

PHIẾU THỰC HÀNH SỐ 1

Nội dung: I/O theo luồng và thao tác tệp cơ bản

Bài 1:

- Trao đổi với sinh viên về các nội dung phiếu bài tập số 1 và phiếu bài tập số 2.
- Tiếp tục đề sinh viên rà soát lại các bài thực hành theo 2 phiếu bài tập để củng cố kiến thức (2 tiết)

Bài 2: Lập trình I/O với các lớp `ObjectInputStream/ObjectOutputStream` sử dụng cơ chế tuần tự hóa.

Yêu cầu: Nhập các chương trình ví dụ, chạy chương trình và phân tích từng chương trình; giải thích từng lệnh chương trình sử dụng cú pháp ghi chú với mỗi dòng lệnh.

a) Lớp `ObjectOutputStream`

- File `Student.java`

```
import java.io.Serializable;
public class Student implements Serializable {
    private static final long serialVersionUID = 1L;
    private int id;
    private String name;
    private String address;
    private int age;

    public void Studet() {}

    public Student(int id, String name, String address, int age) {
        super();
        this.id = id;
        this.name = name;
        this.address = address;
        this.age = age;
    }

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }
}
```

```

    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public String getAddress() {
        return address;
    }
    public void setAddress(String address) {
        this.address = address;
    }
    public int getAge() {
        return age;
    }
    public void setAge(int age) {
        this.age = age;
    }
    public String toString() {
        return "Student@[id=" + id + " , name=" + name + " , "
            + "address= " + address + ",age =" + age+ "]";
    }
}

```

- File ObjectOutputStreamExample.java

```

import java.io.FileOutputStream;
import java.io.IOException;
import java.io.ObjectOutputStream;

public class ObjectOutputStreamExample {
    public static void main(String[] args) throws IOException {
        ObjectOutputStream oos = null;
        try {
            oos = new ObjectOutputStream(new FileOutputStream("D:\\Data.txt"));
            // create student
            Student student = new Student(1, "Tran Hao Phong", "Ha Noi", 17);
            // write student
            oos.writeObject(student);
            System.out.println("Success...");
        } catch (IOException ex) {
            ex.printStackTrace();
        } finally {
            oos.close();
        }
    }
}

```

b) Lớp *ObjectInputStream*

- Lớp Student.java như ý a)
- Lớp ObjectInputStreamExample.java

```
import java.io.FileInputStream;
import java.io.IOException;
import java.io.ObjectInputStream;
public class ObjectInputStreamExample {
    public static void main(String[] args) throws IOException {
        ObjectInputStream ois = null;

        try {
            ois = new ObjectInputStream(new FileInputStream("D:\\Data.txt"));
            // read student
            Student student = (Student) ois.readObject();
            // show student
            System.out.println(student.toString());
        } catch (ClassNotFoundException ex) {
            ex.printStackTrace();
        } catch (IOException ex) {
            ex.printStackTrace();
        } finally {
            ois.close();
        }
    }
}
```

Bài tập 3: Viết chương trình nhập một danh sách sinh viên và ghi vào tệp Data.dat sử dụng lớp RandomAccessFile.

Mục đích: Hiểu và biết sử dụng thành thạo lớp và các phương thức của lớp RandomAccessFile.

Yêu cầu:

- Nhập chương trình, chạy và khảo sát kỹ lớp RandomAccessFile; sử dụng thành thạo kỹ thuật truy xuất tệp ngẫu nhiên với RandomAccessFile.
- Viết một ứng dụng thực tế với lớp RandomAccessFile.

a) Bài mẫu

```
import java.io.*;
```

```
class RandomAccessFileExample {
```

```

public static void main(String args[]) throws IOException {
    double data[] = { 11.2, 13.6, 255.6, 117.92, 2007.96, 8.9, 9.9, 10.0, 100.6 };
    double d;
    RandomAccessFile raf;

    try {
        raf = new RandomAccessFile("E:\\random.dat", "rw");
    } catch (FileNotFoundException exc) {
        System.out.println("Cannot open file.");
        return;
    }

    // Write values to the file.
    for (int i = 0; i < data.length; i++) {
        try {
            raf.writeDouble(data[i]);

        } catch (IOException exc) {
            System.out.println("Error writing to file.");
            return;
        }
    }

    try { // Now, read back specific values
        raf.seek(0 * 8); // seek to first double
        d = raf.readDouble();
        System.out.println("First value is " + d);
        raf.seek(8 * 1); // seek to second double
        d = raf.readDouble();
        System.out.println("Second value is " + d);
        raf.seek(8 * 3); // seek to fourth double
        d = raf.readDouble();
        System.out.println("Fourth value is " + d);
        System.out.println();
    }

    //Read All data
    System.out.println("Read all: ");
    for (int i = 0; i < data.length; i++) {
        raf.seek(8 * i); // seek to ith double
        d = raf.readDouble();
        System.out.print(d + " ");
    }
    System.out.println("");

    // Now, read every other value.
    System.out.println("Here is every other value: ");
    for (int i = 0; i < data.length; i += 2) {
        raf.seek(8 * i); // seek to ith double
    }
}

```

```
        d = raf.readDouble();
        System.out.print(d + " ");
    }

    System.out.println("\n");
} catch (IOException exc) {
    System.out.println("Error seeking or reading.");
}

raf.close();
}
```

b)Viết chương trình quản lý sinh viên sử dụng phương pháp truy xuất ngẫu nhiên với lớp RandomAccessFile.

-----Hết-----