بسم اللّه الرحمن الرحیم

آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

فراخوانی متد جاوا

گردآورنده : مهندس افشین رفوآ

آدرس آموزشگاه : تهران - خیابان شریعتی - بالا تر از خیابان ملک - جنب بانک صادرات - پلاک 561 - واحد 7
88146323 - 88446780 - 88146330

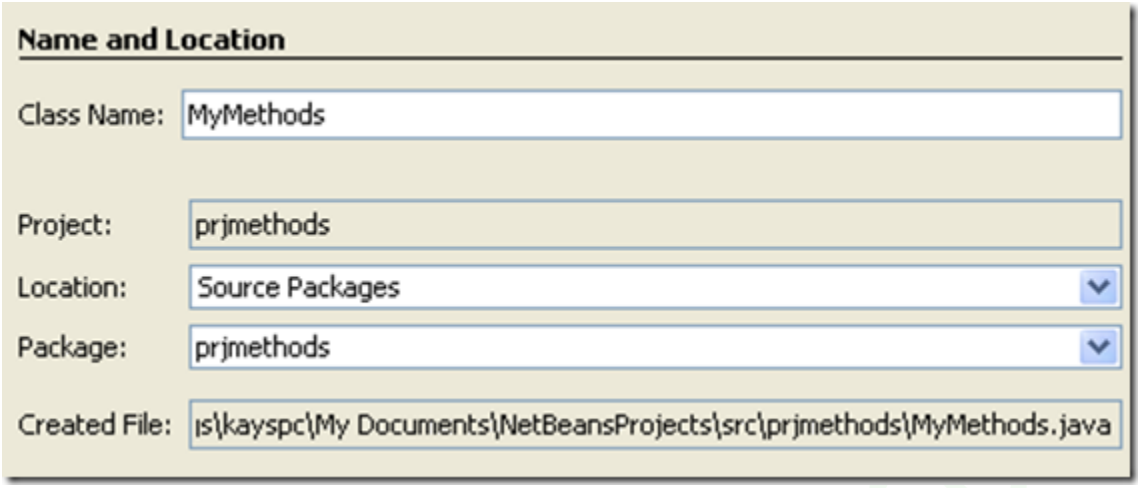
فراخوانی متد جاوا

متدها تا زمانی که آنها را فرا نخوانده و وارد عمل نکنید، کاری انجام نمی دهند. قبل از مشاهده ی چگونگی انجام این کار، اجازه بدهید گروه دیگری را به پروژه اضافه کنیم. به جای مسدود کردن گروه اصلی، می توانیم همه ی متدها را در آن فرا دهیم.(در مورد گروه ها در بخش بعد بیشتر فرا خواهید گرفت).

یک پروژه ی Java Application جدید آغاز کنید. پروژه ی خود را نام گذاری کرده و متد Main نیز تغییر نام دهید. سپس روی Finish کلیک کنید. در تصویر زیر پروژه ی خود را با عنوان prjmethods و گروه را TestMethods نام گذاری کرده ایم.

```
package prjmethods;

public class TestMethods {

    public static void main(String[] args) {


    }

}
```

برای افزودن یک گروه جدید به پروژه ی خود، از منوی NetBeans روی File کلیک کنید. از منوی File عبارت New File را انتخاب کنید. یک دیالوگ باکس برای شما ظاهر خواهد شد. در بخش Categories گزینه ی Java و در بخش File Types گزینه ی Java Class را انتخاب کنید. سپس روی دکمه ی Next در پایین کلیک کنید. در مرحله ی دوم، یک نام برای گروه جدید خود تایپ کنید. ما گروه خود را MyMethods نام نهاده ایم. شما می توانید هر چیز دیگری را به عنوان پیش فرض قرار دهید.



بنابراین ما یک گروه دوم به نام MyMethods ایجاد می کنیم که در پروژه ی prjmethods خواهد بود. روی دکمه Finish کلیک کنید و پس از آن فایل گروه جدید ایجاد خواهد شد. در نرم افزار NetBeans یک تب جدید با کامنت های پیش فرض در مورد چگونگی تغییر الگوها ظاهر خواهد شد. اگر تمایل داشته باشید، می توانید این کامنت ها را حذف کنید. سپس پنجره ای با کد زیر برای شما ظاهر خواهد شد.

```
package prjmethods;


public class MyMethods {


}
```

موردی که باید به آن توجه کنید این است که این بار متد Main وجود ندارد – تنها با یک گروه خالی با نامی که انتخاب کرده اید و یک جفت آکولاد برای کد شما. اجازه بدهید یکی از متدهای خود را اضافه کنیم. بنابراین کد زیر را به گروه خود اضافه کنید.

```
package prjmethods;

public class MyMethods {

      int total() {
          int a_Value = 10 + 10;

          return a_Value;
      }

}
```

این متد int می باشد که قبلا با نام total به آن پرداختیم. در این متد چیزی بین پرانتزها مبنی بر اینکه قصد توزیع هر مقداری بر روی آن نداریم، وجود ندارد. تمام آنچه متد انجام می دهد افزودن 10 + 10 و ذخیره ی پاسخ در متغیری به نام a_Value می باشد. این مقداری است که از این متد بازگردانده خواهد شد. مقداری که پس از لغت کلیدی گزارش شده است، باید با نوع بازگشتی از تیتر متد هماهنگ باشد. در مورد مثال ما این امر درست می باشد، زیرا هر دو از نوع int می باشند.

(این مسئله مهم می باشد که در ذهن داشته باشید که متغیر a_Value در خارج از متد total دیده نمی شود: هر متغیری که در داخل متد تنظیم شده باشد، در خارج آن متد قابل دسترس نمی باشد. این متغیر با عنوان متغیر local شناخته می شود – در واقع در داخل متد قرار می گیرد)

برای فراخوانی متد total تب TestMethods را در NetBeans انتخاب کنید، موردی با متد Main. قصد داریم متد total را از متد Main فرابخوانیم.

اولین کاری که باید انجام شود، ایجاد یک آبجکت جدید از گروه MyMethods می باشد. خط زیر را به متد Main خود اضافه کنید.

```
package prjmethods;

public class TestMethods {

    public static void main(String[] args) {

        MyMethods test1 = new MyMethods();

    }

}
```

برای ایجاد یک آبجکت جدید از یک گروه، با نام گروه آغاز کنید، در مورد مثال ما گروه MyMethods نامیده می شود. این به جای int، double، String و غیره می باشد. به عبارت دیگر نوع متغیری که در حال ایجاد آن هستید یک متغیر MyMethods می باشد. پس از یک فاصله، یک نام برای متغیر جدید MyMethods تایپ کنید. ما مورد خود را test1 نامیده ایم.(زیر این مورد خط کشیده شده، زیرا هنوز با آن کاری نکرده ایم. این یک مورد NetBeans می باشد)

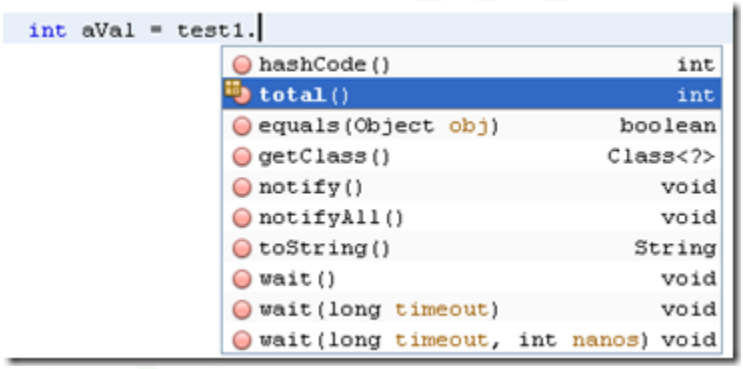
یک علامت تساوی بعد از آن قرار می گیرد که با لغت کلیدی new دنبال می شود که به معنای new object می باشد. پس از لغت کلیدی new یک فاصله قرار دهید که مجددا با نام گروه شما دنبال می شود. این بار پس از نام گروه نیاز به یک جفت پرانز دارید. خط را به روش معمول به پایان برسانید، با یک نقطه ویرگول (semi-colon).

آنچه در اینجا انجام داده ایم، ایجاد یک آبجکت MyMethods جدید با نام test1 می باشد. اکنون متد total در داخل گروه MyMethods از متد Main مربوط به گروه TestMethods در دسترس خواهد بود.

برای وارد عمل کردن متد، خطوط زیر را اضافه کنید.

```
public class TestMethods {

    public static void main(String[] args) {

        MyMethods test1 = new MyMethods();

        int aVal = test1.total();

    }
}
```

ما در حال تنظیم یک متغیر int با نام aVal می باشیم. پس از علامت تساوی نام گروه ما، test1، قرار می گیرد. برای دسترسی به متدها در گروه یک نقطه تایپ کنید. NetBeans جعبه ای را با متدهای موجود نمایش می دهد.

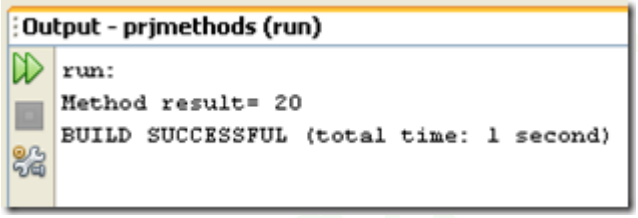


متغیر total در لیست موجود می باشد (دیگر متغیرها در متدها داخلی می باشند.). پرانتزها خالی هستند، زیرا متد ما مقادیر را نمی پذیرد، اما نوع بازگشتی، int، در سمت راست نمایش داده می شود. روی total دابل کلیک کنید تا آن را به کد خود اضافه کنید. سپس در انتهای خط یک نقطه ویرگول تایپ کنید.

در انتها یک خط چاپی اضافه کنید.

```
public class TestMethods {

    public static void main(String[] args) {

        MyMethods test1 = new MyMethods();

        int aVal = test1.total();

        System.out.println( "Method result= " + aVal );

    }
}
```

وقتی که کد اجرا می شود، پنجره ی Output صفحه ی زیر را نمایش خواهد داد.



بنابراین برای فراخوانی یک متد که یک مقدار را بازمی گرداند، دقت کنید که چه نوع مقداری توسط متد شما بازگردانده می شود. سپس این مقدار را به یک متغیر جدید اختصاص دهید، در مورد ما این متغیر aVal می باشد. اما وقتی که یک نقطه پس از نام آبجکت خود تایپ می کنید، متد باید در دسترس باشد.

به هرحال اگر متد شما از نوع void می باشد، نیازی به اختصاص دادن آن به متغیر جدیدی مانند aVal نمی باشد. به عنوان مثال به گروه MyMethods خود بازگشته و متد void را که قبلا بررسی کردید، اضافه کنید.

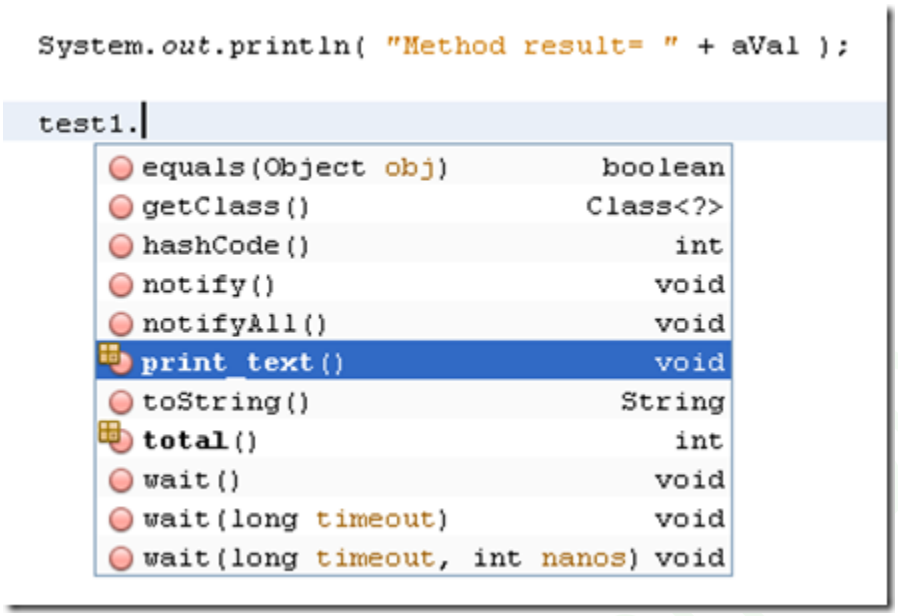
```
package prjmethods;

public class MyMethods {

      int total() {
         int a_Value = 10 + 10;

         return a_Value;
      }

      void print_text() {

         System.out.println( "Some Text Here" );
      }

}
```

این متد جدید print_text نامیده می شود. این متد نیز دارای پرانتزهای خالی می باشد که به آنها هیچ مقداری اختصاص نمی دهیم. تمام کاری که این متد انجام می دهد چاپ کردن متن می باشد.

زمانی که متد void را اضافه کرده اید، به گروه TestMethods بازگردید و خط زیر را اضافه کنید.

```
test1.print_text( )
```

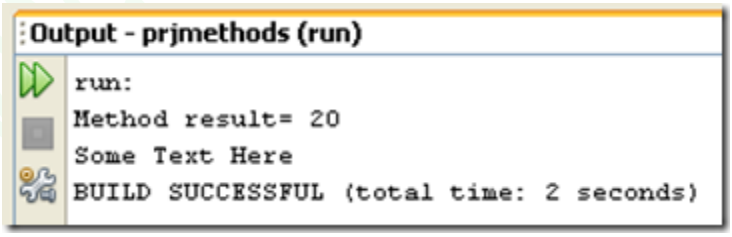
به محض اینکه نقطه را تلیپ کردید، باید متد جدید را مشاهده کنید که روی لیست ظاهر می شود.



اما متدهای ما که اکنون روی لیست هستند عبارتند از total و print_text. مقادیری که این متدها در سمت نمایش می دهند int و void می باشند.

از آنجایی که متد print_text یک متد خالی (void) می باشد، نیازی به تنظیم یک مقدار بازگشتی ندارید. تمام آنچه نیاز دارید نام آبجکت، یک نقطه (dot) و یک متد void می باشد که قصد فراخوانی آن را دارید. سپس جاوا تنها با اجرای کد در داخل متد شما ادامه خواهد داد.

کد خود را اجرا کنید و پنجره ی Output شما باید صفحه ای مانند زیر نمایش دهد.



در بخش بعد نگاه دقیق تری به انتقال مقادیر به متدها خواهیم داشت.