

## LAB GUIDE 3 – Looping Constructs – Classes and Objects

**Mục tiêu:** Sau bài thực hành này, học viên sẽ biết cách :

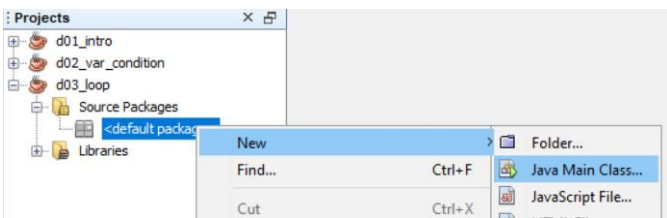
1. Viết chương trình có cấu trúc vòng lặp: for, while, do-while
2. Tạo class và sử dụng biến đối tượng.

### Bài tập 1:

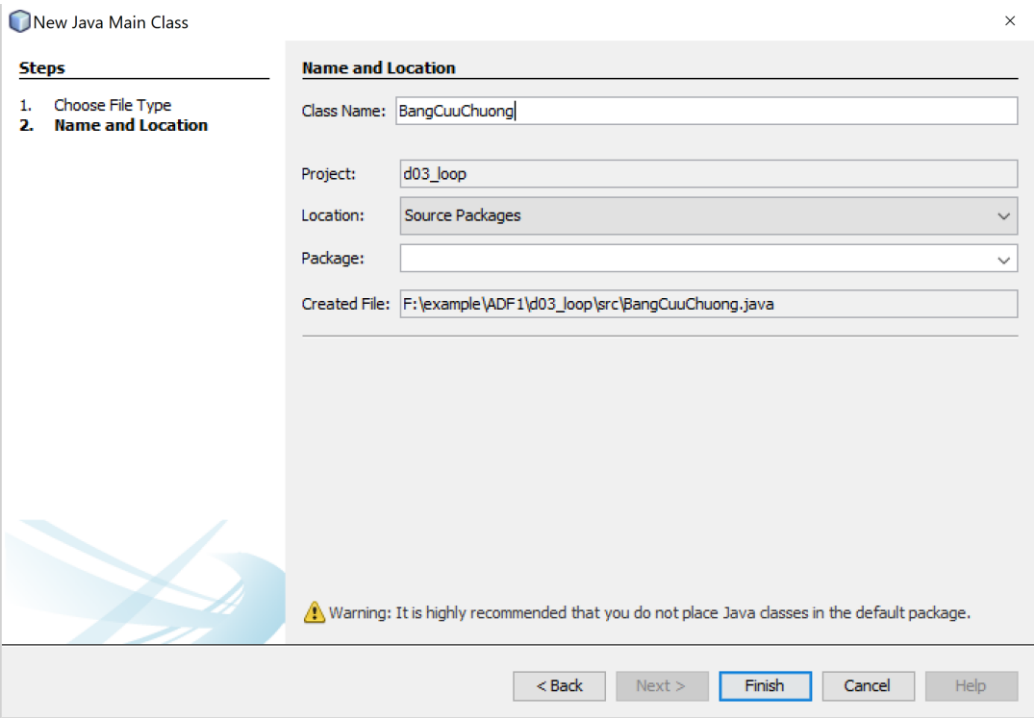
- Lập trình vòng lặp: sử dụng cấu trúc while.
- Viết chương trình in bảng cửu chương N.

HD thực hiện:

- Mở Netbean, Tạo project Java Application, đặt tên **d03\_loop** (nhớ bỏ check **Create Main Class**)
- Trong project **d03\_loop**, tạo class **BangCuuChuong**

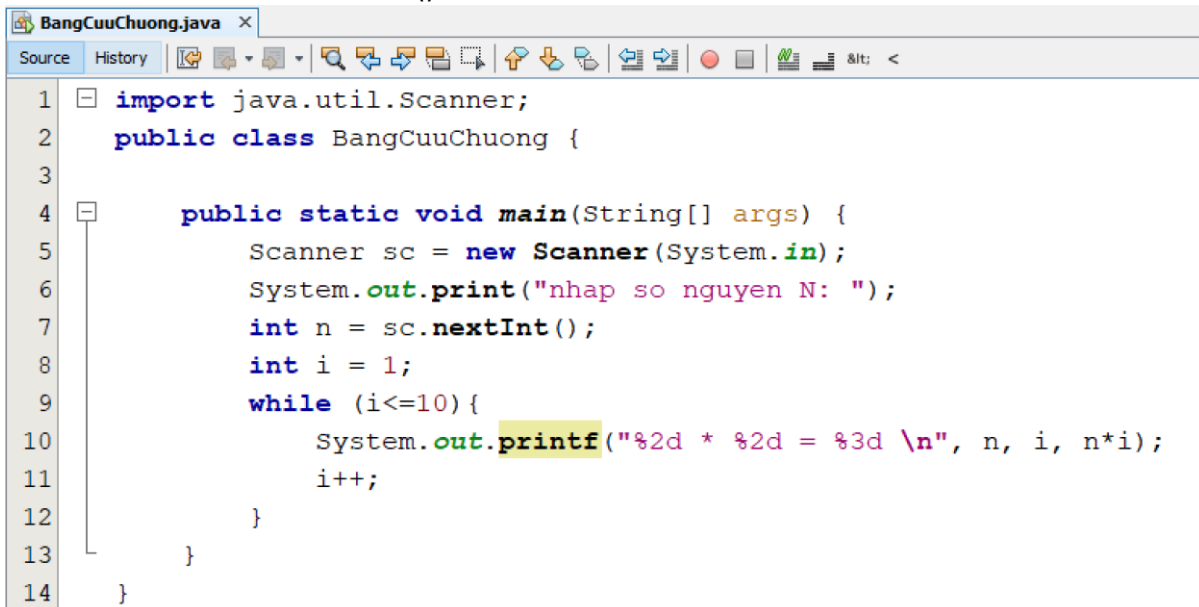


- Nhấp chuột phải trên **<default package>**, chọn **New**, chọn **Java Main Class**, gõ tên Class Name: **BangCuuChuong**, bấm **[Finish]**



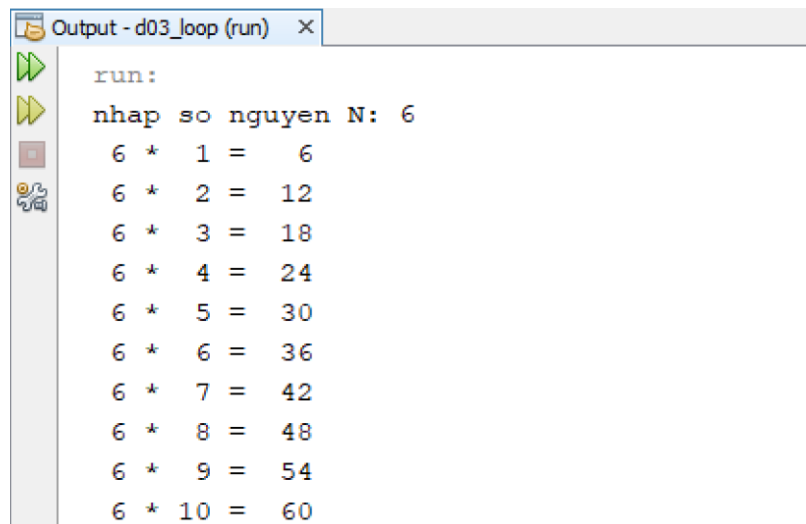
## LAB GUIDE 3 – Looping Constructs – Classes and Objects

- Viết code cho hàm main() như sau:



```
1 import java.util.Scanner;
2 public class BangCuuChuong {
3
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6         System.out.print("nhap so nguyen N: ");
7         int n = sc.nextInt();
8         int i = 1;
9         while (i<=10){
10             System.out.printf("%2d * %2d = %3d \n", n, i, n*i);
11             i++;
12         }
13     }
14 }
```

- Bấm Shift+F6 để chạy thử chương trình, kết quả sẽ xuất hiện trong màn hình **Output**



```
run:
nhap so nguyen N: 6
 6 * 1 = 6
 6 * 2 = 12
 6 * 3 = 18
 6 * 4 = 24
 6 * 5 = 30
 6 * 6 = 36
 6 * 7 = 42
 6 * 8 = 48
 6 * 9 = 54
 6 * 10 = 60
```

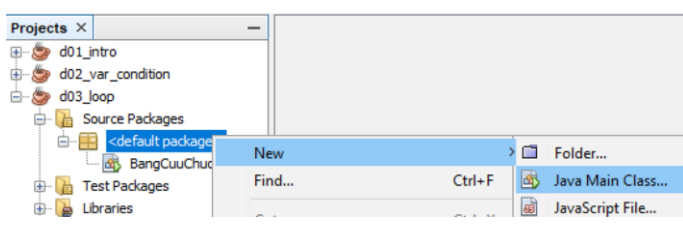
## LAB GUIDE 3 – Looping Constructs – Classes and Objects

### Bài tập 2:

- Lập trình vòng lặp, sử dụng cấu trúc do-while.
- Viết chương trình tính tổng của 10 số nguyên đầu tiên

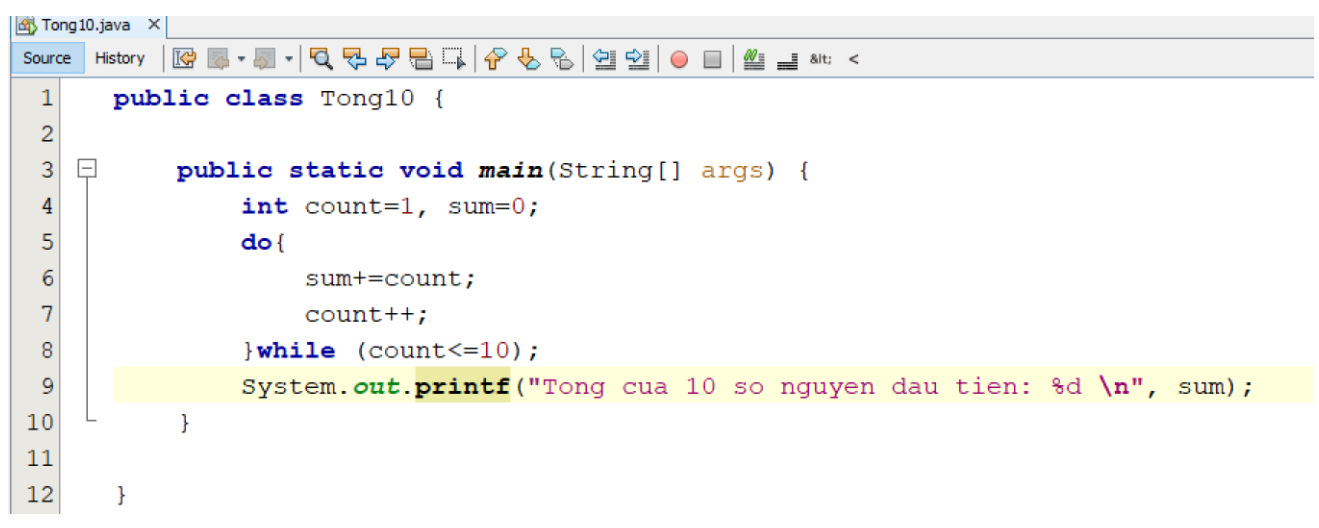
HD thực hiện:

- Trong project **d03\_loop**, tạo class **Tong10**



- Nhấp chuột phải trên "<default package>", chọn **New**, chọn **Java Main Class**, gõ tên Class Name: **Tong10**, bấm **[Finish]**

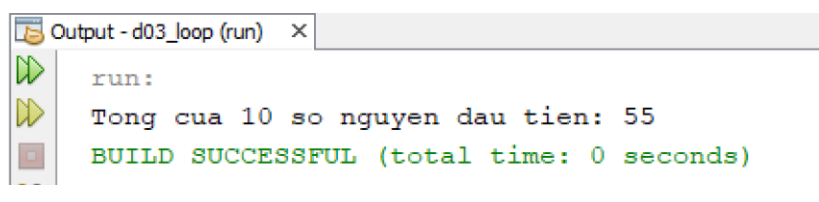
- Viết code cho hàm main() như sau:



```

1  public class Tong10 {
2
3      public static void main(String[] args) {
4          int count=1, sum=0;
5          do{
6              sum+=count;
7              count++;
8          }while (count<=10);
9          System.out.printf("Tong cua 10 so nguyen dau tien: %d \n", sum);
10     }
11
12 }
```

- Bấm Shift+F6 để chạy thử chương trình. Kết quả sẽ xuất hiện trong màn hình **Output**



```

run:
Tong cua 10 so nguyen dau tien: 55
BUILD SUCCESSFUL (total time: 0 seconds)
```

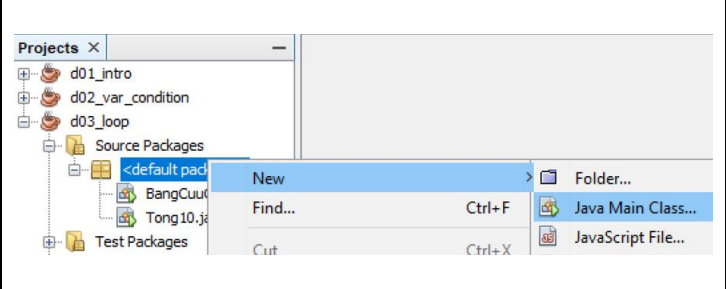
## LAB GUIDE 3 – Looping Constructs – Classes and Objects

### Bài tập 3:

- Lập trình vòng lặp, sử dụng cấu trúc for.
- Viết chương trình tính giai thừa của 1 số nguyên N.

HD thực hiện:

- Trong project **d03\_loop**, tạo class **GiaiThua**

	<ul style="list-style-type: none"> <li>- Nhấp chuột phải trên "&lt;default package&gt;", chọn <b>New</b>, chọn <b>Java Main Class</b>, gõ tên Class Name: <b>GiaiThua</b>, bấm [Finish]</li> </ul>
---	--

- Viết code cho hàm main() như sau:

--

- Bấm Shift+F6 để chạy thử chương trình

<p>Application Development Fundamentals-I</p>
---

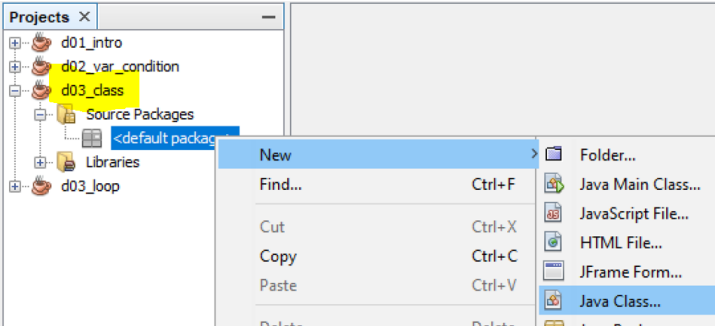
## LAB GUIDE 3 – Looping Constructs – Classes and Objects

### Bài tập 4:

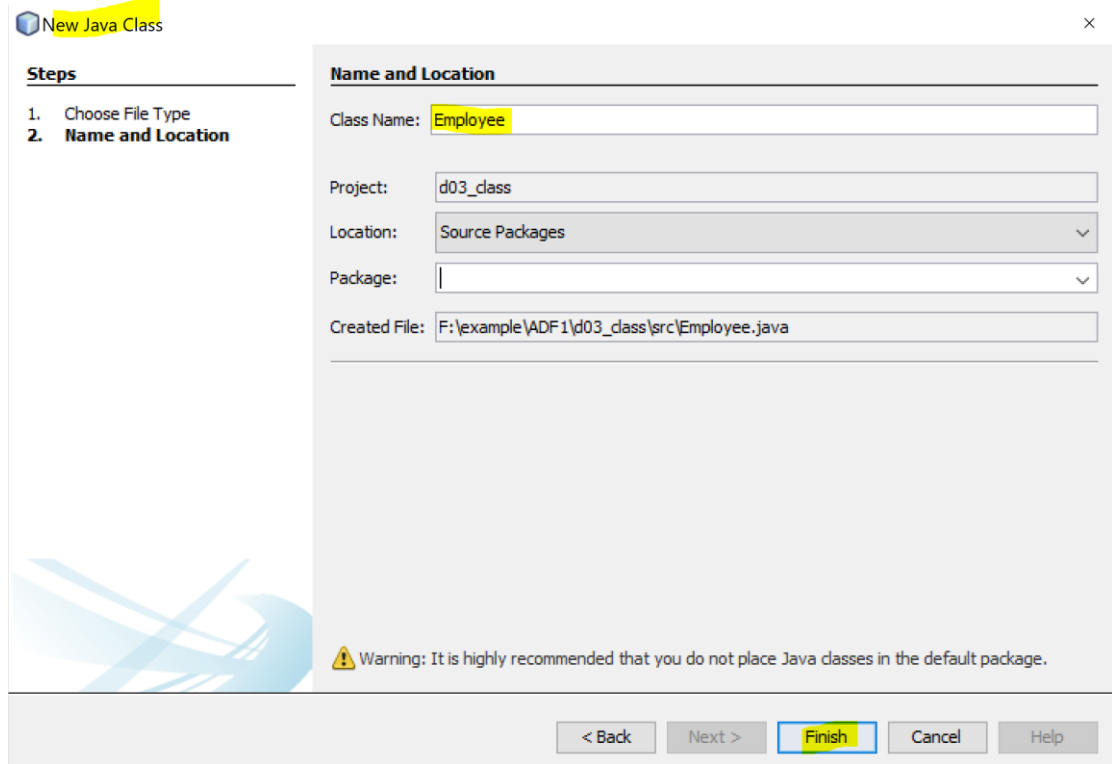
- Định nghĩa class, khai báo và sử dụng biến object.
- Viết chương trình nhập thông tin của 1 nhân viên, in ra bảng lương của nhân viên này. Thông tin nhân viên sẽ được mô tả trong class **Employee**, bao gồm các :  
 . **thuộc tính** : id, name, baseSalary, workDays  
 . **phương thức** : input() – nhập thông tin nhân viên, output() – xuất bảng lương.

HD thực hiện:

- Mở Netbean, Tạo project Java Application, đặt tên **d03\_class** (nhớ bỏ check **Create Main Class**)
- Trong project **d03\_class**, tạo class **Employee**



- Nhấp chuột phải trên **<default package>**, chọn **New**, chọn **Java Class**, gõ tên Class Name: **Employee**, bấm **[Finish]**



## LAB GUIDE 3 – Looping Constructs – Classes and Objects

- Viết code class **Employee** như sau:

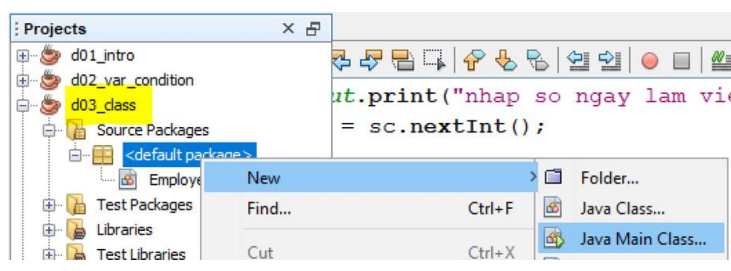
```

Employee.java x
Source History
1 import java.util.Scanner;
2
3 public class Employee {
4
5     //thuoc tinh cua nhan vien
6     public String id, name;
7     public int baseSalary; // luong CB
8     public int workDays; // so ngay lam viec trong thang
9
10    //ham dung ko tham so
11    public Employee() {
12        id = "E01";
13        name = "Nguyen A";
14        baseSalary = 500;
15        workDays = 26;
16    }
17
18    //ham dung co tham so
19    public Employee(String id, String name, int baseSalary, int workDays) {
20        this.id = id;
21        this.name = name;
22        this.baseSalary = baseSalary;
23        this.workDays = workDays;
24    }
25
26    //ham nhap thong tin nhan vien
27    public void input() {
28        Scanner sc = new Scanner(System.in);
29        System.out.print("nhap id: ");
30        id = sc.nextLine();
31        System.out.print("nhap ten: ");
32        name = sc.nextLine();
33        System.out.print("nhap luong CB: ");
34        baseSalary = sc.nextInt();
35        System.out.print("nhap so ngay lam viec: ");
36        workDays = sc.nextInt();
37    }
38
39    // ham xuat bang luong chi tiet
40    public void output() {
41        System.out.println("Bang luong nhan vien");
42        System.out.println(" id: " + id);
43        System.out.println(" ten: " + name);
44        System.out.println(" luong CB: " + baseSalary);
45        System.out.println(" so ngay lam viec: " + workDays);
46
47        float income = baseSalary * workDays / 26.0f ;
48        System.out.printf(" luong thuc lanh : %.2f \n", income);
49    }
50 }

```

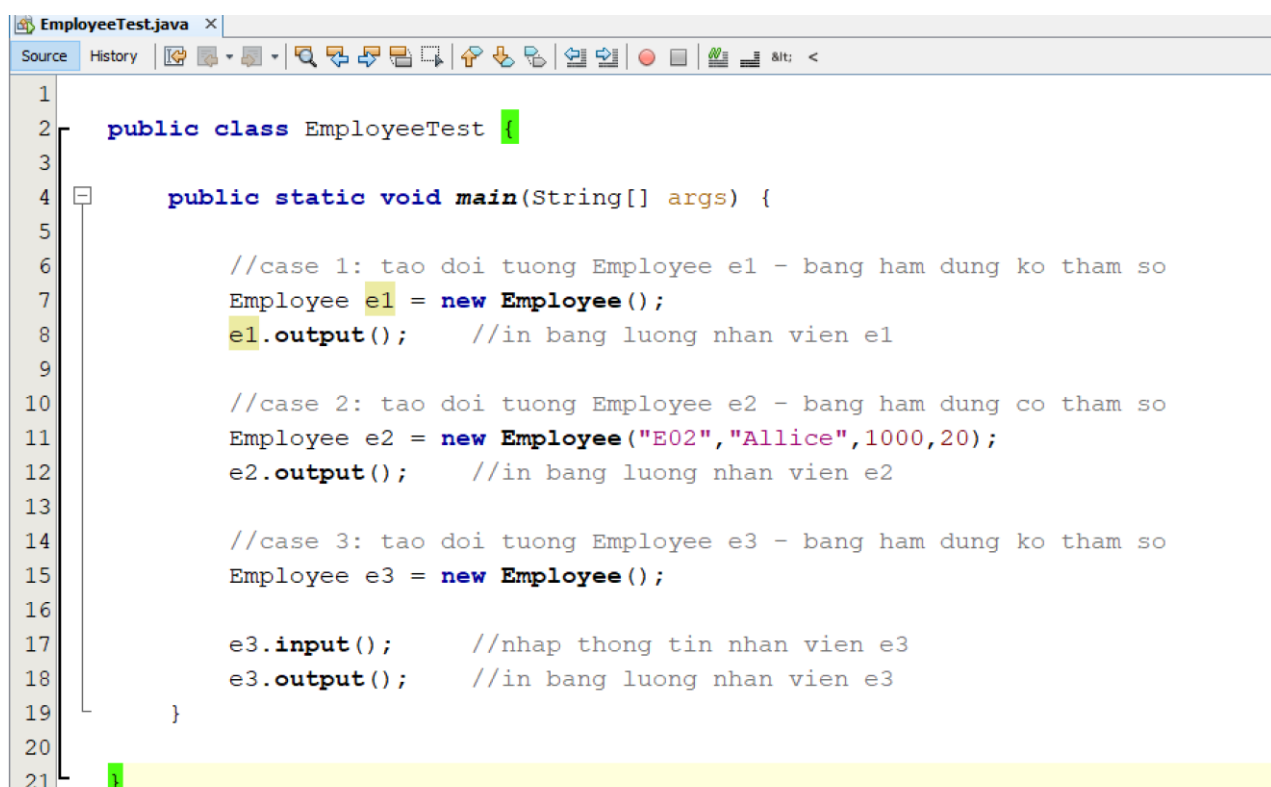
## LAB GUIDE 3 – Looping Constructs – Classes and Objects

- Trong project **d03\_class**, tiếp tục tạo java main class **EmployeeTest**, để tạo đối tượng Employee và kiểm thử các chức năng nhập thông tin và in bảng lương



- Nhấp chuột phải trên "<default package>", chọn **New**, chọn **Java Main Class**, gõ tên Class Name: **EmployeeTest**, bấm [Finish]

- Viết code cho hàm main() như sau:

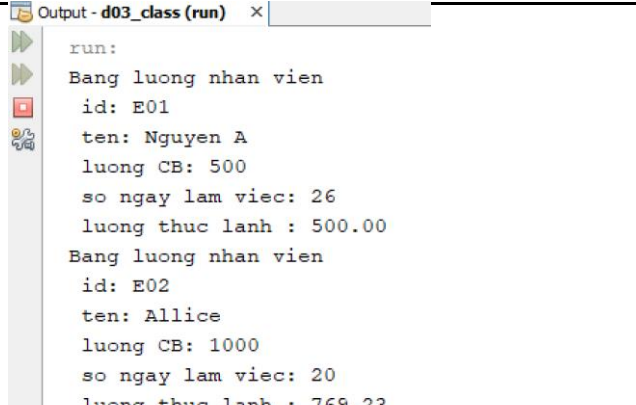


```

1
2 public class EmployeeTest {
3
4     public static void main(String[] args) {
5
6         //case 1: tao doi tuong Employee e1 - bang ham dung ko tham so
7         Employee e1 = new Employee();
8         e1.output();    //in bang luong nhan vien e1
9
10        //case 2: tao doi tuong Employee e2 - bang ham dung co tham so
11        Employee e2 = new Employee("E02", "Allice", 1000, 20);
12        e2.output();    //in bang luong nhan vien e2
13
14        //case 3: tao doi tuong Employee e3 - bang ham dung ko tham so
15        Employee e3 = new Employee();
16
17        e3.input();      //nhap thong tin nhan vien e3
18        e3.output();     //in bang luong nhan vien e3
19    }
20
21 }

```

- Bấm Shift+F6 để chạy thử chương trình.



```

nhap id: E04
nhap ten: Jolie
nhap luong CB: 1200
nhap so ngay lam viec: 20
Bang luong nhan vien
id: E04
ten: Jolie
luong CB: 1200
so ngay lam viec: 20
luong thuc lanh : 923.08
BUILD SUCCESSFUL (total time:

```

## LAB GUIDE 3 – Looping Constructs – Classes and Objects

### Assignment:

Viết ứng dụng Java quản lý 1 tài khoản ngân hàng, cho phép thực hiện các giao dịch gửi, rút tiền và xem số dư tài khoản theo hướng dẫn sau:

1. Tạo project **ass2**:
2. Tạo Java Class, **Account** (**Account.java**) với cấu trúc sau:
  - a. Fields dữ liệu : **accNo**, **balance**
  - b. Hàm dựng: khởi trị ban đầu cho các field dữ liệu.
    - i. Mặc định **Account()** – no-parameter constructor ( dòng 17-20 )
    - ii. Có tham số **Account(String, int)** - parameterized constructor (dòng 22-25)
  - c. Phương thức:
    - i. void **deposit(int)** : thực hiện chức năng gửi tiền vô tài khoản
    - ii. void **withdraw(int)** : thực hiện chức năng rút tiền.
    - iii. int **getBalance()** : trả về số dư tài khoản.

<pre> 12 public class Account { 13 14     String accNo; 15     int balance; 16 17     public Account() { 18         accNo = "ACC01"; 19         balance = 100; 20     } 21 22     public Account(String acc, int bal) { 23         accNo = acc; 24         balance = bal; 25     } 26 27     public void deposit(int amt) { 28         if (amt &gt; 0) { 29             balance += amt; 30             System.out.println("Giao dịch thanh công."); 31         } else { 32             System.out.println("Ko the goi tien am !..."); 33         } 34     } </pre>	<pre> 35 36     public void withdraw(int amt) { 37         if (amt &gt; 0 &amp;&amp; amt &lt; balance) { 38             //cập nhật số dư 39             //tb giao dịch thành công 40         } else { 41             //tb giao dịch thất bại 42         } 43     } 44 45     public int getBalance() { 46         return balance; 47     } 48 49     @Override 50     public String toString() { 51         return String.format("%s : %d", accNo, balance); 52     } 53 54 55 } </pre>
--	---

1. Tạo lớp kiểm thử Java Main Class, **AccountTest.java** , bao gồm 2 phương thức :
  - a. **menu()** : tạo thực đơn thực hiện chức năng gửi và rút tiền.
  - b. **main()** : gọi phương thức menu() để thi hành chương trình.



## LAB GUIDE 3 – Looping Constructs – Classes and Objects

```

11 void menu() {
12     Scanner in = new Scanner(System.in);
13     Account a = new Account();
14     int option = 0, amt = 0;
15
16     do {
17         System.out.println("He thong rut/goi tien tu dong: ");
18         System.out.println("1. goi tien ");
19         System.out.println("2. rut tien ");
20         System.out.println("3. xem so du ");
21         System.out.println("4. thoát ");
22         System.out.print("\t Chon chuc nang ( 1 - 4 ) : ");
23
24         option = in.nextInt();
25         switch (option) {
26             case 1:
27                 System.out.print("Nhap so tien muon goi :");
28                 amt = in.nextInt();
29                 a.deposit(amt);
30                 break;
31             case 2:
32                 //thông báo yêu cầu nhập số tiền muốn rút
33                 //gọi phương thức rút tiền
34                 break;
35             case 3:
36                 System.out.printf("So du = %d\n", a.getBalance());
37                 break;
38         }
39     } while (option != 4);
40
41     System.out.println("Cam on da su dung he thong ! Bye Bye ...");
42 }

```

```

43
44 public static void main(String[] args) {
45     AccountTest o = new AccountTest();
46     o.menu();
47 }

```

### 2. Biên dịch và thi hành chương trình.

### 3. Mở rộng :

#### a. Trong lớp **Account**, sửa lại các phương thức :

- i. deposit(int amt) : điều kiện giao dịch thành công : số tiền gửi phải > 100.
- ii. withdraw(int amt) : điều kiện giao dịch thành công : số tiền muốn rút > 100 và < số dư TK.

#### b. Trong lớp **AccountTest**:

- i. Khi khởi tạo đối tượng acc trong phương thức menu() – cho phép user nhập vào mã TK và số dư ban đầu của TK.
- ii. Khi in ra số dư TK của khách hàng, xuất thêm mã TK.