1. File Handling?
2. File Handling:  
   🡪File Handling Implies how to read from and write to file in java. Java provides the basic I/O package for reading and writing streams. Java.io package allows to do all input and output tasks in java.
3. Stream in Java:  
   Stream is a sequence of data.  
   It can be of two types:  
   1. Byte Stream – Mainly incorporates with byte data when an input and output process happens with the byte data then it is called the File Handling process with byte stream.  
   2. Character Stream – A Stream which incorporates with characters when an input and output process happens with the character then it is called as File Handling process with Character Stream.
4. Java File Methods:  
   a. canRead() – Tests whether file is readable or not  
   b. canWrite() – Tests whether the file is writable or not  
   c. createNewFile() – Creates an empty file  
   d. Delete() – Deletes a file  
   e. Exists() – Tests whether the file exists or not  
   f. getName() – Returns the name of the file  
   g. getAbsolutePath() – Returns the absolute pathname of the file  
   h. Length() – Returns the size of the file in bytes  
   i. List() – Returns the array of the file in the directory  
   j. Mkdir() – Creates a directory
5. Creating a new File:

**package** FileHandling;

**import** java.io.File;

**import** java.io.IOException;

**public** **class** CreateFile {

**public** **static** **void** main(String[] args) {

File myObj = **new** File("C:\\Users\\ADMIN\\eclipse-workspace\\Java Strings\\FileHandling\\Data");

**try** {

**if**(myObj.createNewFile()) {

System.***out***.println("File Created " + myObj.getName());

} **else** {

System.***out***.println("File already exists.");

}

} **catch** (IOException e) {

System.***out***.println("An error occured.");

e.printStackTrace();

}

}

}

Output:  
File Created Data  
Note: If we execute another time it shows an output:  
File already exists.

1. To get File Information:

**package** FileHandling;

**import** java.io.File;

**public** **class** FileInformation {

**public** **static** **void** main(String[] args) {

File myObj = **new** File("C:\\Users\\ADMIN\\eclipse-workspace\\Java Strings\\FileHandling\\Data");

**if**(myObj.exists()) {

System.***out***.println("File Name :" + myObj.getName());

System.***out***.println("Absolute Path :" + myObj.getAbsolutePath());

System.***out***.println("Writable :" + myObj.canWrite());

System.***out***.println("Readable :" + myObj.canRead());

System.***out***.println("File size in Bytes :" + myObj.length());

} **else** {

System.***out***.println("The File does not exists.");

}

}

}

Output:  
File Name :Data

Absolute Path :C:\Users\ADMIN\eclipse-workspace\Java Strings\FileHandling\Data

Writable :true

Readable :true

File size in Bytes :0

Note: If we write any thing in the file, then the file size changes.

1. How to write into a File:

**package** FileHandling;

**import** java.io.File;

**import** java.io.FileWriter;

**import** java.io.IOException;

**public** **class** WriteToFile {

**public** **static** **void** main(String[] args) {

**try** {

FileWriter myWriter = **new** FileWriter("C:\\Users\\ADMIN\\eclipse-workspace\\Java Strings\\FileHandling\\Data");

myWriter.write("Java is the prominent programming language of the millenium");

myWriter.close();

System.***out***.println("Successfully wrote to the File.");

} **catch** (IOException e) {

System.***out***.println("An Error Occured.");

e.printStackTrace();

}

}

}

Output:  
Successfully wrote to the File.

1. Reading Data from File:

**package** FileHandling;

**import** java.io.File;

**import** java.io.FileNotFoundException;

**import** java.util.Scanner;

**public** **class** ReadFromFile {

**public** **static** **void** main(String[] args) {

**try** {

File myObj = **new** File("C:\\Users\\ADMIN\\eclipse-workspace\\Java Strings\\FileHandling\\Data");

Scanner myReader = **new** Scanner(myObj);

**while**(myReader.hasNextLine()) {

String data = myReader.nextLine();

System.***out***.println(data);

}

myReader.close();

} **catch** (FileNotFoundException e) {

System.***out***.println("An Error Occured.");

e.printStackTrace();

}

}

}

Output:  
Java is the prominent programming language of the millenium