

Write a Java program to create a method that reads a file and throws an exception if the file is not found.

Sample Solution:

Java Code:

```
import java.io.File;
import java.io.FileNotFoundException;
import java.util.Scanner;

public class File_Read {
    public static void main(String[] args) {
        try {
            readFile("test1.txt");
        } catch (FileNotFoundException e) {
            System.out.println("Error: " + e.getMessage());
        }
    }

    public static void readFile(String fileName) throws FileNotFoundException {
        File file = new File(fileName);
        Scanner scanner = new Scanner(file);

        // Read and process the contents of the file
        while (scanner.hasNextLine()) {
            String line = scanner.nextLine();
            System.out.println(line);
        }

        scanner.close();
    }
}
```

Sample Output:

```
Error: test1.txt (The system cannot find the file specified)
```

Explanation:

In the above exercise,

- In this program, we have a method called `readFile` that takes a `fileName` parameter. To read the contents of the file, it creates a `Scanner` object using the `File` class.
- In the main method, we call the `readFile` method and provide the name of the file we want to read. If the file is not found, a `FileNotFoundException` is thrown.
- In the `readFile` method, we declare a `File` object and initialize it with the given `fileName`. We then create a `Scanner` object using the file. If the file is not found, a `FileNotFoundException` is thrown.
- A try-catch block is used in the main method to catch the `FileNotFoundException` and print an error message.

Flowchart: