

آموزش زبان برنامه نویسی جاوا

(کار با فایل) کلاس OutputStreamWriter

جلسه سی و دوم

نویسنده: رحمان زارعی

جاوا را ساده، آسان و شیرین بنوشید!!!!



مفاهیم کار با فایل ابتدا از جلسه ۳۰ آموزش جاوا جاولایک شروع شده است، پس برای یادگیری این مفاهیم از جلسه ۳۰ شروع کنید! همه جلسات با رعایت اصل پیشنهادی تهیه شده است.

OutputStreamWriter

کلاس **OutputStreamWriter** برای نوشتن داده ها درون فایل استفاده می شود.

کلاس **OutputStreamWriter** پلی از جریان بایت ها به جریان کاراکتر هاست.

برای استفاده از کلاس **OutputStreamWriter** باید پکیج **Java.io.OutputStreamWriter** را در سورس کد برنامه مون import کنیم:

```
import java.io.OutputStreamWriter;
```

- خب این کلاس هم سازنده های زیادی داره که یکی از سازنده های پر کاربرد و پیشفرض آن به صورت زیر است:

```
OutputStreamWriter(OutputStream out)
```

- هنگام شی سازی از کلاس **OutputStreamWriter** با صدا زدن سازنده فوق ، می توانیم از طریق شی ایجاد شده و دسترسی به متدهای مربوطه کلاس اقدام به نوشتن مجموعه ای از داده های کاراکتری در فایل کنیم.
- پارامتر این سازنده شی ای از نوع کلاس **OutputStream** هستش و از طرفی چون کلاس **FileOutputStream** فرزند کلاس **OutputStream** هستش می توانیم از شی کلاس **FileOutputStream** نیز به عنوان پارامتر استفاده کنیم.
- پس هر بار که خواستیم از کلاس **OutputStreamWriter** استفاده کنیم حتما باید در کنار آن از کلاس **OutputStream** نیز شی ایجاد کنیم چون نیاز به پارامتری از نوع کلاس **OutputStream** داریم.

برخی از متدهای پر کاربرد کلاس OutputStreamWriter :

```
void close()
```

بعد از اتمام کار با فایل از طریق این متد فایل مورد نظر را می بندیم.

```
void write(int c)
```

این متد برای نوشتن کاراکتر به کاراکتر داده ها درون فایل کاربرد دارد. پارامتر این متد از نوع یک عدد صحیح می باشد این بدین معناست که ما باید معادل عددی کاراکتر را درون این متد قرار بدیم مثلا بجای کاراکتر a معادل آن یعنی عدد ۹۷ را وارد این متد می کنیم.

مثال:

```
package www.javapro.ir;

import java.io.*;

public class OutputStreamWriterDemo {

    public static void main(String[] args) {

        try {

            // create a new OutputStreamWriter
            OutputStream os = new FileOutputStream("test.txt");
            OutputStreamWriter writer = new OutputStreamWriter(os);
            // write something in the file
            writer.write(70);
            writer.write(71);
            writer.write(72);

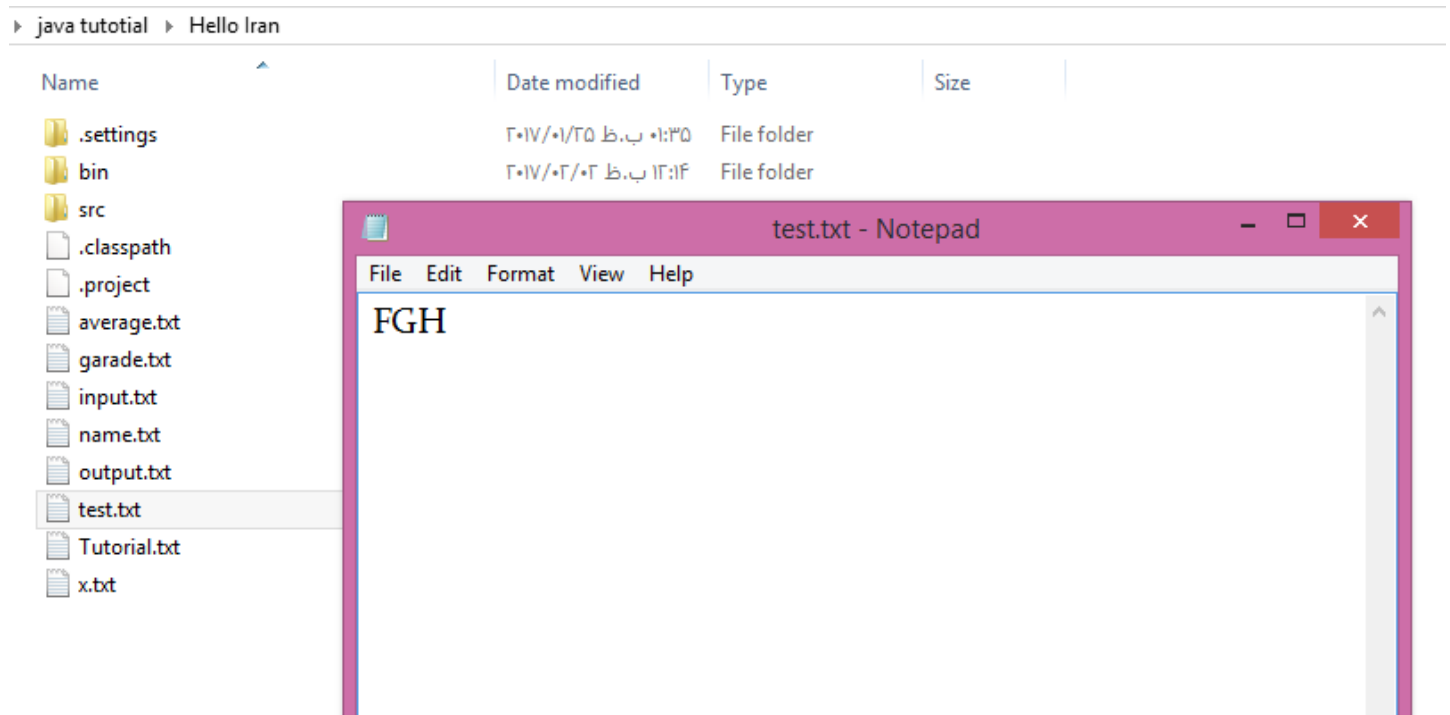
            writer.close();
            // create a new FileInputStream to read what we write
            FileInputStream in = new FileInputStream("test.txt");

            // read what we write
            for (int i = 0; i < 3; i++) {
                System.out.print("" + (char) in.read());
            }
            in.close();
        } catch (Exception ex) {
            ex.printStackTrace();
        }
    }
}
```

خروجی: خروجی این برنامه یک فایل "test.txt" در پوشه پروژه مون می باشد و همچنین داده های این فایل در کنسول نیز بصورت زیر نمایش داده می شود:

FGH

خروجی به صورت فایل "test.txt" را در تصویر (۱) مشاهده می کنید:



تصویر (۱)

توضیحات برنامه:

```
OutputStream os = new FileOutputStream("test.txt");
```

- ایجاد شی از به روش چندریختی از کلاس OutputStream و استفاده از سازنده فرزند این کلاس یعنی OutputStream که بجای پارامتر سازنده آدرس مکانی که قرار است فایل در آن ایجاد شود قرار داده شده است.

```
OutputStreamWriter writer = new OutputStreamWriter(os);
```

- از کلاس OutputStreamWriter شی ساخته و شی ای از نوع کلاس OutputStream را به عنوان پارامتر سازنده آن داده ایم.

```
writer.write(70);
writer.write(71);
writer.write(72);
```

- با استفاده از متد write بایت های خود را درون فایل ذخیره می کنیم. در فایل معادل کاراکتر هر بایت ذخیره می شود. مثلا معادل بایت ۷۰ کاراکتر F در فایل ذخیره می شود.

```
writer.close();
```

- در پایان کار فایل را می بندیم. در این خط کد حتما باید فایل بسته شود چون در خط های بعدی برنامه قصد داریم همین فایل به آدرس "test.txt" را خوانده و داده های آن را در خروجی کنسول چاپ کنیم

```
FileInputStream in = new FileInputStream("test.txt");
```

- برای خواندن فایل از کلاس FileInputStream شی ساخته ایم و آدرس فایل "test.txt" به سازنده آن داده ایم.

```
for (int i = 0; i < 3; i++) {
    System.out.print(" " + (char) in.read());
}
```

- خواندن تک به تک کاراکترهای موجود در فایل

```
in.close();
} catch (Exception ex) {
    ex.printStackTrace();
}
```

- در پایان کار فایل in را می بندیم
- در صورت رخ دادن استثنا دستورات درون بلوک catch اجرا می شود.

void write(String arg)

این متد از پارامتر آن مشخص است که متد بدرد بخور و خوش دستی ست !!! 😊 این متد مستقیم متن یا رشته (String) را درون فایل می نویسد.

مثال :

```
package www.javapro.ir;

import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.OutputStream;
import java.io.OutputStreamWriter;

public class OutputStreamWriterTest {

    public static void main(String[] args) {
        try {
            OutputStream as = new FileOutputStream("Tutorial.txt");
            OutputStreamWriter osw = new OutputStreamWriter(as);
            osw.write("www.javapro.ir");
            osw.close();
        } catch (IOException e) {
```

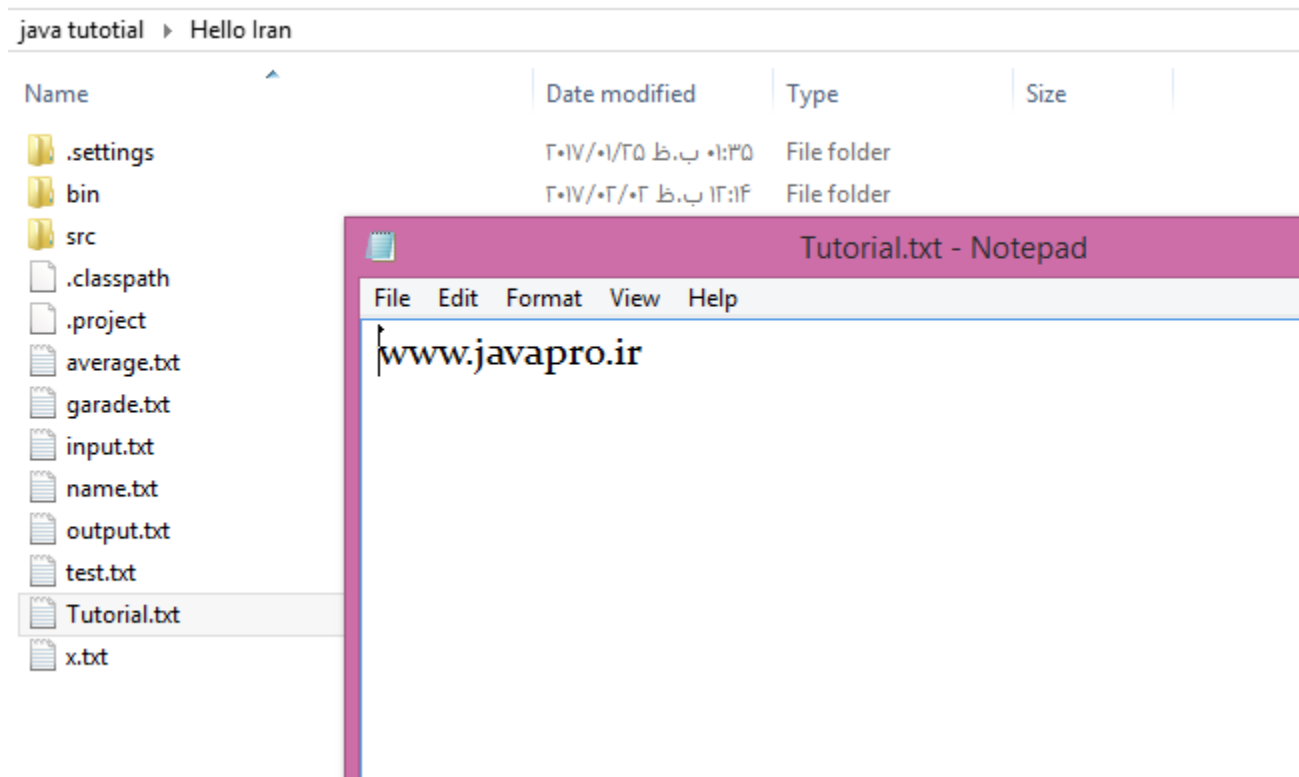
```

        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}
}
}

```

خروجی:

خروجی این برنامه یک فایل با نام و فرمت `Tutorial.txt` در پوشه پروژه می باشد. در تصویر (۲) خروجی این برنامه را مشاهده می کنید:



تصویر (۲)

توضیحات برنامه:

```
OutputStream as = new FileOutputStream("Tutorial.txt");
```

- ایجاد یک شی به روش چندریختی از `OutputStream` و استفاده از سازنده کلاس `FileOutputStream` و قرار دادن آدرس مکانی که قصد داریم فایل در آن ایجاد شود در سازنده این کلاس.

```
OutputStreamWriter osw = new OutputStreamWriter(as);
```

- ایجاد شی از کلاس `OutputStreamWriter` و قرار دادن شی `as` که از نوع کلاس `OutputStream` هستش به عنوان پارامتر

```
osw.write("www.javapro.ir");
```

- با صدا زدن متد `write` و قرار دادن رشته مورد نظر به راحتی متن خود را درون فایل می نویسیم.

```
osw.close();
```

در پایان کار ، فایل مورد نظرمون را با این متد می بندیم.

- خب این بخش از کار با فایل تمام شد در جلسات آینده به سایر کلاس های کار با فایل می پردازیم.
- تمرین فراموش نشه بهترین راه یادگیری کار با فایل حل تمرین از ساده به سخت است.

پیروز و موفق باشید

سایت آموزش زبان جاوا به زبان ساده، آسان و شیرین!!!

www.JAVAPRO.ir

آموزش جاوا SE را با تجربه شخصی و به زبان خودمونی یاد بگیرید!!!!

بازدید از کانال

بازدید از سایت

هر روز مفاهیم و مثال های جدید به سایت اضافه می شود برای اطلاع از مطالب جدید روی سایت عضو کانال شوید.