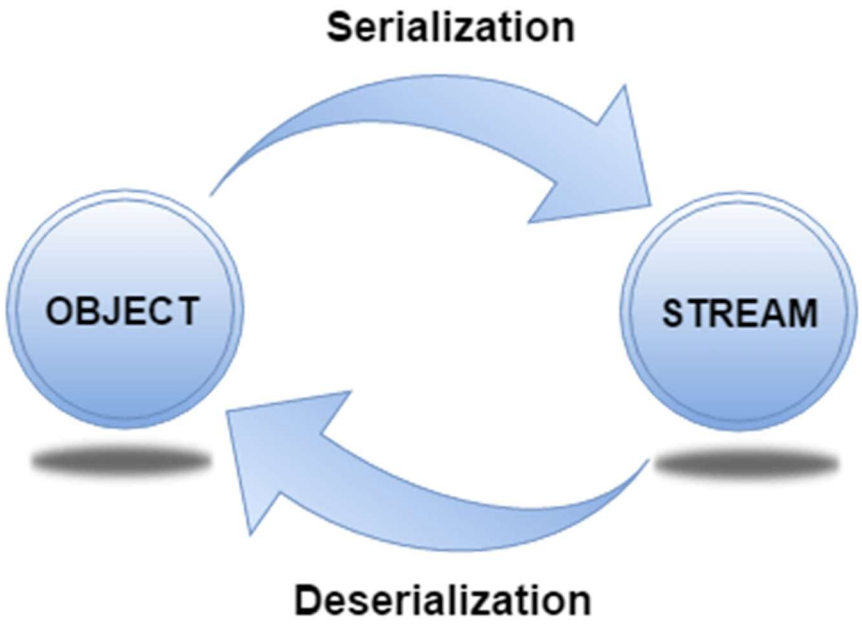
3.1 Khái niệm cơ bản

- Serialization là cơ chế ghi lại trạng thái của object thành các byte dữ liệu (Byte stream).
- Thường dùng trong Hibernate, JPA,...
- Quá trình ngược lại là Deserialization, chuyển đổi từ các byte dữ liệu thành các object .
- Package: java.io.Serializable

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

III. Serializable Interface

3.1 Khái niệm cơ bản



CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

III. Serializable Interface

3.1 Khái niệm cơ bản

```
1  import java.io.Serializable;
2
3  public class Student implements Serializable {
4      int id;
5      String name;
6
7      public Student(int id, String name) {
8          this.id = id;
9          this.name = name;
10     }
11 }
```

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

III. Serializable Interface

3.2 Từ khóa transient

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

III. Serializable Interface

3.2 Từ khóa transient

- Trong quá trình Đọc-Ghi đối tượng, nếu muốn loại bỏ thuộc tính nào thì ta phải khai báo transient cho thuộc tính đó.

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

III. Serializable Interface

3.2 Từ khóa transient

```
1  import java.io.Serializable;
2
3  public class Student implements Serializable {
4      int id;
5      String name;
6      transient int age;
7
8      public Student(int id, String name, int age) {
9          this.id = id;
10         this.name = name;
11         this.age = age;
12     }
13 }
```

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

IV. Đọc-Ghi Object vào file

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

IV. Đọc-Ghi Object vào file

4.1 Lớp ObjectOutputStream

4.2 Lớp ObjectInputStream

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

IV. Đọc-Ghi Object vào file

4.1 Lớp ObjectOutputStream

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

IV. Đọc-Ghi Object vào file

4.1 Lớp ObjectOutputStream

- Sử dụng để ghi các kiểu dữ liệu nguyên thuỷ hoặc các đối tượng Java vào một OutputStream.
- Chỉ có các đối tượng thuộc lớp implements Serializable Interface mới có thể được ghi vào stream.

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

IV. Đọc-Ghi Object vào file

4.1 Lớp ObjectOutputStream

Constructor	Mô tả
public ObjectOutputStream(OutputStream out)	Tạo một ObjectOutputStream ghi vào OutputStream được chỉ định.

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

IV. Đọc-Ghi Object vào file

4.1 Lớp ObjectOutputStream

Phương thức	Mô tả
public final void writeObject(Object obj)	Ghi một đối tượng được chỉ định tới ObjectOutputStream.
public void flush()	Làm sạch ObjectInputStream hiện tại.
public void close()	Đóng ObjectInputStream hiện tại.

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

IV. Đọc-Ghi Object vào file

4.1 Lớp ObjectOutputStream

```
File file = new File("D:\\data.txt");
FileOutputStream fileOutputStream = new FileOutputStream(file);
ObjectOutputStream objectOutputStream = new ObjectOutputStream(fileOutputStream);
ArrayList<Student> listStudents = new ArrayList<>();
listStudents.add(new Student("SV001", "CNTT", 3.0, "Nguyen Van Anh", true, 20));
listStudents.add(new Student("SV002", "TOANTIN", 3.95, "Vu Thu Trang", true, 21));
listStudents.add(new Student("SV003", "VATLI", 3.23, "Tran Ngoc Nam", false, 22));
listStudents.add(new Student("SV004", "HOAHOC", 3.37, "Ngo Minh Hang", true, 20));
listStudents.add(new Student("SV005", "TIENGANH", 3.45, "Dang Tuan Anh", false, 23));
try {
    objectOutputStream.writeObject(listStudents);
    System.out.println("Write list student to \"" + file.getAbsolutePath() + "\" successfully...");
} catch (IOException e) {
    // TODO Auto-generated catch block
    System.out.println("Something went wrong!");
    e.printStackTrace();
} finally {
    objectOutputStream.close();
    fileOutputStream.close();
}
```

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

IV. Đọc-Ghi Object vào file

4.2 Lớp ObjectInputStream

IV. Đọc-Ghi Object vào file

4.2 Lớp ObjectInputStream

- Sử dụng để đọc các kiểu dữ liệu nguyên thuỷ hoặc các đối tượng Java đã được ghi bằng ObjectOutputStream.
- Chỉ có các đối tượng thuộc lớp implements Serializable Interface mới có thể Deserialization để đọc được dữ liệu.

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

IV. Đọc-Ghi Object vào file

4.2 Lớp ObjectInputStream

Constructor	Mô tả
public ObjectInputStream(InputStream in)	Tạo ra một ObjectInputStream đọc từ InputStream đã chỉ định.

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

IV. Đọc-Ghi Object vào file

4.2 Lớp ObjectInputStream

Phương thức	Mô tả
public final Object readObject()	Đọc một đối tượng từ input stream.
public void close()	Đóng ObjectInputStream hiện tại.

CN.TRẦN HẢI LONG – longth@hnue.edu.vn – 0966736098

IV. Đọc-Ghi Object vào file

4.2 Lớp ObjectInputStream

```
File file = new File("D:\\data.txt");
FileInputStream fileInputStream = new FileInputStream(file);
ObjectInputStream objectInputStream = new ObjectInputStream(fileInputStream);
ArrayList<Student> listStudents = new ArrayList<>();
try {
    listStudents = (ArrayList<Student>) objectInputStream.readObject();
} catch (Exception e) {
    // TODO Auto-generated catch block
    System.out.println("Something went wrong!");
    e.printStackTrace();
} finally {
    objectInputStream.close();
    fileInputStream.close();
}
for(int i = 0; i < listStudents.size(); i++) {
    listStudents.get(i).display();
}
```


THANKS FOR WATCHING