

به نام خدا

آموزش سی و پنجم

اهداف آموزشی این قسمت عبارتند از:

۱. معرفی کلاس های مرتبط با هر یک از Primitive Type ها

۲. معرفی متدهای intValue و parseInt

در آموزش گذشته با نحوه Cast کردن متغیر ها به یکدیگر آشنا شدیم. همانطور که در جلسه سی و چهارم گفته شد Cast کردن به منزله یکی از دستورات کاربردی در برنامه نویسی اپلیکیشن های سیستم عامل اندروید می باشد. از این رو به علاقمندان پیشنهاد می شود که برای درک بهتر Cast در زبان برنامه نویسی جاوا، مطالب آموزشی قسمت های سی و چهارم و سی و پنجم را به خوبی مطالعه کنند.

حال در این جلسه قصد داریم تا ببینیم که به چه نحوی می توان یک شیء را به یک متغیر و بالعکس Cast نمود. پس از شروع نیاز است تا پروژه ای جدید تحت عنوان 35th Session به معنی "جلسه سی و پنجم" ایجاد کرده سپس کلاسی با نام Casting در آن بسازیم.

چیزی که در زبان برنامه نویسی جاوا امکان پذیر نیست، Cast کردن یک Object به یک Variable و یا Cast کردن یک Variable به یک Object است. در عوض در API زبان برنامه نویسی جاوا برای هر یک از انواع متغیر ها یک کلاس با همان نام در نظر گرفته شده که عملیات Cast کردن را برای ما انجام دهد. در واقع در زبان برنامه نویسی جاوا ما هشت نوع متغیر از نوع Primitive Type داریم که عبارتند از boolean, double, float, int, short, long, byte, char و اگر توجه کنیم می بینیم که کلیه این متغیر ها با حروف کوچک شروع می شوند. حال کلاس هایی که در API زبان جاوا برای هر یک از این متغیر های در نظر گرفته شده هم نام با نام متغیر بوده با این تفاوت که با حرف بزرگ نوشته می شوند که عبارتند از Boolean, Integer, Double, Float, Short, Long, Byte, Character (البته دو مورد Integer

دوره آموزش جاوا

کلیه حقوق متعلق به وب سایت نردبان است.

مدرس: بهزاد مرادی

و Character به صورت کامل نوشته می شوند و این در حالی است که نوع متغیر آن ها به ترتیب عبارت است از int و char).

اکنون قصد داریم تا یک Object از جنس کلاس Integer ایجاد کرده و یک Value هم به آن اختصاص دهیم (به حرف اول کلید واژه Integer توجه کنید که به صورت بزرگ نوشته شده است). برای این منظور به صورت زیر عمل می کنیم:

```
public class Casting {  
    public static void main(String[] args) {  
        Integer integerObject = new Integer(123);  
    }  
}
```

همانطور که ملاحظه می شود طرز ساخت یک شیء از روی کلاس Integer به این شکل است که اول نام کلاس مد نظر را نوشته سپس نامی مرتبط را آن را در نظر می گیریم که در این مثال نام integerObject در نظر گرفته شده است. سپس یک علامت مساوی قرار داده و از آنجا که می خواهیم یک شیء جدید از روی این کلاس بسازیم نیاز است تا کلید واژه new را بنویسیم و سپس نام کلاس را مجدد بنویسم. در داخل پرانتزی که مقابل کلید واژه Integer قرار دارد می بایست یک پارامتر تعریف کنیم. به عبارت دیگر این پارامتر همان Value کلاس Integer خواهد بود. حال همانند مابقی Object ها می توانیم از این شیء ساخته شده در هر کجای برنامه که بخواهیم استفاده کنیم. زمانی که بخواهیم از این شیء به عنوان یک متغیر استفاده کنیم، در کلاس Integer متدی تعریف شده است که این کار را برای ما خواهد کرد. اگر بخواهیم یک Value از جنس متغیر int از شیئی که در بالا ساختیم به دست آوریم، می بایست به شکل زیر عمل کنیم:

```
public class Casting {  
    public static void main(String[] args) {  
        Integer integerObject = new Integer(123);  
        int newValue = integerObject.intValue();  
    }  
}
```

در واقع در خطی از کد که اضافه کردیم این دستور را به سیستم می دهیم که یک متغیر از جنس int تحت عنوان newValue به معنی "مقدار جدید" بسازد سپس مقدار این متغیر را مساوی

با نام شیئی که در بالا ساختیم قرار داده به ضمیمه متد `intValue()` که وظیفه اش تبدیل مقدار شیئی بالا به یک متغیر است (لازم به ذکر است که اگر شیئی ساخته شده در مرحله قبل از جنس کلاس `Double` بود می توانستیم از متد `doubleValue()` استفاده کنیم).

یکی از موارد رایج استفاده از تبدیل یک شیئی به یک متغیر زمانی است که می خواهیم مقدار یک کلاس از جنس `String` را به یک متغیر از جنس مثلاً `int` تبدیل کنیم. برای روشن شدن مطلب فوق نیاز است تا یک سناریو تعریف کنیم سپس با توجه به سناریوی موجود یک برنامه بنویسیم. فرض کنیم که می خواهیم برنامه ای طراحی کنیم که در آن کاربر اول نام خود را وارد کند سپس تاریخ تولد خود را بر اساس تاریخ شمسی وارد کند. سپس برنامه سن ایشان را محاسبه کرده و در نهایت جمله ای در پنجره `Console` به نمایش در می آورد مبنی بر اینکه مثلاً آقا یا خانم ایکس شما ۳۰ سال سن دارید. برای شروع این پروژه کار خود را به صورت زیر آغاز می کنیم:

```
import java.util.Scanner;

public class Casting {
    public static void main(String[] args) {
        Scanner keyboardInput = new Scanner(System.in);
        System.out.println("Please enter your name");
    }
}
```

در این کد از کلاس `Scanner` برای دسترسی به ورودی از طریق کیبورد استفاده خواهیم کرد (در آموزش های پیشین به کرات از این کلاس استفاده شده است. برای آشنایی بیشتر با این کلاس به آموزش های یازدهم، دوازدهم و سیزدهم رجوع نمایید). نام کلاس `Scanner` خود را `keyboardInput` به معنی **"ورودی کیبورد"** می گذاریم (لازم به ذکر است برای دسترسی به کلاس `Scanner` می بایست این کلاس را در برنامه خود `Import` یا وارد کنیم بنابراین اولین خط از کد فوق برای همین مسئله نوشته شده است). سپس در دستور `System.out.println()` عبارت `Please enter your name` به معنی **"لطفاً نام خود را وارد کنید"** را نوشته ایم. حال قصد داریم تا این امکان را برای کاربر فراهم کنیم که بتواند نام خود را از طریق کیبورد وارد برنامه کند. برای همین منظور کد فوق را به شکل زیر تکمیل می کنیم:

دوره آموزش جاوا

کلیه حقوق متعلق به وب سایت نردبان است.

مدرس: بهزاد مرادی

```
import java.util.Scanner;

public class Casting {
    public static void main(String[] args) {
        Scanner keyboardInput = new Scanner(System.in);
        System.out.println("Please enter your name");
        System.out.print("My name is ");
        String name = keyboardInput.next();
    }
}
```

سپس در دستور `System.out.print()` عبارت `My name is` به معنی **"نام من هست"** را وارد می کنیم. اگر توجه کرده باشید در این دستور از واژه `println` استفاده نکرده و در عوض از عبارت `print` استفاده کرده ایم چرا که نمی خواهیم برنامه پس از اجرای این دستور به خط بعد برود. سپس یک نمونه از کلاسی از جنس `String` تحت عنوان `name` به معنی **"نام"** ایجاد کرده و مقدار آن را ورودی کیبورد به کلاس `Scanner` قرار می دهیم. برای همین منظور پس از علامت مساوی نام کلاس `Scanner` را نوشته سپس از متد `next()` استفاده می کنیم که وظیفه اش گرفتن ورودی کیبورد است. حال قصد داریم تا کد خود را تکمیل کنیم تا بتوانیم تاریخ تولد را هم از طریق کیبورد بگیریم:

```
import java.util.Scanner;

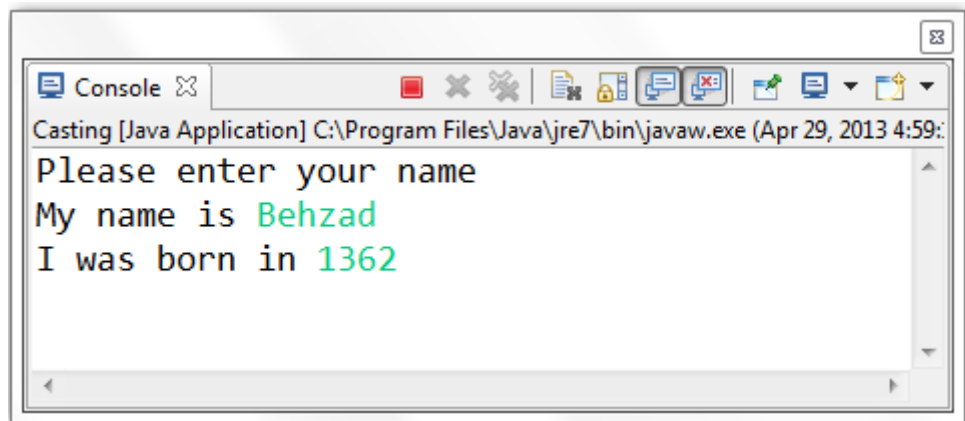
public class Casting {
    public static void main(String[] args) {
        Scanner keyboardInput = new Scanner(System.in);
        System.out.println("Please enter your name");
        System.out.print("My name is ");
        String name = keyboardInput.next();
        System.out.print("I was born in ");
        String date = keyboardInput.next();
    }
}
```

مجدد از دستور `System.out.print()` استفاده می کنیم تا عبارت `I was born in` به معنی **"من در ... به دنیا آمدم"** را به نمایش در آوریم. سپس یک نمونه دیگر از کلاس `String` به اسم `date` به معنی **"تاریخ"** ایجاد می کنیم و مقدار آن را همانند `name` قرار می دهیم. حال اگر برنامه را اجرا کنیم با تصویر زیر مواجه خواهیم شد:

دوره آموزش جاوا

کلیه حقوق متعلق به وب سایت نردبان است.

مدرس: بهزاد مرادی



تا اینجای کار برنامه ما به درستی کار می کند. حال می خواهیم مقدار کلاس String تحت عنوان date را گرفته و از سال ۱۳۹۲ کم کنیم تا ببینیم فردی به اسم بهزاد چند ساله است. نکته ای که در اینجا می بایست مد نظر قرار داد این است که مقدار date اگر چه که ۱۳۶۲ است اما این در حالی است که جنس آن همانند مقدار name چیزی از جنس کلمه است و ما به هیچ وجه نمی توانیم آن را از عدد ۱۳۹۲ کم کنیم. برای این منظور نیاز است تا از Cast کردن date که یک String است به یک متغیر از جنس int که در این قسمت از آموزش فرا خواهیم گرفت استفاده کنیم. برای این منظور کد خود را به شکل زیر تکمیل می کنیم:

```
import java.util.Scanner;

public class Casting {
    public static void main(String[] args) {
        Scanner keyboardInput = new Scanner(System.in);
        System.out.println("Please enter your name");
        System.out.print("My name is ");
        String name = keyboardInput.next();
        System.out.print("I was born in ");
        String date = keyboardInput.next();
        int dateValueToInt = Integer.parseInt(date);
    }
}
```

در این کد تکمیل شده ما یک متغیر از جنس int تحت عنوان dateValueToInt به معنی

"تبدیل مقدار date به متغیری از جنس int" ایجاد کرده و مقدار آن را

Integer.parseInt(date) قرار داده ایم. در واقع کاری که ما در این خط از کد انجام خواهیم

داد این است که مقدار متغیر خود را مساوی با کلاسی تحت عنوان Integer قرار داده سپس

متدی تحت عنوان parseInt() را فرا می خوانیم تا به ما کمک کند تا با قرار دادن پارامتری

دوره آموزش جاوا

کلیه حقوق متعلق به وب سایت نردبان است.

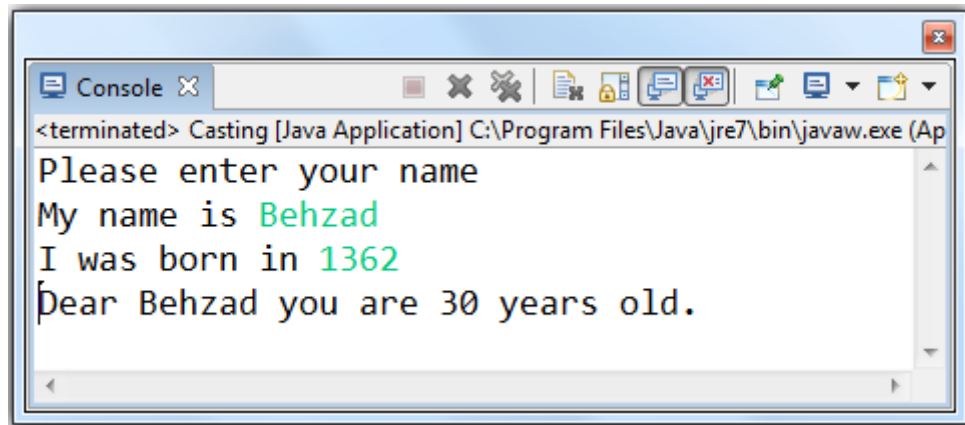
مدرس: بهزاد مرادی

داخل پرانتز آن، آن را به یک داده از جنس `int` تبدیل کند. برای همین منظور نام شیئی که برای گرفتن تاریخ تولد مورد استفاده قرار گرفت را که همان `date` است را داخل پرانتز وارد می کنیم. تا اینجای کار توانسته ایم با موفقیت تاریخ تولد کاربر را گرفته و آن را از یک `string` یا "کلمه" به یک `int` یا "عدد" صحیح تبدیل کنیم. حال نیاز داریم تا آن را از عدد ۱۳۹۲ کم کرده و نتیجه را روی صفحه مانیتور نمایش دهیم. برای این منظور کد خود را به شکل زیر تکمیل می کنیم:

```
import java.util.Scanner;

public class Casting {
    public static void main(String[] args) {
        Scanner keyboardInput = new Scanner(System.in);
        System.out.println("Please enter your name");
        System.out.print("My name is ");
        String name = keyboardInput.next();
        System.out.print("I was born in ");
        String date = keyboardInput.next();
        int dateValueToInt = Integer.parseInt(date);
        int yourAge = 1392 - dateValueToInt;
        System.out.println("Dear " + name + " you are " + yourAge
            + " years old.");
    }
}
```

در کد فوق پس از تکمیل یک متغیر از جنس `int` به اسم `yourAge` به معنی "سن شما" ایجاد کرده ایم و مقدار آن را برابر با حاصل تفریق عدد ۱۳۹۲ از مقدار متغیر `dateValueToInt` قرار داده ایم. سپس در دستور `System.out.println()` که پس از آن قرار گرفته است کلمه `Dear` به معنی "عزیز" را به علاوه `name` می کنیم که در برگیرنده نام کاربر خواهد بود. سپس عبارت `you are` به معنی "شما هستید" را به علاوه نام متغیری می کنیم که قرار است تاریخ تولد کاربر را از عدد ۱۳۹۲ کسر کند که همان `yourAge` است. در نهایت عبارت `years old` را به انتهای عبارت خود اضافه می کنیم. پس از اجرای برنامه خروجی زیر مشاهده خواهد شد:



```
<terminated> Casting [Java Application] C:\Program Files\Java\jre7\bin\javaw.exe (Ap
Please enter your name
My name is Behzad
I was born in 1362
Dear Behzad you are 30 years old.
```

در جلسه سی و ششم پیرامون Access Control یا به عبارتی "کنترل کردن سطح دسترسی" به کلاس ها، متغیرها و متدها در زبان برنامه نویسی جاوا صحبت خواهیم کرد. پس از مطالعه این آموزش انتظار می رود بتوانیم به سؤالات زیر پاسخ بدهیم:

۱. کلاس های مرتبط با هر یک از Primitive Type ها کدامند؟
۲. وظیفه متدهای intValue و doubleValue چیست؟
۳. وظیفه متد parseInt چیست؟