|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Method Name:**  calculateGrossPay | | | **Class Name:** Salary Employee | | | **ID:** 1 |
| **Contract ID:** 1 | | | **Programmer:** Brooks Robinson | | | **Date Due:** 11/15/22 |
| **Programming Language:**  □ **Visual Basic**  □ **Smalltalk** þ **C#**  □ **Java** | | | | | | |
| **Triggers/Events:**  The system is calculating payroll for the employee list, and one of the employees is on salary. | | | | | | |
| **Arguments Received:**  **Data Type:** | **Notes:** | | | | |
|  |  | | | | |
| **Messages Sent & Arguments Passed:**  **ClassName.MethodName:** | | | **Data Type:** | **Notes:** | |
|  | | |  |  | |
| **Argument Returned:**  **Data Type:** | **Notes:** | | | | |
| Decimal | This is gross pay for the employee. Should be rounded to the hundredths place. | | | | |
| **Algorithm Specification:**  RETURN self.Salary | | | | | |
| **Misc. Notes:**  An employee’s salary property is the amount per payroll, so just returning that is what he/she will make for any payroll.  This method will override its parent’s abstract method. | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Method Name:**  calculateGrossPay | | | **Class Name:** Hourly Employee | | | **ID:** 2 |
| **Contract ID:** 2 | | | **Programmer:** Brooks Robinson | | | **Date Due:** 11/15/22 |
| **Programming Language:**  □ **Visual Basic**  □ **Smalltalk** þ **C#**  □ **Java** | | | | | | |
| **Triggers/Events:**  The system is calculating payroll for the employee list, and one of the employees is hourly. | | | | | | |
| **Arguments Received:**  **Data Type:** | **Notes:** | | | | |
|  |  | | | | |
| **Messages Sent & Arguments Passed:**  **ClassName.MethodName:** | | | **Data Type:** | **Notes:** | |
|  | | |  |  | |
| **Argument Returned:**  **Data Type:** | **Notes:** | | | | |
| Decimal | This is gross pay for the employee. Should be rounded to the hundredths place. | | | | |
| **Algorithm Specification:**  RETURN self.hourlyPay \* self.hoursWorked | | | | | |
| **Misc. Notes:**  An employee’s gross pay is hour works for that pay period multiplied by the hourly wages.  This method will override its parent’s abstract method. | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Method Name:**  generateEmployeeList | | | **Class Name:** Employee List | | | **ID:** 3 |
| **Contract ID:** 3 | | | **Programmer:** Ryan Porter | | | **Date Due:** 11/15/22 |
| **Programming Language:**  □ **Visual Basic**  □ **Smalltalk** þ **C#**  □ **Java** | | | | | | |
| **Triggers/Events:**  The user needs a list of all employees (deletion or payroll use cases). | | | | | | |
| **Arguments Received:**  **Data Type:** | **Notes:** | | | | |
| compFedID: Int | The company ID | | | | |
|  |  | | | | |
| **Messages Sent & Arguments Passed:**  **ClassName.MethodName:** | | | **Data Type:** | **Notes:** | |
| Employee.Create(…) | | | Employee | Creates new employee.  Employee has many needed parameters, so did not list them. | |
| **Argument Returned:**  **Data Type:** | **Notes:** | | | | |
|  |  | | | | |
| **Algorithm Specification:**  CONNECT to Database  IF connected  FOR EACH Employee with that comdFedID  Add to self.employeeList Employee.Create(…)  CLOSE connection  ELSE  Display connection error message | | | | | |
| **Misc. Notes:**  Will need to connect to the main database once. | | | | | |