

Ch12上課練習

鄭安翔

ansel_cheng@hotmail.com

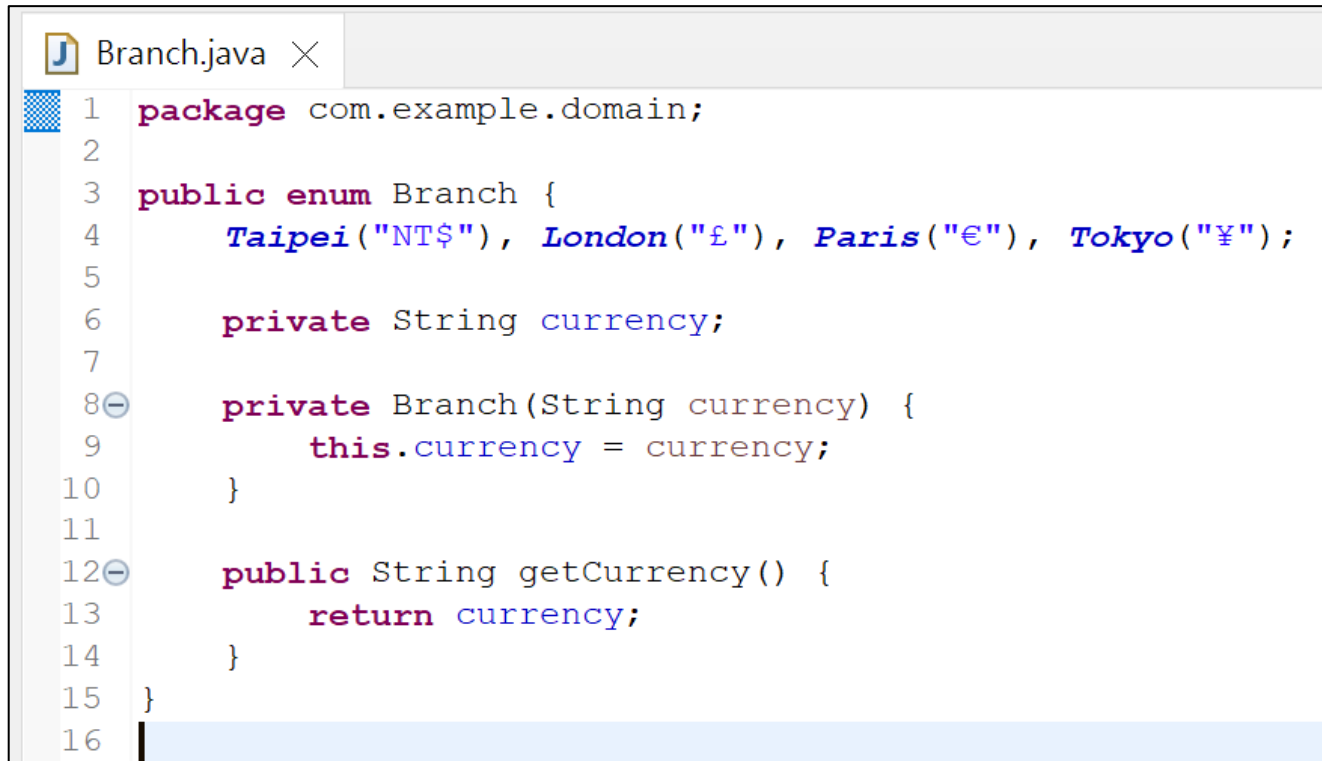
練習一 使用Enum

1. 修改 EmployeePractice 專案
2. 建立分公司 Enum 檔案 : com.example.domain.Branch.java
 - ❑ 宣告 String currency 屬性及建立對應建構子
 - ❑ 提供 public String getCurrency() 方法
 - ❑ 宣告 Taipei(NT\$), London(£), Paris(€) 及Tokyo(¥)
3. 修改 Employee 類別
 - ❑ 建構子傳入分公司屬性
 - ❑ 修改toString() 顯示薪水時依分公司加上貨幣符號
4. 修改 EmployeeTest 類別
 - ❑ 建構員工時增加分公司資訊
5. 測試、執行

<<enumeration>> Branch	
Taipei	
London	
Paris	
Tokyo	
- currency: String	
- Branch(curr: String)	
+ getCurrency: String	

<<abstract>> Employee	
- nextId : int	
- empld : int	
- name : String	
- ssn : String	
- salary : double	
- branch: Branch	
+ Employee(name: String, ssn: String, salary: double, branch: Branch)	
+ toString() : String	

新增 Branch.java



```
Branch.java ×
1 package com.example.domain;
2
3 public enum Branch {
4     Taipei("NT$"), London("£"), Paris("€"), Tokyo("¥");
5
6     private String currency;
7
8     private Branch(String currency) {
9         this.currency = currency;
10    }
11
12    public String getCurrency() {
13        return currency;
14    }
15 }
16
```

修改Employee.java

```
Employee.java ×
1 package com.example.domain;
2
3 import java.text.NumberFormat;
4
5
6
7 public abstract class Employee {
8     public static int nextId = 101;
9     private int empId;
10    private String name = "John";
11    private String ssn = "A123456789";
12    private double salary = 26400;
13    private Branch branch;
14    protected NumberFormat formatter = NumberFormat.getInstance();
15
16    public Employee(String name, String ssn, double salary, Branch branch) {
17        this.empId = nextId++;
18        if(name!=null && name.length()!=0)
19            this.name = name;
20        if(ssn!=null && ssn.length()!=0)
21            this.ssn = ssn;
22        if(salary>27420)
23            this.salary = salary;
24        this.branch = branch;
25    }
26
```

```
60
61    public Branch getBranch() {
62        return branch;
63    }
64
65    @Override
66    public String toString() {
67        return "====員工資料===="+"\n"+
68            "編號: "+this.empId+"\n"+
69            "姓名: "+this.name+"\n"+
70            "SSN: "+this.ssn+"\n"+
71            "薪水: "+this.branch.getCurrency()+formatter.format(salary)+"元"+"\\n";
72    }
73
```

修改Admin/Engineer/Manager.java

```
Admin.java X
1 package com.example.domain;
2
3 public class Admin extends Employee {
4     private int hours;
5
6     public Admin(String name, String ssn, double salary, Branch branch, int hours) {
7         super(name, ssn, salary, branch);
8         this.hours = hours;
9     }
10
11     @Override
12     public double getPay() {
13         return this.getSalary()/160*h
14     }
15
16
17 }
18
```

```
*Engineer.java X
1 package com.example.domain;
2
3
4 public class Engineer extends Employee {
5     private String[] skills;
6     private int skillCount;
7
8     public Engineer(String name, String ssn, double salary, Branch branch, {
9         super(name, ssn, salary, branch);
10        skills = new String[5];
11        skillCount = 0;
12    }
13
```

```
Manager.java X
1 package com.example.domain;
2
3 import java.util.ArrayList;
4
5 public class Manager extends Employee {
6     private String deptName;
7     protected ArrayList employees;
8
9     public Manager(String name, String ssn, double salary, Branch branch, String deptName) {
10        super(name, ssn, salary, branch);
11        this.deptName = deptName;
12        this.employees = new ArrayList();
13    }
14
```

修改Director.java

```
Director.java ×
1 package com.example.domain;
2
3 public class Director extends Manager {
4     private double budget;
5
6     public Director(String name, String ssn, double salary, Branch branch, String deptName, double budget) {
7         super(name, ssn, salary, branch, deptName);
8         this.budget = budget;
9     }
10
11
12     public double getPay() {
13
14
15
16     public double getBudget() {
17
18
19
20     @Override
21     public String toString() {
22         return super.toString() +
23             "管理預算: " + this.getBranch().getCurrency() + formatter.format(budget) + "\n";
24     }
25
26 }
27
```

修改EmployeeTest.java

```
EmployeeTest.java ×
1 package com.example;
2
3 import com.example.domain.Admin;
4
5
6
7
8
9
10 public class EmployeeTest {
11
12     public static void main(String[] args) {
13         Employee[] emps = new Employee[5];
14         emps[0] = new Admin("Sean", "A123456789", 50000, Branch.Taipei, 180);
15         emps[1] = new Admin("Amy", "B210987654", 70000, Branch.London, 120);
16         emps[2] = new Engineer("David", "C109876543", 80000, Branch.Tokyo);
17         emps[3] = new Manager("Louis", "D124680135", 100000, Branch.Taipei, "TW Sales");
18         emps[4] = new Director("Nicole", "R202468135", 120000, Branch.Paris, "Global Sales", 1000000);
19
20         for(int i=0; i<emps.length; i++)
21             System.out.println(emps[i]);
22
23         System.out.println("David 學會了Java, Android");
24         if(emps[2] instanceof Engineer) {
25             Engineer eng = (Engineer) emps[2];
26             eng.addSkill("Java");
27             eng.addSkill("Android");
28         }
29
30         System.out.println("部門分配.....");
31         if(emps[3] instanceof Manager) {
32             Manager m1 = (Manager) emps[3];
33             m1.addEmployee(emps[0]);
34             m1.addEmployee(emps[1]);
35             m1.addEmployee(emps[2]);
36         }
37
38         ((Manager) emps[4]).addEmployee(emps[3]);
39
40         for(int i=0; i<emps.length; i++)
41             System.out.println(emps[i].getName()+"本月薪資"+emps[i].getBranch().getCurrency()+emps[i].getPay()+"元");
42
43     }
44 }
45
```

測試、執行

Console ×

<terminated> EmployeeTest

=====員工資料=====

編號: 101
姓名: Sean
SSN: A123456789
薪水: NT\$50,000元

=====員工資料=====

編號: 102
姓名: Amy
SSN: B210987654
薪水: £70,000元

=====員工資料=====

編號: 103
姓名: David
SSN: C109876543
薪水: ¥80,000元

=====員工資料=====

編號: 104
姓名: Louis
SSN: D124680135
薪水: NT\$100,000元
管理部門: TW Sales

=====員工資料=====

編號: 105
姓名: Nicole
SSN: R202468135
薪水: €120,000元
管理部門: Global Sales
管理預算: €1,000,000