

## به نام خدا

### آموزش بیست و هشتم

همانطور که در آموزش قسمت بیست و هفتم توضیح داده شد، Array ها به منظور ذخیره سازی اطلاعاتی مورد استفاده قرار می گیرند که همگی به یک نوع متغیر اختصاص دارند. حال چنانچه تعداد Element ها یا همان مواردی که در یک Array قرار می گیرند خیلی زیاد نباشند می توان نحوه نوشتن Array را تغییر داد تا کد ما خلاصه تر گردد. در حقیقت روشی که در جلسه آموزشی بیست و هفتم توضیح داده شد برای مواردی کاربرد دارد که تعداد عناصر Array ها بسیار زیاد باشند مثلاً ۲۰۰ مورد.

به منظور خلاصه نویسی Array ها در ابتدا یک پروژه جدید تحت عنوان 28<sup>th</sup> Session به معنی جلسه "بیست و هشتم" ایجاد کرده و کلاسی با StuffToBuy به معنی "چیزهایی که بایستی خریداری شود" می گذاریم. سپس یک Array از جنس String در آن ایجاد می کنیم و نام آن را items به معنی "موارد" می گذاریم. حال کد ما می بایست به صورت زیر باشد:

```
public class StuffToBuy {  
    public static void main(String[] args) {  
        String[] items = {};  
    }  
}
```

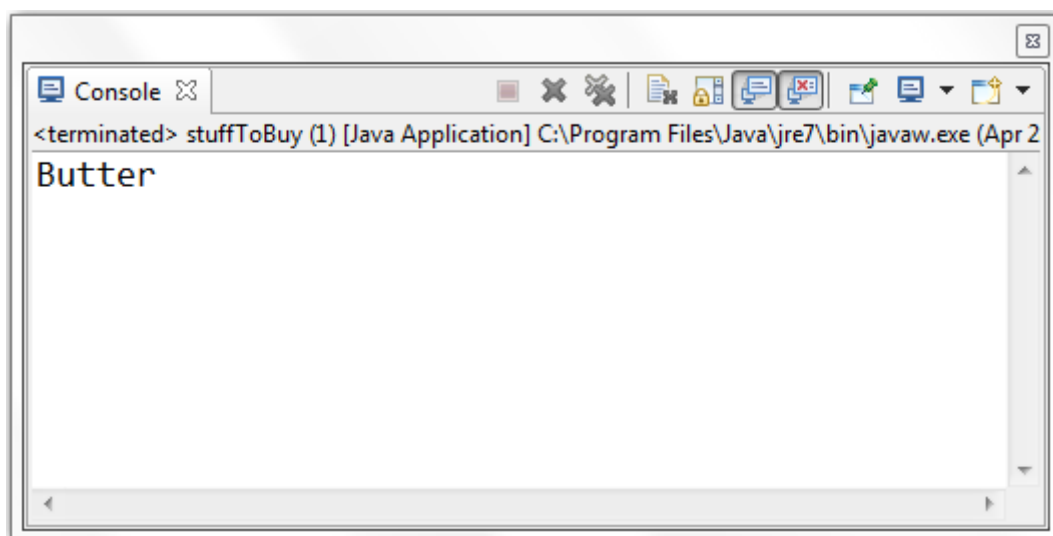
همانطور که ملاحظه می شود در آموزش پیشین نحوه ساخت یک Array به این صورت بود که پس از علامت مساوی از کلید واژه new استفاده کرده سپس مجدداً نوع متغیر را نوشته و در داخل علامت [] تعداد Element هایی را نوشته که می خواهیم داشته باشیم. اما در اینجا از آنجا که می خواهیم عددی را تعریف نکنیم و مستقیماً نام Element ها یا عناصر تشکیل دهنده Array را وارد کنیم از علامت {} پس از مساوی استفاده می کنیم. حال فرض کنیم که نیاز داریم تانان، پنیر، کره، چای، سیب زمینی، گوجه، پیاز، قهوه، دوغ و آب معدنی در لیست خرید ما باشد. برای این منظور کد فوق را به شکل زیر تکمیل می کنیم:

```
public class StuffToBuy {
    public static void main(String[] args) {
        String[] items = {"Bread", "Cheese", "Butter", "Tea", "Potato",
                           "Tomato", "Onion", "Coffe", "Yogurt Drink", "Mineral Water"};
    }
}
```

در واقع از آنجاییکه عناصر تشکل دهنده Array ما از جنس string هستند، از اینرو نیاز است تا این عناصر را داخل علامت “” قرار دهیم و آنها را با یک کاما از یکدیگر جدا سازیم. حال چنانچه بخواهیم نام یکی از این عناصر را روی صفحه نمایش نشان دهیم صرفاً لازم است تا در دستور `System.out.println()` نام Array را نوشته و در داخل علامت [] شماره مورد مد نظر را بنویسیم. به طور مثال در لیست فوق اگر بخواهیم واژه Butter به معنی کره را در پنجره Console به نمایش در آوریم، صرفاً نیاز است تا عبارت `items[2]` را در داخل دستور `System.out.println()` بنویسیم (در واقع از آنجا که شمارش این عناصر از صفر شروع می شود، نیاز است تا عدد ۲ را وارد کنیم و چنانچه عدد ۳ را وارد نماییم، واژه Tea نمایش داده خواهد شد). حال کد تکمیل شده ما به شکل زیر خواهد بود:

```
public class StuffToBuy {
    public static void main(String[] args) {
        String[] items = {"Bread", "Cheese", "Butter", "Tea", "Potato",
                           "Tomato", "Onion", "Coffe", "Yogurt Drink", "Mineral Water"};
        System.out.println(items[2]);
    }
}
```

پس از اجرای برنامه، واژه Butter در پنجره Console به نمایش در خواهد آمد:



دوره آموزش جاوا

کلیه حقوق متعلق به وب سایت نردبان است.

مدرس: بهزاد مرادی

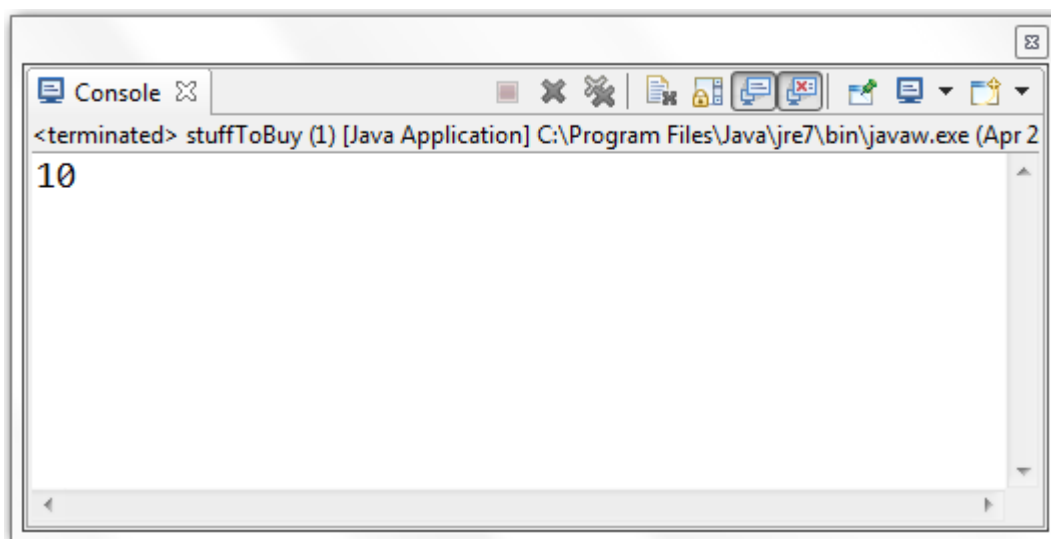
اگر چه که تعداد عناصر ما ده عدد است، اما اگر بخواهیم مورد شماره ده که Mineral Water به معنی آب معدنی است را روی صفحه به نمایش در آوریم، می بایست عدد ۹ را داخل [] وارد نماییم. و این در صوتی است که اگر عدد ۱۰ را وارد نماییم، برنامه در حین Compile شدن از ما ایراد خواهد گرفت و عبارت `ArrayIndexOutOfBoundsException` به معنی اینکه شماره وارد شده خارج ای محدوده تعریف شده است نمایش داده خواهد شد.

## نحوه شمردن تعداد Element ها داخل یک Array

چنانچه بخواهیم تعداد عناصری را که از این طریق وارد برنامه خود کرده ایم را شمارش کنیم، از متدی به اسم `length` استفاده می کنیم. برای این منظور کد فوق را به شکل زیر بازنوسی خواهیم کرد:

```
public class StuffToBuy {  
    public static void main(String[] args) {  
        String[] items = {"Bread", "Cheese", "Butter", "Tea", "Potato",  
                           "Tomato", "Onion", "Coffe", "Yogurt Drink", "Mineral Water"};  
        System.out.println(items.length);  
    }  
}
```

در واقع فقط نیاز است تا در دستور `System.out.println()` نام `Array` را نوشته و متد `length` را ضمیمه آن نماییم. حال پس از اجرای برنامه، تعداد عناصر داخل `Array` ما که معادل با ۱۰ مورد است نمایش داده خواهد شد:



دوره آموزش جاوا

کلیه حقوق متعلق به وب سایت نردبان است.

مدرس: بهزاد مرادی

## نحوه ایجاد Multidimensional Arrays

به طور خلاصه یک Array برای ذخیره سازی داده ها در یک Dimension یا "بعد" استفاده می شوند. حال اگر بخواهیم که یک Array ایجاد کنیم که چند بعدی باشد به طور مثال Array یی که در برگیرنده ستون و ردیف یک جدول باشد، می بایست از Multidimensional Array ها یا Array های چند بعدی استفاده کنیم. مثالی که این گونه Array ها در آن کاربرد فراوانی دارد نمودارهایی با محورهای X و Y می باشد.

برای روشن شدن این مسئله، یک کلاس جدید تحت عنوان MultidimensionalArray ایجاد می کنیم و به شکل زیر آن را تکمیل می کنیم:

```
public class MultidimensionalArray {  
    public static void main(String[] args) {  
        int[] row = {};  
    }  
}
```

در حقیقت این کد نشانگر یک Array تک بعدی یا عادی است. حال اگر بخواهیم آن را به یک Array چند بعدی تبدیل نماییم، می بایست آن را به شکل زیر تغییر دهیم:

```
public class MultidimensionalArray {  
    public static void main(String[] args) {  
        int[][] row = {{},{}};  
    }  
}
```

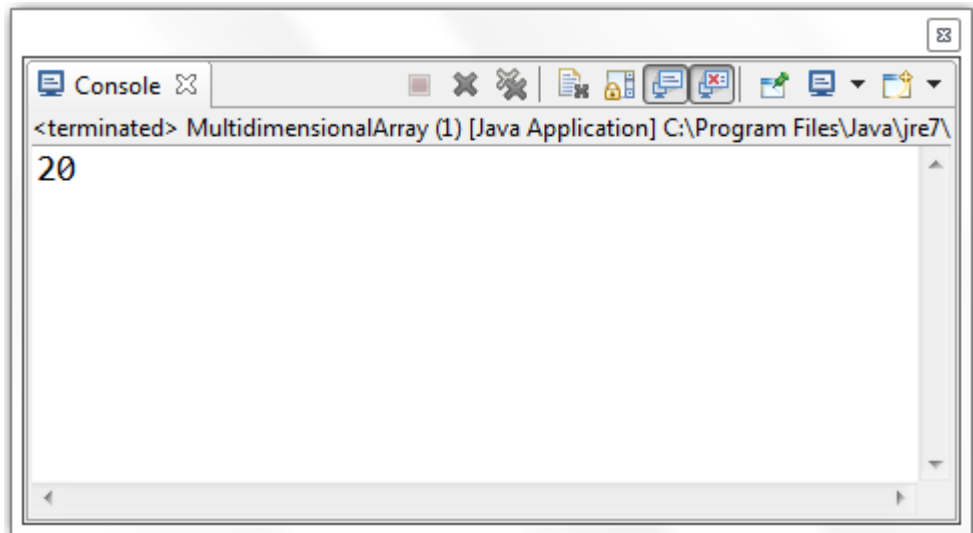
به عبارت دیگر تنها با اضافه کردن یک علامت [] دیگر پس از نوع Array و همچنین اضافه کردن دو جفت علامت {} دیگر داخل علامت {} که پس از علامت مساوی قرار دارد و مجزا کردن آنها از یکدیگر بوسیله یک کاما ما یک Array دو بعدی ایجاد خواهیم کرد. حال کد فوق را به شکل زیر با اعدادی دلخواه تکمیل می کنیم:

```
public class MultidimensionalArray {  
    public static void main(String[] args) {  
        int[][] row = {{10, 20, 30}, {100, 200, 300}};  
    }  
}
```

همانطور که مشاهده می شود، در Array دو بعدی فوق ما دو ردیف داریم که ردیف اول حاوی اعداد ۱۰ و ۲۰ و ۳۰ است و ردیف دوم حاوی اعداد ۱۰۰ و ۲۰۰ و ۳۰۰ است. حال نحوه فراخوانی یک عدد مد نظر از این Array دو بعدی به این شکل است که در کد زیر مشاهده می شود:

```
public class MultidimensionalArray {  
    public static void main(String[] args) {  
        int[][] row = {{10, 20, 30}, {100, 200, 300}};  
        System.out.println(row[0][1]);  
    }  
}
```

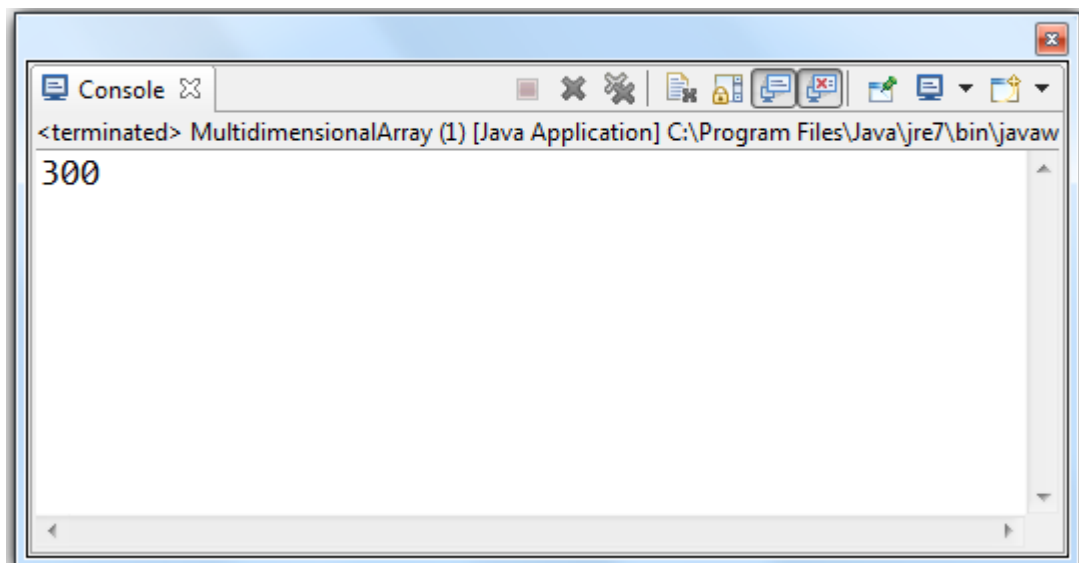
در حقیقت در دستور `System.out.println()` نام Array را نوشته و از آنجا که Array ما دو بعدی است پس دو جفت `[]` مقابل آن قرار می دهیم. همانطور که قبلاً توضیح داده شد شمارش Array ها از صفر آغاز می شود و در این مثال هم ما دو ردیف بیشتر نداریم ردیف اول که `{10,20,30}` است و ردیف دوم که `{100,200,300}` و از آنجا که شمارش در Array ها از صفر آغاز می شود پس عدد اختصاص داده شده به ردیف اول صفر خواهد بود و عدد اختصاص داده شده به ردیف دوم یک است. در ردیف اول ما سه Element یا عنصر داریم که عدد ۱۰ جایگاه صفر را دارا است، عدد ۲۰ جایگاه یک و در نهایت عدد ۳۰ جایگاه دو را دارا است. حال اگر بخواهیم از ردیف اول عدد دوم را در پنجره Console به نمایش در آوریم، صرفاً نیاز است تا در دستور `System.out.println()` پارامتر `row[0][1]` را وارد کنیم. پس از اجرای برنامه خروجی زیر را خواهیم دید:



به نظر می رسد با توضیحات فوق کاملاً این مسئله روشن شده باشد. حال برای آن که تمرین دیگری انجام داده باشیم قصد داریم تا از ردیف دوم عدد آخر که عدد ۳۰۰ است رابه نمایش در آوریم. از آنجا که عدد اختصاص داده شده به ردیف دوم در این Array عدد یک است و از آن جا که عدد اختصاص داده شده به آخرین عدد ردیف دوم عدد دو است، پس کد خود را به شکل زیر بازنویسی می کنیم:

```
public class MultidimensionalArray {
    public static void main(String[] args) {
        int[][] row = {{10, 20, 30}, {100, 200, 300}};
        System.out.println(row[1][2]);
    }
}
```

پس از اجرای برنامه خروجی زیر مشاهده خواهد شد:



دوره آموزش جاوا

کلیه حقوق متعلق به وب سایت نردبان است.

مدرس: بهزاد مرادی

در جلسه آموزشی آینده با نحوه Sort کردن یا مرتب کردن عناصر داخل Array ها آشنا خواهیم شد. لازم به ذکر است به منظور درک بهتر مطالب آموزشی جلسه آتی، آشنایی با مقدمات Array ها و هم چنین آشنایی به نحوه به کارگیری Loop ها ضروری است.