

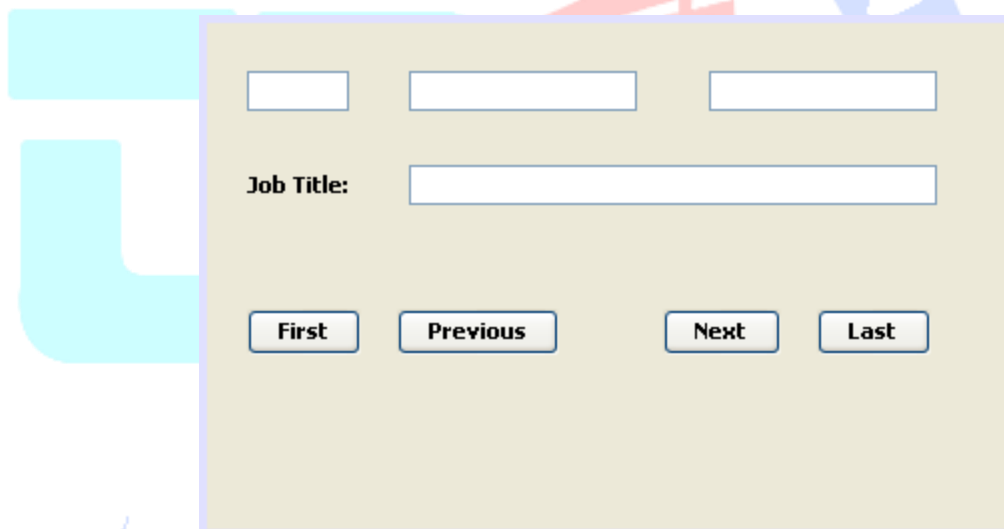
آموزش دکمه های پیمایش در جاوا

آنچه در اینجا انجام خواهیم داد، افزودن چهار دکمه به فرم می باشد. دکمه ها ما را قادر خواهند کرد تا از طریق رکوردها به جلو برویم، به عقب بازگردیم، یا وارد اولین و یا آخرین رکورد شویم.

بنابراین یک پنل جدید به فرم خود اضافه کنید. آن را بزرگ کرده و سپس دکمه ها را به پنل اضافه کنید. نام های متغیرهای دکمه ها را مانند زیر تغییر دهید:

btnNext btnPrevious btnLast btnFirst

متن روی هر دکمه را به Next, Previous, Last, First تغییر دهید. سپس فرم شما چیزی مشابه تصویر زیر خواهد بود:



The screenshot shows a form with a light beige background. At the top, there are three empty text input fields. Below them is a label 'Job Title:' followed by a single wide text input field. At the bottom of the form, there are four buttons labeled 'First', 'Previous', 'Next', and 'Last' arranged horizontally.

حرکت به رکورد بعدی:

روی دکمه ی Next دابل کلیک کنید تا یک code stub ایجاد کنید.

شما با دکمه ی Next دو کار می توانید انجام دهید: اول بررسی اینکه آیا رکورد دیگری وجود دارد تا به آن وارد شوید و دوم اینکه اگر رکورد دیگری وجود دارد آن را در Text Fields نمایش دهید. برای این مسئله می توانیم یک IF Statement ایجاد کنیم. اما لازم است در یک بلوک try ... catch قرار گیرد. بنابراین کد زیر را به code stub مربوط به دکمه ی Next وارد کنید:

مثال :

```
try {  
if ( rs.next( ) ) {
```

آدرس آموزشگاه : تهران - خیابان شریعتی - بالا تر از خیابان ملک - جنب بانک صادرات - پلاک 651 طبقه دوم - واحد7

88146323 - 88446780 - 88146330

```

}
else {
rs.previous( );
JOptionPane.showMessageDialog(Workers.this, "End of File");
}
}
catch (SQLException err) {
JOptionPane.showMessageDialog(Workers.this, err.getMessage());
}
}

```

IF Statement در واقع **ResultSet** را در یک زمان روی یک رکورد جابه جا می کند. اگر رکورد دیگری وجود نداشته باشد، یک مقدار **false** گزارش می شود. بخش **Else** در رکورد **resultSet** را به عقب بازمی گرداند، به این علت که نشانگر آخرین رکورد را انتقال خواهد داد.

در آکولادهای مربوط به **IF Statement** ، می توانیم کدی را برای نمایش رکورد در **Text Fields** اضافه کنیم:

مثال :

```

int id_col = rs.getInt("ID");
String id = Integer.toString(id_col);
String first = rs.getString("First_Name");
String last = rs.getString("Last_Name");
String job = rs.getString("Job_Title");
textID.setText(id);
textFirstName.setText(first);
textLastName.setText(last);
textJobTitle.setText(job);

```

این همان کدی است که در متود **DoConnect** داریم. (می توانستیم یک متود جدید ایجاد کنیم، بنابراین برای جلوگیری از تکرار هر گونه کد و ساده بودن آن این کار را کردیم).

کد مربوط به دکمه ی **Next** باید مشابه کد زیر باشد:

آموزشگاه تحلیگر داده ها

```

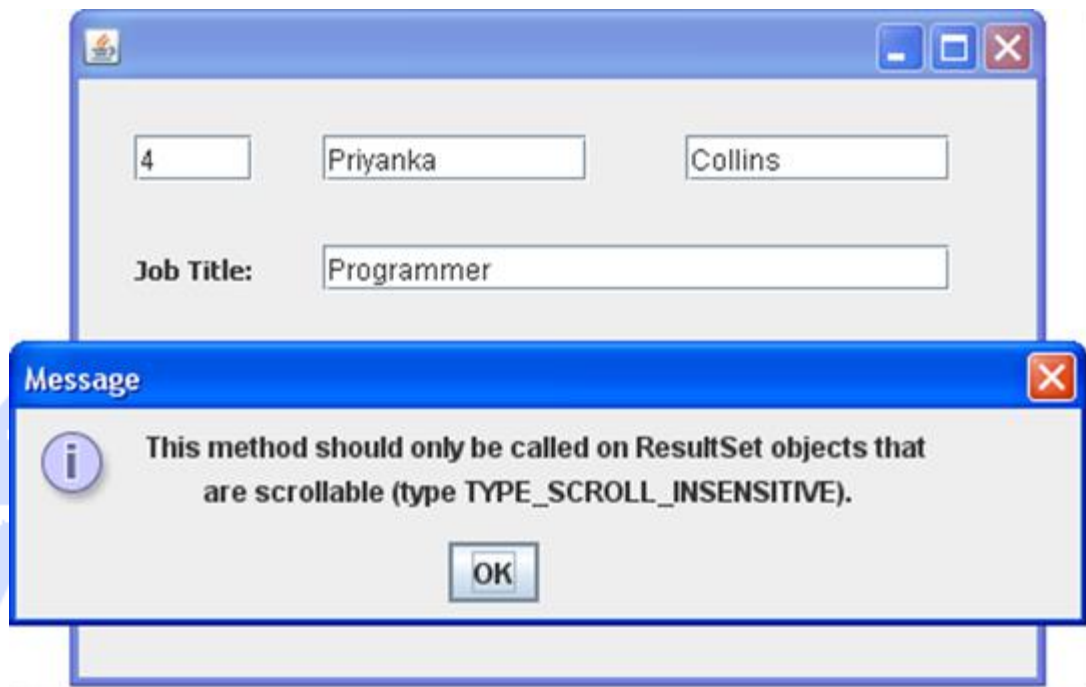
try {
    if (rs.next()) {
        int id_col = rs.getInt("ID");
        String id = Integer.toString(id_col);
        String first = rs.getString("First_Name");
        String last = rs.getString("Last_Name");
        String job = rs.getString("Job_Title");

        textID.setText(id);
        textFirstName.setText(first);
        textLastName.setText(last);
        textJobTitle.setText(job);
    }
    else {
        rs.previous();
        JOptionPane.showMessageDialog(Workers.this, "End of File");
    }
}
catch (SQLException err) {
    JOptionPane.showMessageDialog(Workers.this, err.getMessage());
}

```

وقتی کد خود را اضافه کردید، برنامه ی خود را اجرا کرده و آن را امتحان کنید. اگر به کلیک کردن روی دکمه ی next ادامه دهید، همه ی داده ها را در جدول مشاهده خواهید کرد. به هر حال در اینجا مشکلی وجود دارد.

وقتی به آخرین رکورد می رسید، باید یک پیغام خطا مشاهده کنید:



مشکل این است که ما یک خط `rs.previous` اضافه کرده ایم. به هر حال ما از نوع `ResultSet` پیش فرض استفاده کرده ایم. همانطور که در آخرین بخش توضیح دادیم، این خط به شما یک `ResultSet` ارائه می دهد که فقط می تواند به جلو حرکت کند. ما می توانیم از نوع پیشنهاد شده در پیغام خطا استفاده کنیم.

برنامه ی خود را متوقف کرده و به پنجره ی برنامه نویسی (coding window) خود بازگردید. در متود `DoConnect`، خط زیر را قرار دهید:

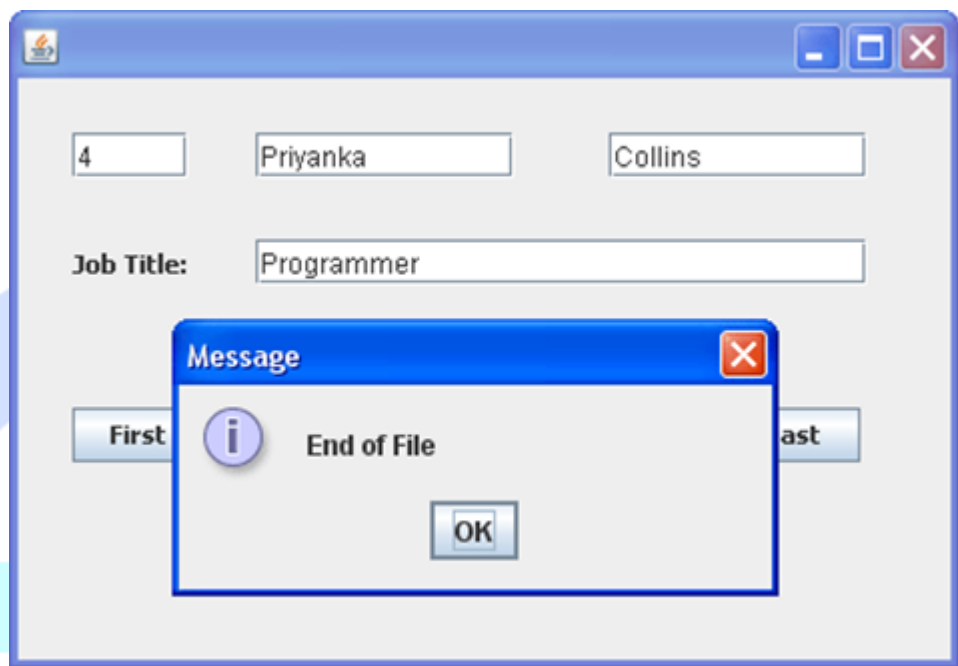
```
stmt = con.createStatement( );
```

آن را مانند زیر تغییر دهید (کد شما باید در یک خط قرار گیرد.):

```
stmt = con.createStatement(  
ResultSet.TYPE_SCROLL_INSENSITIVE, ResultSet.CONCUR_UPDATABLE );
```

اکنون نوع `ResultSet` به ما اجازه خواهد داد تا به عقب نیز برگردیم.

مجددا برنامه ی خود را اجرا کنید. آنقدر روی دکمه ی `Next` کلیک کنید تا به آخرین رکورد برسید. شما باید پیغام خطایی از بخش `try ... catch` مشاهده کنید:



در بخش بعد در مورد چگونگی بازگشتن به عقب از طریق رکوردهای دیتابیس صحبت خواهیم کرد .

آدرس آموزشگاه : تهران - خیابان شریعتی - بالا تر از خیابان ملک - جنب بانک صادرات - پلاک 651 طبقه دوم - واحد 7

88146323 - 88446780 - 88146330

<http://www.tahlildadeh.com/>