

## User Guide 5.3

### Getting values from a Selected Row of a Gridview

There are various means to getting the values back from a 'selected' row in a grid. This is one example. Assume we have a list of students and we want to click on one student in a grid and then move to another page and display their entire profile.

Build a gridview that displays their First Name, Last Name and a Select Link (Button), also pull the primary key for the personID. You might display it as:

#### **Select Student to Update**

Select Carrot, Jill  
Select Date, Debbie  
Select Eggplant, Edgar

The business logic is when a user click the Select Link, you want to retrieve their personID as well as their First and Last Names to display on the next page.

1. Build the grid as normal, Enable Selection on the Grid View options dialog box, Insure that the datakey field for the grid is the personID.
2. Now we need to trap the event when the user clicks on the SELECT link and retrieve the ID and First and Last Names to use on the next page.
3. Click on the grid, then in the properties window click EVENTS (lightning bolt),
  - a. Double Click on the ROW COMMAND event
  - b. Double Click on the SELECTED INDEX CHANGED
4. Add a variable to be used on the page, add the definition after the Partial Class definition at the top of the page, and after the first {
  - a. bool isSelected = false;
5. In the RowCommand Subroutine capture which link button was pressed with a SWITCH (in VB would be a SELECT Case) command, notice below I have added Response.Write commands that will display on the top left of your screen which button was selected. You may remove once they work properly

```
switch (e.CommandName)
{
    case "Select":
        Response.Write ("Selected ");
        isSelected = true;
        break;
    case "Edit":
```

```

        Response.Write ("Edit");
        break;
    case "Delete":
        Response.Write("Delete");
        break;
    }

```

6. In the selected index subroutine we will capture the personID as well as the first name and last name for display on the next web page

In this subroutine we need to capture if the SELECT was click versus another link button (i.e. EDIT or DELETE)

```

// verify that the SELECT was clicked versis another link, only grab values for the SELECT

```

```

if (isSelected)
{
    // Grab the primary key for the row selected
    int primaryKey = GridView1.SelectedValue;

    // get the alues of the current row selected and move to a variable named 'row'
    GridViewRow row = GridView1.SelectedRow;

    // get the values for first and last name, you must have converted First and Last
to Template Columns
    // you must have also named the LABELS in the ITEM template for these objects
    Label firstname = (Label)row.FindControl("_firstname");
    Label lastname = (Label)row.FindControl("_lastname");

    // now move these items to an object (person)
    // create an instance of the PersonInfo Class
    // See hint 3.1 to create a quick Person Class

    personInfo personSelected = new personInfo();
    personSelected.PersonId = primaryKey;
    personSelected.FirstName = firstname.Text;
    personSelected.LastName = lastname.Text;

    // save this info in a SESSION Variable to use on the next page
    Session["personSelected"] = personSelected;

    //redirect to proper page
}

```

7. To retrieve on the next web page:
  - a. Declare a variable to be used on the page:

```

PersonInfo personSelected = new PersonInfo();

```

8. In the Page Load Sub, retrieve the Session variable that is passed from the first page:

```

//the above is the person who logged in
//check to see if the admin wants to update someone else
var testifNull = session["personSelected"];

```

```
if (testifNull != null)
{
    personSelected = (personInfo)Session["personselected"];
    iPK = Convert.ToInt32(personSelected.PersonId);
}
```

9. Now in `personSelected.PersonID` you know the ID of the student to be updated and can pull their profile or other information, likewise you can display their name too!