**Pseudo-code:**  
1. Start the program

2. Declare a string variable `filePath` and set its value to "example.txt"

3. Write to the file "example.txt" using a try-catch block:

a. Open a `FileWriter` object and pass the `filePath` as parameter

b. Call the `write()` method of the `FileWriter` object with "Hello, world!\n" as parameter to write to the file

c. Close the `FileWriter` object

d. Print "Successfully wrote to the file."

e. If an `IOException` is caught, print "An error occurred while writing to the file." and print the stack trace

4. Append to the file "example.txt" using a try-catch block:

a. Open a `FileWriter` object and pass the `filePath` and `true` as parameters to enable append mode

b. Call the `write()` method of the `FileWriter` object with "This is an appended message.\n" as parameter to append to the file

c. Close the `FileWriter` object

d. Print "Successfully appended to the file."

e. If an `IOException` is caught, print "An error occurred while appending to the file." and print the stack trace

5. Read the file "example.txt" using a try-catch block:

a. Open a `FileReader` object and pass the `filePath` as parameter

b. Create a `BufferedReader` object by passing the `FileReader` object as parameter to read the file line by line

c. Declare a string variable `line`

d. While there are still lines to read from the file:

I. Read the next line of the file using the `readLine()` method of the `BufferedReader` object and assign it to `line`

II. If `line` is not null, print `line` to the console

e. Close the `FileReader` and `BufferedReader` objects

f. Print "Successfully read the file."

g. If an `IOException` is caught, print "An error occurred while reading the file." and print the stack trace

6. End the program