**Java Reading and Writing to Files:**

**Reading from a file:**

*// Exception to method signature***public static void** readOnlyFirstLine1() **throws** IOException {  
  
 *// FileReader to open a file and BufferedReader to read first only* BufferedReader br = **new** BufferedReader(**new** FileReader(**"./sam3.csv"**));  
 String s = br.readLine();  
 System.***out***.println(s);  
 br.close();  
}  
  
*// Exception with try-catch block***public static void** readOnlyFirstLine2() {  
 BufferedReader br = **null**;  
 **try** {  
 br = **new** BufferedReader(**new** FileReader(**"./sam3.csv"**));  
 } **catch** (FileNotFoundException e) {  
 e.printStackTrace();  
 }  
 String s = **null**;  
 **try** {  
 **if** (br != **null**) {  
 s = br.readLine();  
 }  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 System.***out***.println(s);  
 **try** {  
 br.close();  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
}  
  
*// Exception try-with-resources***public static void** readFirstLine3(String pathToFile) **throws** IOException {  
  
 **try**(BufferedReader br=**new** BufferedReader(**new** FileReader(pathToFile))) {  
 String s=br.readLine();  
 System.***out***.println(s);  
 }  
}  
  
*// Simple Read from file***public static void** readAllLines() **throws** IOException {  
  
 FileReader fr=**new** FileReader(**"./sam3.csv"**);  
 BufferedReader br=**new** BufferedReader(fr);  
 String line;  
 **while**((line=br.readLine())!=**null**){  
 System.***out***.println(line);  
 }  
 br.close();  
}  
  
*// Exception to method signature***public static void** readAllLine1() **throws** FileNotFoundException, IOException {  
  
 *// throws FileNotFoundException* BufferedReader br = **new** BufferedReader(**new** FileReader(**"./sam3.csv"**));  
  
 String rows = **""**;  
 *// throws IOException* **while** ((rows = br.readLine()) != **null**) {  
 String[] s = rows.split(**","**);  
 System.***out***.println(s[0] + **" : "** + s[1]);  
 }  
 *// throws IOException* br.close();  
}  
  
*// Exception with try-catch blocks***public static void** readAllLines2() **throws** IOException {  
  
 BufferedReader br = **null**;  
 **try** {  
 br = **new** BufferedReader(**new** FileReader(**"./sam3.csv"**));  
 } **catch** (FileNotFoundException e) {  
 e.printStackTrace();  
 }  
 String rows = **""**;  
 **while** (**true**) {  
 **try** {  
 **if** (!((rows = br.readLine()) != **null**)) **break**;  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 String[] s = rows.split(**","**);  
 System.***out***.println(s[0] + **" : "** + s[1]);  
 }  
  
 br.close();  
}

**Reading file using Java 8:**

*// Read file using Java 8 Path and Stream***public static void** readFile() **throws** IOException {  
 Path filePath= Paths.*get*(**"./mytext.txt"**);  
 Stream<String> lines= Files.*lines*(filePath);  
 lines.forEach(System.***out***::println);  
}  
  
*// Read file using try-with-resource and Java 8 path and stream***public static void** readFile1() **throws** IOException {  
 Path filePath=Paths.*get*(**"./mytext.txt"**);  
 **try** (Stream<String> lines = Files.*lines*(filePath)) {  
 lines.forEach(System.***out***::println);  
 }  
}  
  
*// Read file using Java 8 filter stream of lines***public static void** readFile2() **throws** IOException {  
 Path filePath=Paths.*get*(**"./mytext.txt"**);  
 **try** (Stream<String> lines = Files.*lines*(filePath)) {  
 List<String> list=lines  
 .filter(s-> s.contains(**"Hello World"**))  
 .collect(Collectors.*toList*());  
 lines.forEach(System.***out***::println);  
 }  
}

**Writing to a file:**

*// Write to a file -single line with method exception***public static void** writeSingleLine() **throws** IOException {

String s=**"Hello World"**;  
 FileWriter fr=**new** FileWriter(**"./mytext.txt"**);  
 BufferedWriter bw=**new** BufferedWriter(fr);  
 bw.write(s);  
 bw.flush();  
  
}

*// Write to a file – multiple line*

**public static void** writeLines() **throws** IOException {

String s=**"Hello World"**+System.*getProperty*(**"line.separator"**);  
 BufferedWriter bw=**new** BufferedWriter(**new** FileWriter(**"./mytext.txt"**));  
 **for**(**int** i=0;i<10;i++){  
 bw.write(s + System.*getProperty*(**"line.separator"**) );  
 }  
 bw.flush();  
  
}

**Writing to a file using Java 8**:

*// Simple write file using Java 8 Path* **public static void** writeToFile() **throws** IOException {  
 String str=**"Hello World"**;  
  
 Path filePath= Paths.*get*(**"./"**,**"demo.csv"**);  
 BufferedWriter br=Files.*newBufferedWriter*(filePath);  
 br.write(str);  
 br.flush();  
 }  
*// Exception try-with-resources - Java 8* **public static void** writeToFile1(){  
 String str=**"Hello World Java 8"**;  
  
 Path filePath= Paths.*get*(**"./"**,**"demo.csv"**);  
 **try** (BufferedWriter br = Files.*newBufferedWriter*(filePath)) {  
 br.write(str);  
 br.flush();  
 } **catch** (IOException e) {  
 e.printStackTrace();  
 }  
 }