

Code Review 紀錄內容			更新日期：2024/08/14
日期	CR 人員	題號	備註
2024/08/14	林家暉	3. 員工資料存取	
9. 變數型別改為 BigDecimal，並改變因應算法	<div>修改前：</div> <pre>private int bonus; private int payment; public Sales(String name, String department, int salary, int bonus) { super(name, department, salary); this.bonus = bonus / 20; this.payment = salary + getBonus(); }</pre> <div>修改後：</div> <pre>private BigDecimal bonus; private BigDecimal payment; public Sales(String name, String department, BigDecimal salary, BigDecimal bonus) { super(name, department, salary); this.bonus = bonus.multiply(new BigDecimal("0.05")); this.payment = salary.add(getBonus()); }</pre> <div>※8/14 更新：固定常數宣告為成員變數</div> <pre>private BigDecimal bonus; private BigDecimal payment; private static final BigDecimal BONUS_PERCENTAGE = new BigDecimal("0.05"); public Sales(String name, String department, BigDecimal salary, BigDecimal bonus) { super(name, department, salary); this.bonus = bonus.multiply(BONUS_PERCENTAGE); this.payment = salary.add(getBonus()); }</pre>		
日期	CR 人員	題號	備註
2024/08/14	林家暉	4. 輸出員工資料	
12. 優化程式碼	<div>相同條件可用同一個迴圈</div> <div>修改前：</div> <pre>for (Employee employee : employeeList) { if (employee instanceof Supervisor) { sb.append(employee.getName()).append(',').append(((Supervisor) employee).getPayment()).append('\n'); bufferedWriter.write(sb.toString()); sb.setLength(0); } } for (Employee employee : employeeList) { if (employee instanceof Sales) { sb.append(employee.getName()).append(',').append(((Sales) employee).getPayment()).append('\n'); bufferedWriter.write(sb.toString()); sb.setLength(0); } }</pre>		

12. 優化程式碼	<div>修改後：</div> <pre>for (Employee employee : employeeList) { if (employee instanceof Supervisor) { sb.append(employee.getName()).append(',').append(((Supervisor) employee).getPayment()).append('\n'); bufferedWriter.write(sb.toString()); sb.setLength(0); } else { sb.append(employee.getName()).append(',').append(((Sales) employee).getPayment()).append('\n'); bufferedWriter.write(sb.toString()); sb.setLength(0); } }</pre> <div>※8/14 更新：使用 if...else if 判斷式避免其他型別誤入，強轉成 Sales</div> <pre>for (Employee employee : employeeList) { if (employee instanceof Supervisor) { sb.append(employee.getName()).append(',').append(((Supervisor) employee).getPayment()).append('\n'); bufferedWriter.write(sb.toString()); sb.setLength(0); } else if (employee instanceof Sales) { sb.append(employee.getName()).append(',').append(((Sales) employee).getPayment()).append('\n'); bufferedWriter.write(sb.toString()); sb.setLength(0); } }</pre>		
日期	CR 人員	題號	備註
2024/08/14	林家暉	10. 員工資料存取	
價格排序用 BigDecimal	<div>使用 String 只會比較第一位數字</div> <div>修改前：</div> <pre>List<Map<String, String>> carMapList = readCsvToList(filePath); Collections.sort(carMapList, new Comparator<Map<String, String>>() { public int compare(final Map<String, String> o1, final Map<String, String> o2) { return o2.get("Price").compareTo(o1.get("Price")); } });</pre> <div>修改後：</div> <pre>List<Map<String, String>> carMapList = readCsvToList(filePath); Collections.sort(carMapList, new Comparator<Map<String, String>>() { public int compare(final Map<String, String> o1, final Map<String, String> o2) { BigDecimal price1 = new BigDecimal(o1.get("Price")); BigDecimal price2 = new BigDecimal(o2.get("Price")); return price2.compareTo(price1); } });</pre> <div>※8/14 更新：BigDecimal 變數只有使用一次，不需宣告為變數，可直接使用</div> <pre>List<Map<String, String>> carMapList = readCsvToList(filePath); Collections.sort(carMapList, new Comparator<Map<String, String>>() { public int compare(final Map<String, String> o1, final Map<String, String> o2) { return new BigDecimal(o2.get("Price")).compareTo(new BigDecimal(o1.get("Price"))); } });</pre>		