

## به نام خدا

### آموزش بیست و هفتم

پس از آشنایی به انواع Loop ها در قسمت های پیشین، در این آموزش قصد داریم تا به بررسی Array ها در زبان برنامه نویسی جاوا بپردازیم (در این آموزش از ترجمه کردن واژه Array خودداری می کنیم چرا که برای درک بهتر این مسئله، بهتر است آن را تحت عنوان Array بشناسیم). یکی از ساده ترین راه های ذخیره اطلاعات در زبان های برنامه نویسی استفاده از Variable ها است. حال فرض کنیم که تعدادی Variable داریم که از یک جنس یکسان هستند مثلاً همگی از جنس int می باشند. حال به جای آنکه به تعداد مواردی از Variable ها که می خواهیم در برنامه خود مورد استفاده قرار دهیم یک متغیر جدید تعریف می کنیم، به سادگی می توانیم کلیه متغیرهای خود را در یک Array قرار دهیم. به عبارت دیگر یک Array به گروهی از متغیرها گفته می شود که دارای جنسی یکسان هستند. به طور خلاصه می توان Array ها را به شکل یک Variable دید که قادرند بیش از یک Value را در خود اختصاص دهند.

حال فرض کنیم که می خواهیم یک برنامه دفترچه تلفن بنویسیم که در آن بتوانیم به راحتی نام و شماره تلفن دوستان و آشنایان را در برنامه خود ذخیره کنیم (در این برنامه قصد داریم، برای کوتاه شدن توضیحات، فرض کنیم که ما فقط سه دوست داریم). حال اگر بخواهیم نام و شماره تلفن این سه دوست خود را در متغیرها ذخیره سازیم، مجبور خواهیم بود سه متغیر از جنس int برای ذخیره سازی شماره تلفن ها و همچنین سه کلاس از جنس String برای ذخیره سازی نام دوستان خود تعریف کنیم. اما با استفاده از Array ها خواهیم توانست هر سه شماره تلفن را در یک Array از جنس int و هر سه نام را در Array دیگری از جنس String ذخیره ساخته و کد خود را کوتاه تر و تمیز تر سازیم.

برای این منظور پروژه ای جدید در محیط برنامه نویسی اکلیپس تحت عنوان Session 27<sup>th</sup> به معنی "جلسه بیست و هفتم" ایجاد کرده و یک کلاس به نام Array در آن ایجاد می کنیم. همانطور که قبلاً توضیح داده شد، Comment های اضافی را از درون کد خود حذف می کنیم

دوره آموزش جاوا

کلیه حقوق متعلق به وب سایت نردبان است.

مدرس: بهزاد مرادی

و شروع به برنامه نویسی می کنیم. همانطور که قبلاً توضیح داده شد، برای ایجاد یک متغیر ما نیاز داریم تا نوع متغیر را نوشته، سپس نام در نظر گرفته شده برای آن و پس از قرار دادن علامت مساوی یک مقدار برای آن در نظر می گیریم و در نهایت یک علامت ; قرار می دهیم. به عبارت دیگر متغیر ما به شکل `int number = 12` خواهد بود. نحوه ایجاد Array ها هم تا حدودی به همین نحو است اما با اعمال تغییراتی کوچک. به طور مثال، برای تبدیل یک متغیر به یک Array می بایست پس از نوع متغیر که در مثال فوق `int` است علامت `[]` را قرار دهیم. در واقع این علامت به کامپیوتر این دستور را صادر می کند که با یک Array طرف است. در نهایت برای اینکه یک Array کامل داشته باشیم، می بایست به نحو زیر عمل کنیم:

```
public class Array {
    public static void main(String[] args) {
        int[] number = new int[3];
    }
}
```

در این کد ما یک Array از جنس `int` تحت عنوان `number` به معنی "عدد" ایجاد کرده ایم. حال پس از علامت مساوی از کلید واژه `new` استفاده کرده ایم که به این معنی است که یک Array جدید برای ما ایجاد کن. حال مجدداً نوع متغیر را به همراه علامت `[]` نوشته و در انتها یک علامت ; قرار می دهیم. از آنجا که ما سه دوست بیشتر نداریم، داخل علامت `[]` دوم عدد ۳ را می نویسم. به طور خلاصه این Array سه داده از جنس عدد صحیح را در خود ذخیره خواهد ساخت. اگر خاطرمาน باشد در آموزش پیشین زمانیکه از کلاس `Random` استفاده کردیم به موردی این چنین برخوردیم که در آن کلاس اگر عدد مثلاً ۱۰۰ را به عنوان طیفی از اعداد تعریف کنیم که برنامه از آن میان یک عدد را به طور تصادفی بیرون بکشد، طیف ما از عدد صفر تا عدد نود و نه را شامل می شد. در مورد Array ها هم مسئله به همین صورت است چرا که در Array ها اولین جایگاه برای ذخیره داده به عدد صفر اختصاص داده می شود. حال ما سه عدد دوست داریم که شماره تلفن دوست اول در جایگاه صفر و شماره تلفن دوست دوم در جایگاه یک و در نهایت شماره تلفن دوست سوم در جایگاه دوم ذخیره خواهد شد. این مسئله جزو نکاتی

است که در برنامه نویسی بسیار می بایست مورد توجه قرار گیرد. به طور خلاصه کد تکمیل شده ما به شکل زیر خواهد بود:

```
public class Array {  
    public static void main(String[] args) {  
        int[] number = new int[3];  
        number[0] = 91816122;  
        number[1] = 91217945;  
        number[2] = 91144423;  
    }  
}
```

در کد فوق همانطور که مشاهده می شود به منظور اختصاص دادن Value به مواردی که در Array تعریف شده است، نام Array را به همراه جایگاه آن مثلا [0] نوشته و پس از علامت مساوی مقداری برای آن در نظر می گیریم و در نهایت یک علامت ; قرار می دهیم. حال قصد داریم تا نام سه دوست خود را در یک Array ذخیره سازیم. از آنجا که تعدادی کاراکتر پشت سرهم تشکیل یک string را می دهند و برای ذخیره سازی داده هایی از نوع string می بایست از کلاس String استفاده کنیم، از این رو یک Array دیگر از جنس String ایجاد می کنیم و کد خود را به شکل زیر تکمیل می کنیم:

```
public class Array {  
    public static void main(String[] args) {  
        int[] number = new int[3];  
        number[0] = 91816122;  
        number[1] = 91217945;  
        number[2] = 91144423;  
  
        String[] name = new String[3];  
        name[0] = "Iman";  
        name[1] = "Aria";  
        name[2] = "Mohammad Ali";  
    }  
}
```

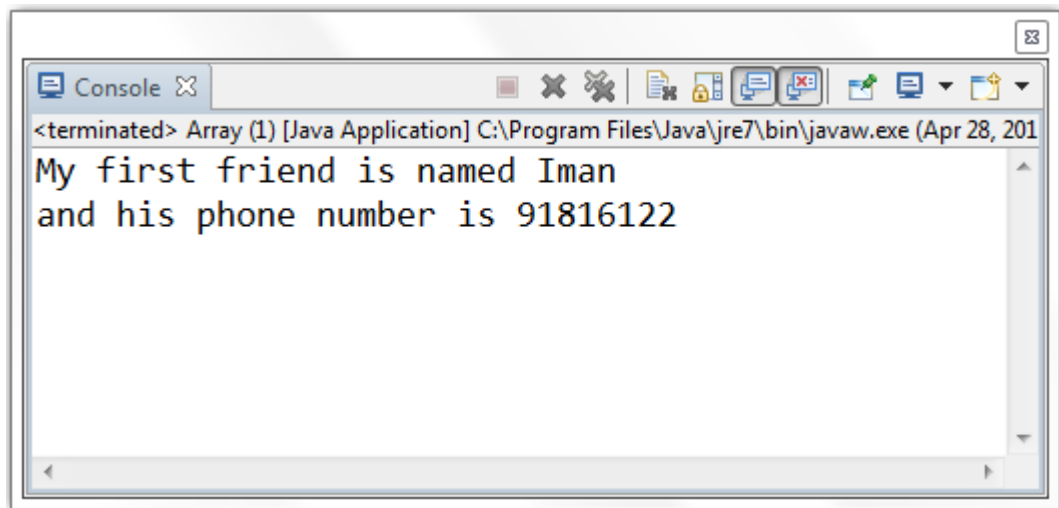
همانطور که ملاحظه می شود، نوع ایجاد یک Array از جنس String نیز همانند int است که در بالای کد ایجاد کردیم فقط تنها تفاوتی که وجود دارد این است که برای ذخیره سازی مقادیر string از علامت “ ” استفاده کرده ایم.

در این مرحله کد ما تکمیل شده است، فقط نیاز است تا اطلاعات را روی صفحه مانیتور به نمایش در آوریم از این رو کد فوق را به شکل زیر تکمیل می کنیم:

```
public class Array {
    public static void main(String[] args) {
        int[] number = new int[3];
        number[0] = 91816122;
        number[1] = 91217945;
        number[2] = 91144423;

        String[] name = new String[3];
        name[0] = "Iman";
        name[1] = "Aria";
        name[2] = "Mohammad Ali";
        System.out.println("My first friend is named " + name[0]
            + " and his phone number is " + number[0]);
    }
}
```

در کد فوق تنها بخشی که اضافه شده است دستور `System.out.println()` است که وظیفه آن نمایش اطلاعات روی صفحه مانیتور است. حال جمله `My first friend is named` به معنی **"نام اولین دوست من هست"** را نوشته سپس با استفاده از علامت + داده ای را که برای اولین جایگاه در Array مربوط به نام دوستان ایجاد کرده بودیم را اضافه می کنیم. سپس یک علامت + دیگر قرار داده و عبارت `and his phone number is` به معنی **"و شماره تلفن او هست"** را به جمله پیشین اضافه می کنیم (همانطور که قبلاً توضیح داده شد، برای ارسال جمله ای به خط بعد از دستور `\n` استفاده می کنیم). حال مجدداً یک علامت + قرار داده و داده مربوط به شماره تلفن اولین دوست را اضافه می کنیم. حال پس از اجرای برنامه خروجی زیر مشاهده خواهد شد:



برای آن که نام و شماره دو دوست دیگر را نیز به نمایش در آوریم، کد خود را به صورت زیر تکمیل می کنیم:

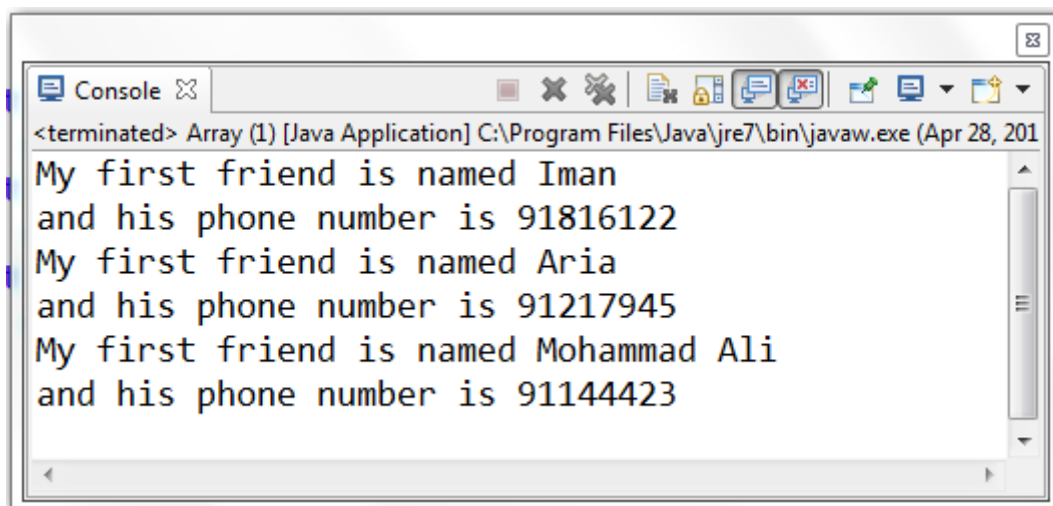
```
public class Array {  
    public static void main(String[] args) {  
        int[] number = new int[3];  
        number[0] = 91816122;  
        number[1] = 91217945;  
        number[2] = 91144423;  
  
        String[] name = new String[3];  
        name[0] = "Iman";  
        name[1] = "Aria";  
        name[2] = "Mohammad Ali";  
        System.out.println("My first friend is named " + name[0]  
            + " \nand his phone number is " + number[0]);  
        System.out.println("My first friend is named " + name[1]  
            + " \nand his phone number is " + number[1]);  
        System.out.println("My first friend is named " + name[2]  
            + " \nand his phone number is " + number[2]);  
    }  
}
```

پس از اجرای مجدد برنامه، خروجی برنامه همانند تصویر زیر خواهد بود:

دوره آموزش جاوا

کلیه حقوق متعلق به وب سایت نردبان است.

مدرس: بهزاد مرادی



The screenshot shows a Java console window titled "Console". The text inside the console is as follows:

```
<terminated> Array (1) [Java Application] C:\Program Files\Java\jre7\bin\javaw.exe (Apr 28, 201  
My first friend is named Iman  
and his phone number is 91816122  
My first friend is named Aria  
and his phone number is 91217945  
My first friend is named Mohammad Ali  
and his phone number is 91144423
```

در آموزش آتی به بحث پیرامون نحوه خلاصه نویسی Array ها، نحوه شمارش تعداد عناصر داخل یک Array و همچنین نحوه به کار گیری Array های چند بعدی بحث خواهیم کرد.