

# CIS 284 Exercise 3 – EmployeeApp II

Make a copy of your original EmployeeApp solution created in Exercise 2. In this exercise, you will enhance the copy as follows.

- The “Give ... raise” button uses the object’s Raise method to give each employee a raise.
- Adjust the DataGridView to display the salaries in currency format. You may set the column’s DefaultCellStyle.Format property using either the Designer, or a statement.
- Hide the Phone column.
- Replace (or supplement) the custom navigation buttons with a BindingNavigator control.
- Use a *For Each* loop to determine and display the total monthly and annual payrolls, which are the sum of the MonthlySalary and AnnualSalary property values for all employees.

The screenshot shows the 'Employees' application window. It features a DataGridView with 7 rows and 4 columns: Name, MonthlySalary, AnnualSalary, and an empty column. The data is as follows:

|   | Name            | MonthlySalary | AnnualSalary |
|---|-----------------|---------------|--------------|
|   | Tom Thompson    | \$1,210.00    | \$14,520.00  |
|   | Georgia Kenedy  | \$1,452.00    | \$17,424.00  |
|   | Pete Peterson   | \$1,306.00    | \$15,672.00  |
|   | Meredith Jones  | \$1,974.00    | \$23,688.00  |
| ▶ | Jean Cummings   | \$1,887.00    | \$22,644.00  |
|   | Michael Robbins | \$1,320.00    | \$15,840.00  |
|   | Polly Marks     | \$1,672.00    | \$20,064.00  |

Below the table, there are input fields for the selected employee (Jean Cummings):

Name: Jean Cummings  
Phone: (618) 555-2195  
Monthly Salary: \$1,887.00  
Annual Salary: \$22,644.00

A button labeled 'Give everyone a 10% raise' is located below these fields.

To the right, there is a 'Search' section with a text box containing 'jean' and two buttons: 'Previous' and 'Next'.

At the bottom, a 'Payroll Summary' section displays:

Total Monthly Payroll: \$10,821.00  
Total Annual Payroll: \$129,852.00

Finally, add a Search feature that uses a loop to do a sequential search for a partial employee name. The *Next* button searches forward from the current position to the next employee with a name that includes the specified string. If found, the employee becomes the current employee. The *Previous* button moves backward through the employees to the closest previous employee with a matching name. The search is not case sensitive. In both cases, if the search fails, an appropriate message appears.