

# بسم الله الرحمن الرحيم

## آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

متغیرهای Field در جاوا

گردآورنده : مهندس افشین رفوآ

## متغیرهای Field در جاوا

در بخش قبل در مورد متغیرهای داخل متدها صحبت کردیم. متغیرهایی که داخل متدها تنظیم می کنید، تنها در دسترس همان متدها می باشند و در متدهای دیگر قرار نمی گیرند، گفته می شود که دارای دامنه ی داخلی می باشند.

به هر حال می توانید متغیرهایی را در خارج از متدها تنظیم کنید که همه ی متدهای موجود در گروه شما می توانند ببینند. این متغیرها، متغیرهای **Field** (یا متغیرهای **Instance**) نامیده می شوند. می توانید آنها را دقیقاً به روش دیگر متغیرها تنظیم کنید. چهار فیلد زیر را گروه جدید **StudentResult** اضافه کنید.

```
package exams;

public class StudentResults {

    String Full_Name;
    String Exam_Name;
    String Exam_Score;
    String Exam_Grade;

}
```

ما چهار رشته متغیر (چهار رشته فیلد) تنظیم می کنیم. همانطور که نام فیلدها نشان می دهد، رشته حاوی نام یک شخص، نام یک امتحان، یک نمره و یک درجه می شود. این چهار فیلد در دسترس همه ی متدهایی که در این گروه نوشتیم، قرار می گیرند و برای متدها داخلی نیستند. گفته می شود که دارای دامنه ی جهانی می باشند.

برای اینکه چگونگی جهانی بودن آنها را مشاهده کنیم، به گروه **ExamDetails** خود بازگردید، گروهی با متد اصلی (**main**). برای ایجاد یک آبجکت جدید از گروه **StudentResults**، کد زیر را اضافه کنید.

```

package exams;

public class ExamDetails {

    public static void main(String[] args) {

        StudentResults aStudent = new StudentResults();

    }

}

```

این همان کاری می باشد که در بخش قبل انجام دادیم - استفاده از یک لغت کلیدی جدید برای ایجاد یک آبجکت جدید. نام آبجکت **aStudent** خواهد بود و از نوع **StudentResults** می باشد که گروه ماست. در خط بعدی نام متغیر (**aStudent**) را که با یک نقطه (**dot**) دنبال می شود، تایپ کنید. به محض اینکه نقطه را تایپ کردید، **NetBeans** لیستی از متدها و پراپرتی های موجود در آبجکت شما را ارائه می دهد.

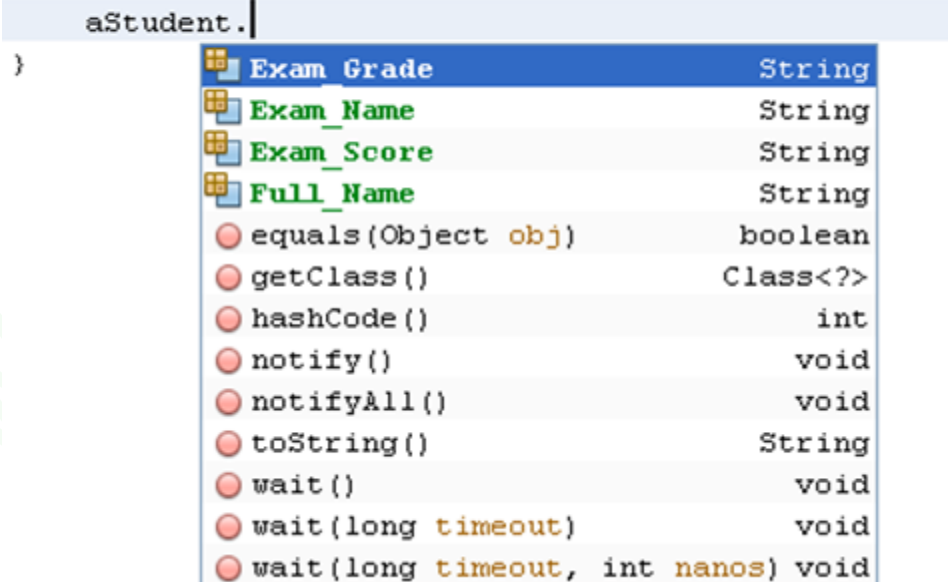
```

public static void main(String[] args) {

    StudentResults aStudent = new StudentResults();

    aStudent.
}

```



|                               |          |
|-------------------------------|----------|
| Exam_Grade                    | String   |
| Exam_Name                     | String   |
| Exam_Score                    | String   |
| Full_Name                     | String   |
| equals(Object obj)            | boolean  |
| getClass()                    | Class<?> |
| hashCode()                    | int      |
| notify()                      | void     |
| notifyAll()                   | void     |
| toString()                    | String   |
| wait()                        | void     |
| wait(long timeout)            | void     |
| wait(long timeout, int nanos) | void     |

چهار فیلدی که تنظیم کردیم، روی لیست قرار دارند. اینها متد نیستند، بلکه چیزی به نام پراپرتی می باشند. این واقعیت که آنها روی لیست هستند، به این معناست که دارای دامنه ی جهانی می باشند. اگر دارای دامنه ی داخلی بودند، روی لیست قرار نمی گرفتند.

شما می توانید مقادیری را برای پراپرتی ها تنظیم کنید. این مورد را امتحان کنید: کد مشخص شده ی زیر را به متد اصلی خود اضافه کنید.

```
public class ExamDetails {  
  
    public static void main(String[] args) {  
  
        StudentResults aStudent = new StudentResults();  
  
        aStudent.Exam_Name = "VB NET";  
  
        String exam = aStudent.Exam_Name;  
        System.out.println( exam );  
    }  
  
}
```

ما فیلد **Exam\_Name** را از لیست انتخاب کرده ایم و به آن مقدار **"VB NET"** اختصاص داده ایم. سپس خط بعدی مقدار **Exam\_Name** را از آبجکت **aStudent** می گیرد. نتیجه در متغیری به نام امتحان (**exam**) ذخیره شده و سپس چاپ می شود. وقتی برنامه را اجرا می کنید، خروجی رشته ی **"VB Net"** می باشد.

بنابراین چهار متغیری که تنظیم کرده ایم، در هر دو گروه در دسترس می باشند.

به هر حال جهانی ساختن متغیرهای فیلد، مانند این مورد، فکر خوبی نیست. شما تمایل به از دست دادن پیگیری مقادیری را دارید که در این متغیرها می باشند، بنابراین اشکال زدایی کد شما بسیار سخت تر می شود. محدود کردن دامنه ی متغیرهای فیلد، برنامه نویسی خوبی تلقی می شود.

برای اینکه یک متغیر فیلد تنها در دسترس یک گروه خاص باشد، درست قبل از اعلام فیلد لغت کلیدی **private** را وارد کنید. کد را در گروه **StudentResults** مانند زیر تغییر دهید.

```

package exams;

public class StudentResults {

    private String Full_Name;
    private String Exam_Name;
    private String Exam_Score;
    private String Exam_Grade;

}

```

اکنون تنها کد موجود در گروه **StudentResults** می تواند این متغیرها را ببیند. برای بررسی به عقب، به متد اصلی خود بازگردید. باید چند اخطار و خطوط قرمز مشاهده کنید.

```

public static void main(String[] args) {

    StudentResults aStudent = new StudentResults();
    Exam_Name has private access in exams.StudentResults

    aStudent.Exam Name = "VB NET";

    String exam = aStudent.Exam Name;
    System.out.println( exam );

}

```

سه خط را در پایین حذف کنید. **aStudent** را و سپس یک نقطه تایپ کنید تا لیست **NetBeans** را مشاهده کنید.

aStudent.

|                                 |          |
|---------------------------------|----------|
| ● equals(Object obj)            | boolean  |
| ● getClass()                    | Class<?> |
| ● hashCode()                    | int      |
| ● notify()                      | void     |
| ● notifyAll()                   | void     |
| ● toString()                    | String   |
| ● wait()                        | void     |
| ● wait(long timeout)            | void     |
| ● wait(long timeout, int nanos) | void     |

همانطور که مشاهده می کنید، اکنون چهار متغیر فیلد محو شده اند. علت حذف آنها این است که دیگر دامنه ی جهانی ندارند و بنابراین از گروه ExamDetails قابل مشاهده نیستند.

در بخش بعد در مورد چیزی به نام **class constructor** خواهید آموخت.

www.tahlildadeh.com