1- What happen if we change the text “Welcome” to “Welcome To Newtork Programming”?

It output just “Welcome” and throw an exception: “Exception in thread "main" java.util.InputMismatchException”.

2- What is I/O redirection in java? Give a code example.

Redirecting in that form just replaces stdin with a file stream coming from that file

```
Scanner scanner = new Scanner(System.in);
while (scanner.hasNext())
 {
   System.out.println(scanner.next());
 }
```

3- Compare between serial access and random-access files in terms of their definition, usage, advantages, disadvantages.

Type	Serial Access	Random-Access
Definition	Serial files are stored in chronological order, that is as each record is received it is stored in the next available storage position	A random-access data file enables you to read or write information anywhere in the file.
Usage	Both used for both reading and writing files	
Advantages	Simple to handle. The space on the storage medium can be utilized to the maximum possible extent. Widely used in small-scale applications.	• Immediate access to record is possible. • Up-to-date information will always be available on the file. • Addition & deletion of record is not very complex.

Disadvantages	We can't go directly to a specific record. It is not possible to add or modify records within an existing file.	• Records in a particular file must be of the Same length. • Given string field must be of the same length for all records on the file. • Numeric data is not in human-readable form

4- What is I/O redirection in java? Give a code example.

Java provides a mechanism, called object serialization where an object can be represented as a sequence of bytes that includes the object's data as well as information about the object's type and the types of data stored in the object. After a serialized object has been written into a file, it can be read from the file and deserialized that is, the type information and bytes that represent the object and its data can be used to recreate the object in memory.

Example:

```
import java.io.*;

public class SerializeDemo {

public static void main(String [] args) {

Employee employee = new Employee();

employee.name = "Ahmed Hassan";

employee.address = "Egypt, Luxor, Luxor";

employee.SSN = 123;

employee.number = 25;

    try {
```

```
            FileOutputStream file =
            new FileOutputStream("/folder/employee.txt");
            ObjectOutputStream cout = new ObjectOutputStream(file);
            cout.writeObject(employee);
            cout.close();
            file.close();
            System.out.printf("Serialized data is saved in: /folder /employee.txt");
        } catch (IOException i) {
            i.printStackTrace();
        }
    }
}
```

5- What is JFileChooser in java with code example

JFileChooser: is a quick and easy way to prompt the user to choose a file or a file saving location.

JFileChooser has 6 constructors:

- JFileChooser(): empty constructor that points to user's default directory
- JFileChooser(String): uses the given path
- JFileChooser(File): uses the given File as the path
- JFileChooser(FileSystemView): uses the given FileSystemView
- JFileChooser(String, FileSystemView): uses the given path and the FileSystemView
- JFileChooser(File, FileSystemView): uses the given current directory and the FileSystemView

Example:

```
package com.mkyong.jfileChooser;

import java.io.File;

import javax.swing.JFileChooser;

import javax.swing.filechooser.FileSystemView;

public class FileChooser1 {

	public static void main(String[] args) {

		JFileChooser JF = new
JFileChooser(FileSystemView.getFileSystemView().getHomeDirectory());

		int returnValue = JF.showOpenDialog(null);

		if (returnValue == JFileChooser.APPROVE_OPTION)

		}

			File file = JF.getSelectedFile();

			System.out.println(file.getAbsolutePath());

		}

	}

}
```