

به نام خدا

آموزش چهل و نهم

اهداف آموزشی این قسمت عبارتند از:

۱. معرفی کلید واژه final در زبان برنامه نویسی جاوا

۲. کاربردی های مختلف کلید واژه final

پس از آشنایی با کلید واژه static در دو آموزش پیشین، در این آموزش قصد داریم تا کلید واژه final را مورد بررسی قرار دهیم. به طور کلی می توان گفت که از کلید واژه final می توان در سه موقعیت متفاوت بهره مند شد.

مورد اول استفاده از این کلید واژه برای کلاس ها است. در واقع زمانی که یک کلاس را final کنیم دیگر امکان ایجاد Subclass از آن کلاس وجود نخواهد داشت. به عبارت دیگر با استفاده از final یک از مهم ترین قابلیت های زبان برنامه نویسی جاوا که یک زبان شیء گرا است را از دست خواهیم داد و آن Inheritance است (برای کسب اطلاعات بیشتر پیرامون مبحث شیء گرایی به آموزش سوم مراجعه نمایید).

مورد دوم استفاده از کلید واژه final به همراه متدها است. در واقع زمانی که یک متد final شود دیگر به هیچ وجه امکان بازنویسی آن متد برای ما وجود نخواهد داشت. به عبارت دیگر نوع کاری که آن متد انجام می دهد را نمی توانیم تغییر دهیم.

مورد سوم استفاده از کلید واژه final در ارتباط با Local Variable ها، Instance Variable ها و پارامترها است. در حقیقت زمانی که final را در کنار یک متغیری بنویسیم که دارای یک مقدار اولیه است، مقدار اولیه آن متغیر نهایی شده و دیگر به هیچ وجه امکان اختصاص یک مقدار جدید به آن را نخواهیم داشت.

حال به منظور درک بهتر مطالب تئوری فوق پروژه ای تحت عنوان 49th Session ایجاد کرده و کلاسی در آن تحت عنوان WhatIsFinal به معنی "کلید واژه final چیست؟" ایجاد می کنیم.

دوره آموزش جاوا

کلیه حقوق متعلق به وب سایت نردبان است.

مدرس: بهزاد مرادی

به خاطر داشته باشیم که در حین ساخت این کلاس گزینه `public static void main` را تیک
نزنیم چرا که این کلاس به منزله کلاس اجرایی ما نخواهد بود:

```
public class WhatIsFinal {  
  
}
```

حال می بایست متدی در این کلاس ایجاد کنیم که وظیفه آن به نمایش در آوردن نام، نام خانوادگی
و تاریخ تولد ما باشد. برای این منظور کد فوق را به شکل زیر تکمیل می کنیم:

```
public class WhatIsFinal {  
  
    public void showData() {  
        String name = "Behzad";  
        String lastName = "Moradi";  
        int dateOfBirth = 1362;  
        System.out.println(name + "\n" + lastName + "\n" +  
dateOfBirth);  
    }  
}
```

همانطور که در کد فوق ملاحظه می شود متدی تحت عنوان `showData` به معنی "اطلاعات را
نشان بده" ایجاد کرده و دو کلاس `String` تحت عناوین `name` و `lastName` به ترتیب به
معنی "نام" و "نام خانوادگی" در آن ایجاد کرده ایم و در نهایت یک متغیر از جنس `int` تحت
عنوان `dateOfBirth` به معنی "تاریخ تولد" اضافه نموده ایم. اکنون برای اختصاص مقادیر به
آنها می توانیم اطلاعات شخصی خود را در نظر بگیریم. سپس به منظور نمایش دادن این
اطلاعات روی صفحه مانیتور از دستور `System.out.println` استفاده کرده به این صورت که
اول `name` را نوشته سپس دستور `\n` را می نویسیم (کاری که این دستور انجام می دهد این است
که ادامه اطلاعات را به خط بعد منتقل می کند). به همین منوال مابقی اسامی را هم داخل دستور
`System.out.println` قرار می دهیم.

حال کلاس دیگری تحت عنوان `ActionClass` به معنی "کلاس عملیاتی" ایجاد کرده و در
حین ساخت آن گزینه `public static void main` را تیک دار می کنیم چرا که این کلاس
قرار است به عنوان نقطه شروع برنامه ما باشد:

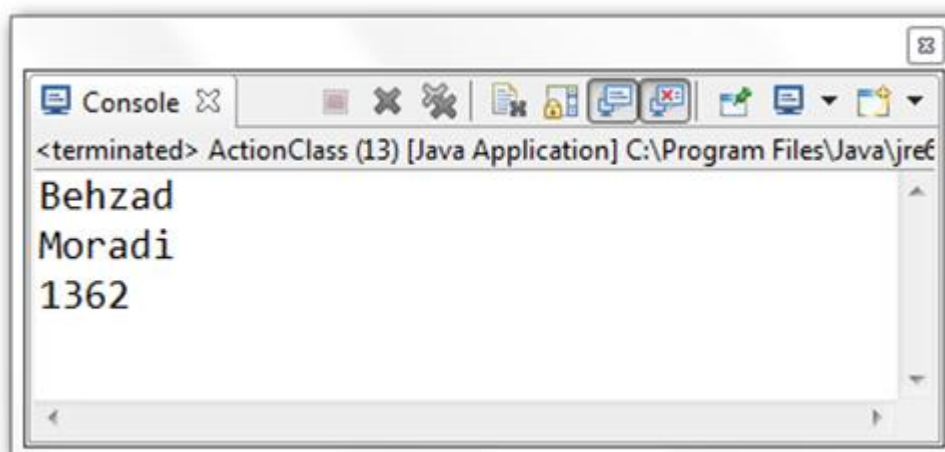
```
public class ActionClass {
    public static void main(String[] args) {

    }
}
```

در این مرحله نیاز است تا شیئی از روی کلاس WhatIsFinal تحت عنوان objectOne به معنی "شیئی اول" ایجاد کنیم و سپس با استفاده از آن شیئی متد موجود در کلاس را فرا بخوانیم. برای این منظور کد فوق را به شکل زیر تکمیل می کنیم:

```
public class ActionClass {
    public static void main(String[] args) {
        WhatIsFinal objectOne = new WhatIsFinal();
        objectOne.showData();
    }
}
```

اکنون در صورتیکه برنامه خود را اجرا کنیم با خروجی زیر مواجه خواهیم شد:



می بینیم که متد showData که وظیفه داشت اطلاعات ما را نشان دهد کار خود را به درستی انجام داد. حال نیاز است تا یک کلاس دیگر تحت عنوان Change به معنی "تغییر" ایجاد کنیم اما این در حالی است که این کلاس می بایست از کلاس WhatIsFinal ارث بری کند. برای این منظور کد خود را به شکل زیر تکمیل می کنیم (برای آشنایی بیشتر با مفهوم وراثت در زبان برنامه نویسی جاوا به آموزش های سی و دوم و سی سوم مراجعه نمایید):

```
public class Change extends WhatIsFinal {

}
}
```

دوره آموزش جاوا

کلیه حقوق متعلق به وب سایت نردبان است.

مدرس: بهزاد مرادی

همانطور که قبلاً توضیح داده شد به منظور ارث بری یک کلاس از کلاس دیگر می بایست پس از نام کلاس خود کلید واژه extends را نوشته سپس نام کلاسی را بنویسیم که می خواهیم از آن ارث بری کنیم.

در ادامه کار خود نیاز داریم تا در کلاس Change متدی که قبلاً ایجاد کرده بودیم را Override را رونویسی کنیم. برای این منظور کد فوق را به شکل زیر تکمیل می کنیم:

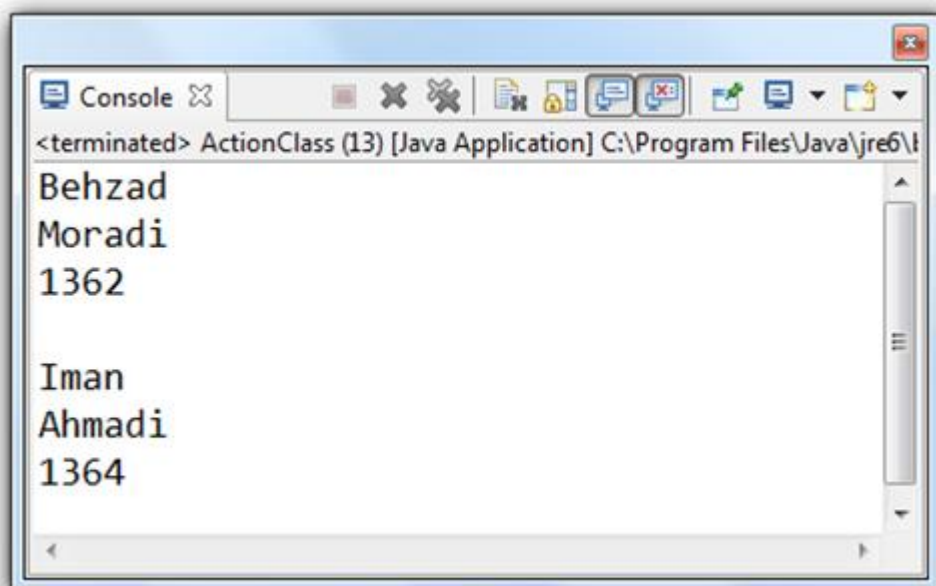
```
public class Change extends WhatIsFinal {
    @Override
    public void showData() {
        String name = "Iman";
        String lastName = "Ahmadi";
        int dateOfBirth = 1364;
        System.out.println(name + "\n" + lastName + "\n" +
dateOfBirth);
    }
}
```

همانطور که در کد فوق ملاحظه می شود برای رونویسی یک متد نیاز است تا ابتدا دستور @Override را نوشته سپس نام متد خود را می نویسیم. تغییری که در این متد انجام داده ایم این است که مقادیر کلاس String و متغیر int را به مقادیر جدیدی تغییر داده ایم. حال مجدداً به کلاس ActionClass بازگشته و یک شیء جدید از روی کلاس Change ایجاد می کنیم:

```
public class ActionClass {
    public static void main(String[] args) {
        WhatIsFinal objectOne = new WhatIsFinal();
        objectOne.showData();
        System.out.println();
        Change objectTwo = new Change();
        objectTwo.showData();
    }
}
```

همانطور که در کد فوق مشاهده می شود پس از شیء اولی که ساختیم دستور System.out.println را می نویسیم. در حقیقت این دستور هیچ تاثیری در روند اجرای برنامه ما نداشته و صرفاً جهت خوانایی بیشتر خروجی برنامه اضافه شده است چرا که مابین خروجی شیء اول و خروجی شیء دوم یک خط فاصله ایجاد خواهد کرد. حال همانطور که شیء اول را از روی

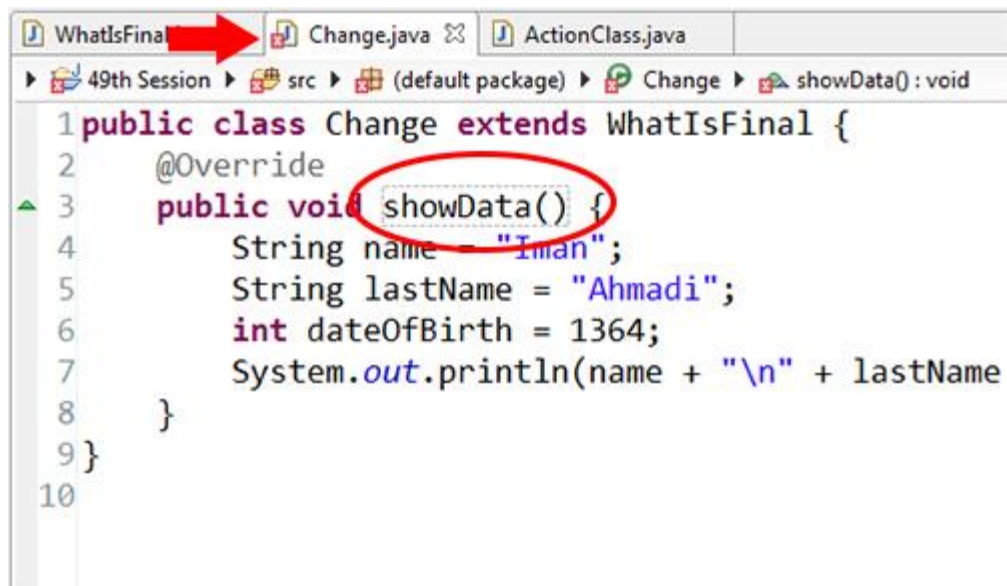
کلاس WhatIsFinal ایجاد کردیم، اکنون یک شیء جدید تحت عنوان objectTwo به معنی "شیء دوم" از روی کلاس Change ایجاد می کنیم. حال مجدد برنامه را اجرا می کنیم:



می بینیم که هر دو شیء ساخته شده در برنامه اجرا شده اند. اکنون می خواهیم تاثیر کلید واژه final را در روند اجرای برنامه تست کنیم. برای این منظور به کلاس WhatIsFinal بازگشته و آن را به صورت زیر تغییر می دهیم:

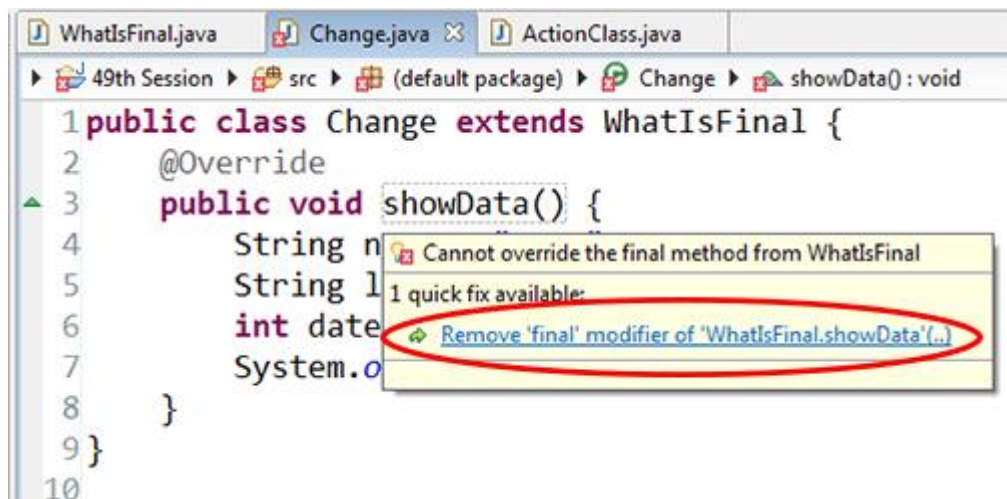
```
public class WhatIsFinal {  
    public final void showData() {  
        String name = "Behzad";  
        String lastName = "Moradi";  
        int dateOfBirth = 1362;  
        System.out.println(name + "\n" + lastName + "\n" +  
dateOfBirth);  
    }  
}
```

همانطور که در کد فوق ملاحظه می شود در بخش نام متد پس از کلید واژه public کلید واژه final را نوشته ایم. به طور خلاصه final این معنی را اطلاق می کند که ما به هیچ وجه امکان ایجاد تغییری را در این متد نخواهیم داشت و به عبارت دیگر نمی توانیم این متد را Override کنیم. در واقع به محض آنکه فایل خود را Save کنیم با تصویر زیر مواجه خواهیم شد:



```
1 public class Change extends WhatIsFinal {
2     @Override
3     public void showData() {
4         String name = "Iman";
5         String lastName = "Ahmadi";
6         int dateOfBirth = 1364;
7         System.out.println(name + "\n" + lastName);
8     }
9 }
10
```

همانطور که در تصویر فوق می بینیم، یک علامت ضربدر کنار نام کلاس Change قرار گرفته است و این بدان معنی است که این کلاس دارای مشکلی است. حال اگر به مرور کد خود پردازیم می بینیم که دور نام متد showData یک نقطه چین قرار گرفته است که همین نقطه چین بخشی از کد ما که دارای مشکل است را نشان می دهد. حال به قرار دادن نشانگر موس خود روی نقطه چین با تصویر زیر مواجه خواهیم شد:



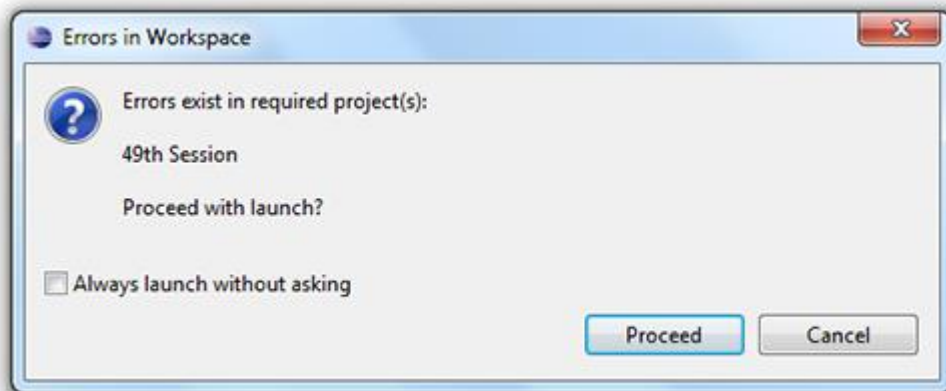
```
1 public class Change extends WhatIsFinal {
2     @Override
3     public void showData() {
4         String n
5         String l
6         int date
7         System.o
8     }
9 }
10
```

می بینیم که در این باکس زرد رنگ نوشته شده است که "ما نمی توانیم یک متد final که در کلاس WhatIsFianl قرار دارد را Override کنیم" و راه کاری هم که به ما پیشنهاد می شود این است که کلید واژه final را از کنار نام متد showData حذف کنیم که با انجام این کار مشکل برنامه ما هم رفع خواهد شد. اکنون اگر با وجود این ایراد در برنامه اقدام به اجرای آن نماییم با تصویر زیر رو به رو خواهیم شد:

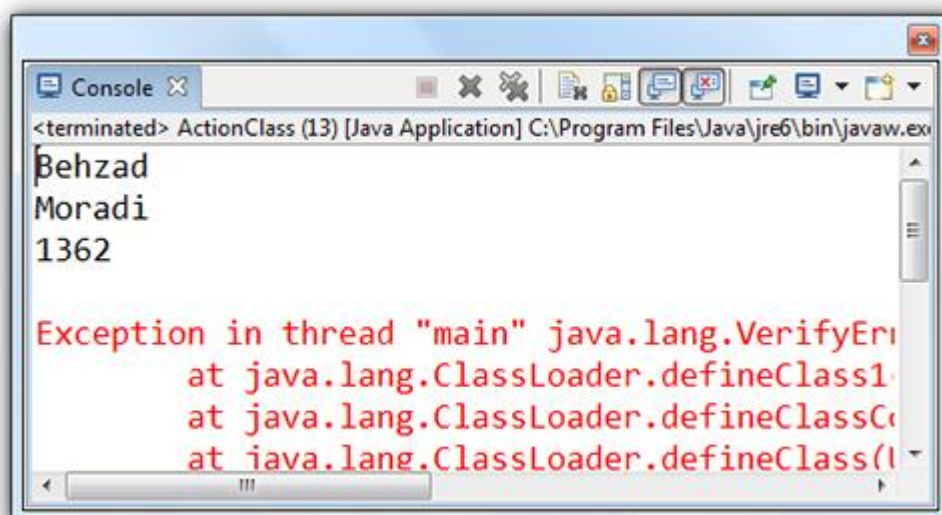
دوره آموزش جاوا

کلیه حقوق متعلق به وب سایت نردبان است.

مدرس: بهزاد مرادی



در این پنجره اکتیپس به ما اخطار می دهد که در پروژه خود دارای مشکلی هستیم اما امکان ادامه کار را نیز با وجود این مشکل برای ما فراهم کرده است. در واقع با کلیک بر روی کلید Proceed می توانیم برنامه خود را Compile کنیم که در این صورت اجرای برنامه ما به این شکل خواهد بود:



می بینیم که شیء ساخته شده اول ما به خوبی نمایش داده شده است اما شیء دوم ما که در آن یک متد final را Override کرده بودیم با مشکل مواجه شده است. پس از مطالعه این آموزش انتظار می رود بتوانیم به سؤالات زیر پاسخ بدهیم:

۱. کاربرد کلید واژه final در جاوا چیست؟
۲. کلید واژه final را به همراه چه عناصری در جاوا می توان مورد استفاده قرار داد و چه تاثیری روی آن دارد؟

دوره آموزش جاوا

کلیه حقوق متعلق به وب سایت نردبان است.

مدرس: بهزاد مرادی