

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

- Tạo lớp trừu tượng.
- Lập trình kế thừa từ lớp cha trừu tượng

Bài tập:

Viết chương trình tính diện tích và chu vi của hình chữ nhật và hình tròn.

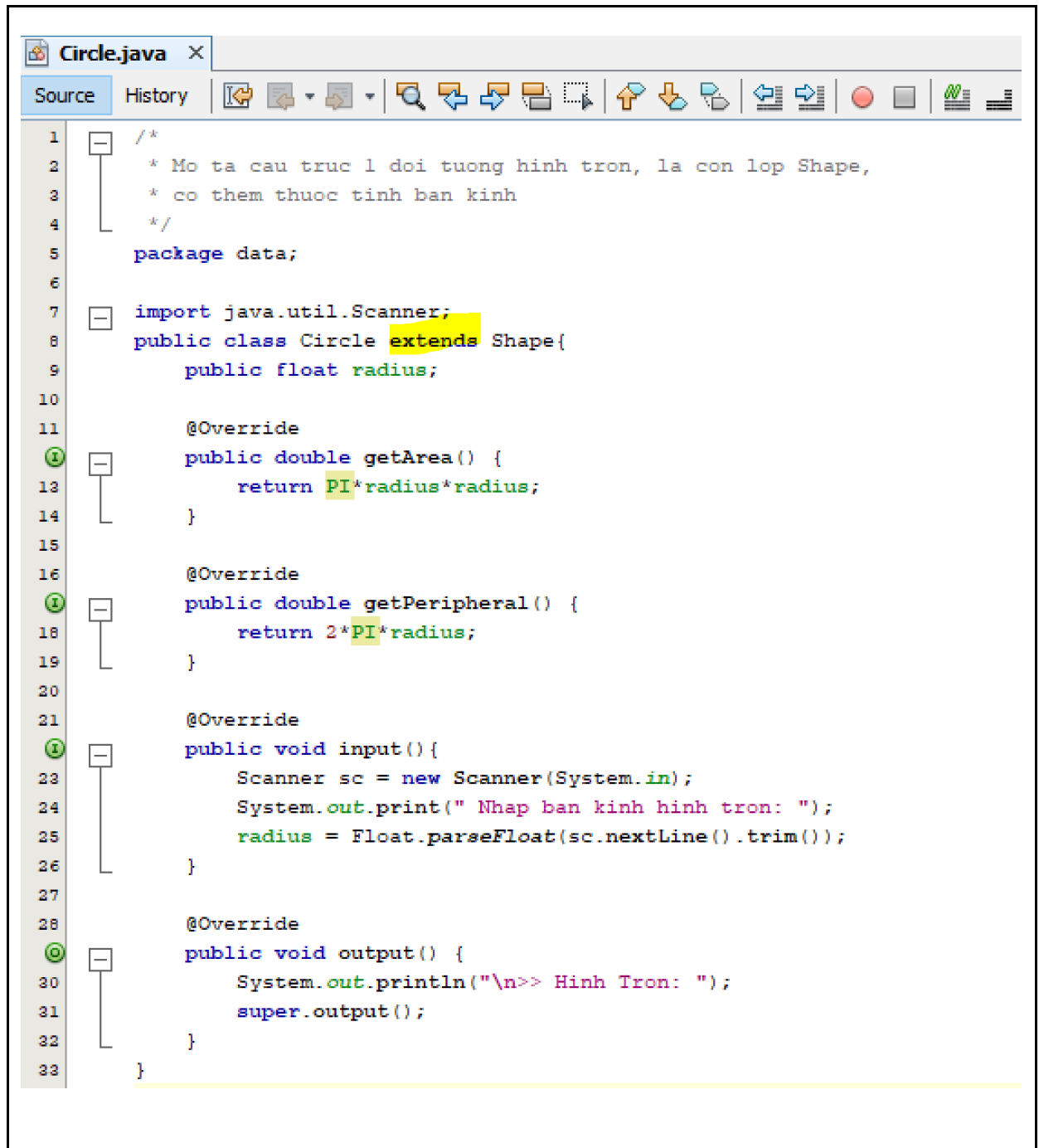
HD thực hiện:

- Mở Netbean, Tạo **project Java Application**, đặt tên **d06_abstract** (nhớ bỏ check **Create Main Class**)
- Trong project **d06_abstract**, tạo **package data**
- Trong **package data**, tạo **Java Class Shape**
- Viết code cho class **Shape**

```
Shape.java x
Source History
1  /*
2   * Đây là 1 lớp trừu tượng mô tả cấu trúc 1 đối tượng hình học, bao gồm:
3   * - 1 thuộc tính : PI
4   * - 3 hàm trừu tượng: tính diện tích, tính chu vi và hàm nhập liệu
5   * - 1 hàm xuất diện tích và chu vi của đối tượng hình học
6   */
7   package data;
8
9   public abstract class Shape {
10       public double PI = 3.14;
11
12       //hàm trừu tượng tính diện tích
13       public abstract double getArea();
14
15       //hàm trừu tượng tính chu vi
16       public abstract double getPeripheral();
17
18       //hàm trừu tượng nhập dữ liệu cho 1 đối tượng hình học
19       public abstract void input();
20
21       //hàm in ra diện tích và chu vi của 1 đối tượng hình học
22       public void output() {
23           System.out.printf("Diện tích: %.2f \n", getArea());
24           System.out.printf("Chu vi: %.2f \n", getPeripheral());
25       }
26   }
```

LAB GUIDE 8 – Abstract class

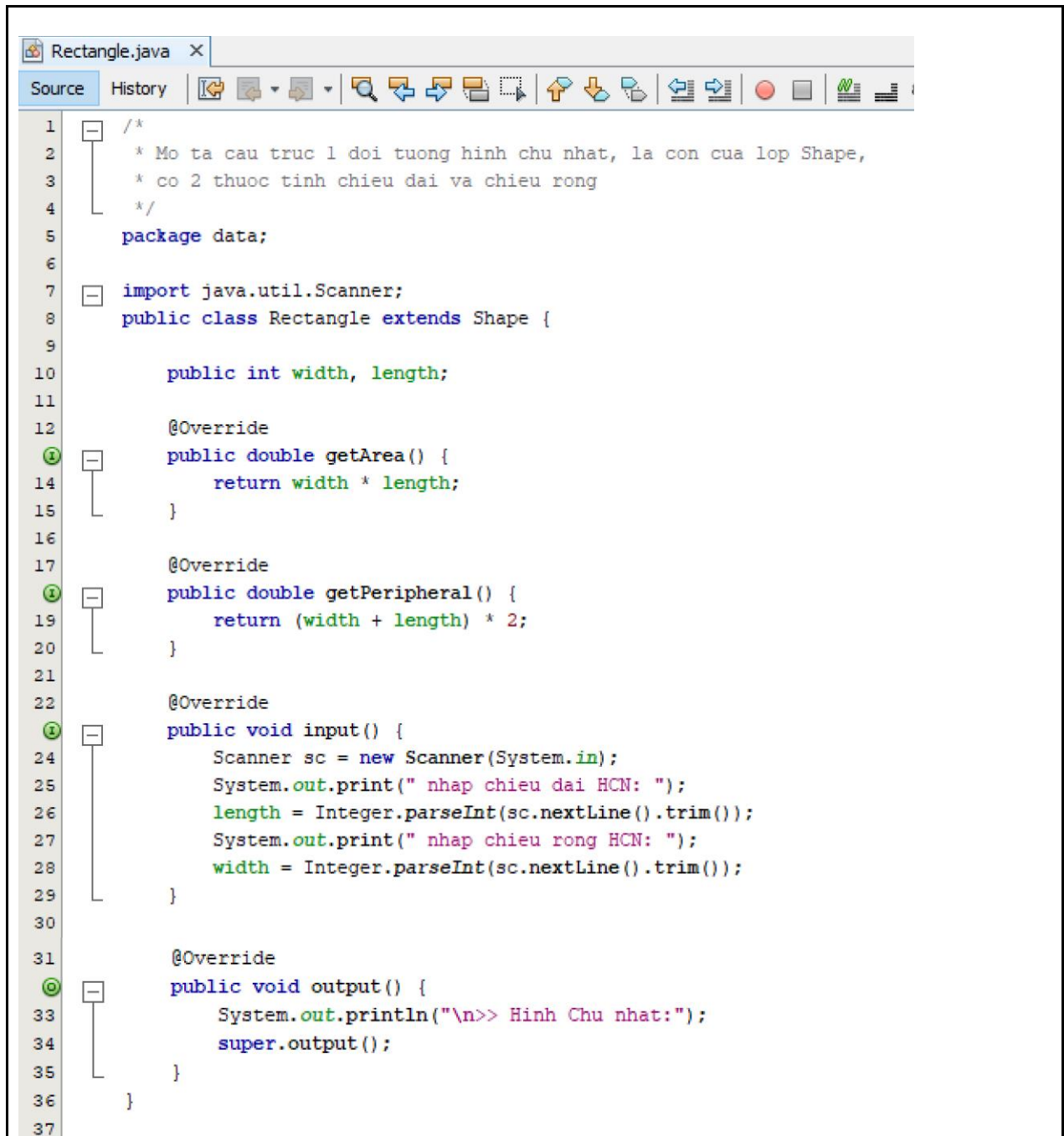
- Trong package **data**, tạo Java Class **Circle**, là lớp con của lớp **Shape**
- Viết code cho class **Circle**, override các phương thức **getArea()**, **getPeripheral()**, **input()**, **output()** của lớp cha trừu tượng **Shape**



```
1  /*
2   * Mô tả cấu trúc 1 đối tượng hình tròn, là con lớp Shape,
3   * có thêm thuộc tính bán kính
4   */
5   package data;
6
7   import java.util.Scanner;
8   public class Circle extends Shape{
9       public float radius;
10
11       @Override
12       public double getArea() {
13           return PI*radius*radius;
14       }
15
16       @Override
17       public double getPeripheral() {
18           return 2*PI*radius;
19       }
20
21       @Override
22       public void input() {
23           Scanner sc = new Scanner(System.in);
24           System.out.print(" Nhập bán kính hình tròn: ");
25           radius = Float.parseFloat(sc.nextLine().trim());
26       }
27
28       @Override
29       public void output() {
30           System.out.println("\n>> Hình Tròn: ");
31           super.output();
32       }
33   }
```

LAB GUIDE 8 – Abstract class

- Trong package **data**, tạo Java Class **Rectangle**, là lớp con của lớp **Shape**
- Viết code cho class **Rectangle**, override các phương thức **getArea()**, **getPeripheral()**, **input()**, **output()** của lớp cha trừu tượng **Shape**



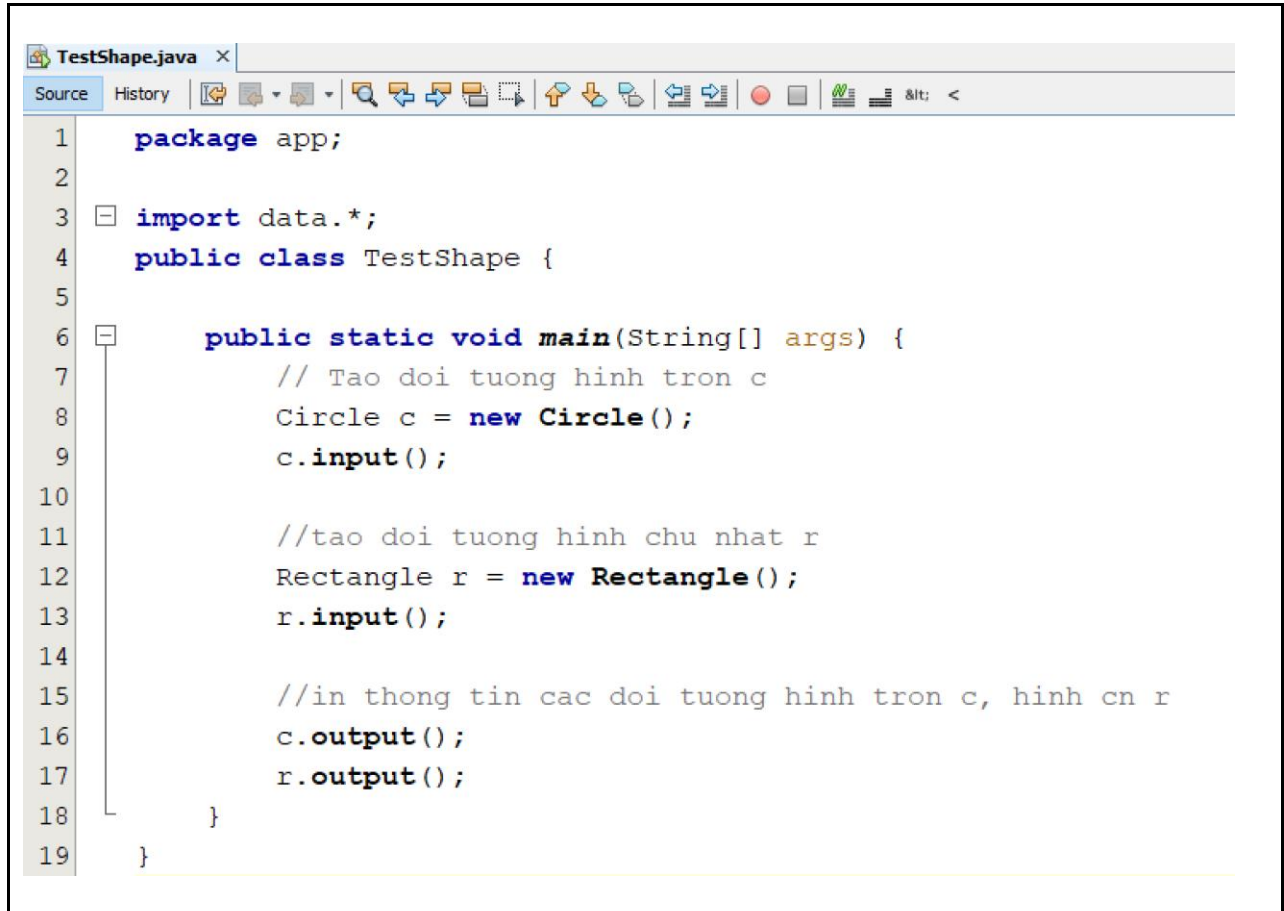
```

1  /*
2   * Mô tả cấu trúc 1 đối tượng hình chữ nhật, là con của lớp Shape,
3   * có 2 thuộc tính chiều dài và chiều rộng
4   */
5  package data;
6
7  import java.util.Scanner;
8  public class Rectangle extends Shape {
9
10     public int width, length;
11
12     @Override
13     public double getArea() {
14         return width * length;
15     }
16
17     @Override
18     public double getPeripheral() {
19         return (width + length) * 2;
20     }
21
22     @Override
23     public void input() {
24         Scanner sc = new Scanner(System.in);
25         System.out.print("nhập chiều dài HCN: ");
26         length = Integer.parseInt(sc.nextLine().trim());
27         System.out.print("nhập chiều rộng HCN: ");
28         width = Integer.parseInt(sc.nextLine().trim());
29     }
30
31     @Override
32     public void output() {
33         System.out.println("\n>> Hình Chữ nhật:");
34         super.output();
35     }
36 }
37

```

LAB GUIDE 8 – Abstract class

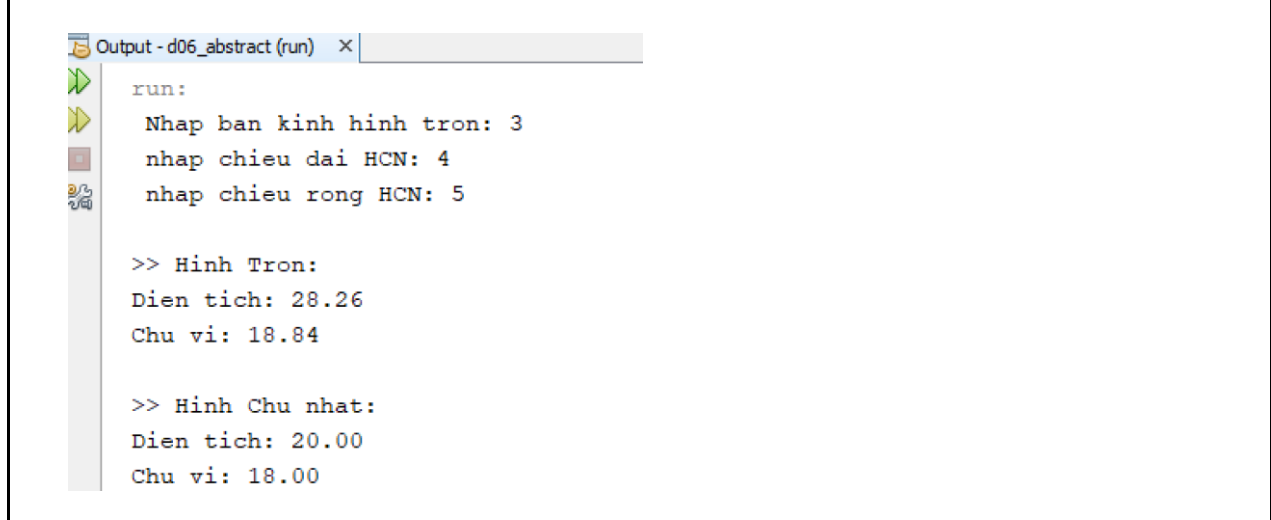
- Tạo java main class **TestShape**, trong package **app**, để kiểm thử các lớp Circle, Rectangle
- Viết code cho class **TestShape**



```

1  package app;
2
3  import data.*;
4  public class TestShape {
5
6      public static void main(String[] args) {
7          // Tao doi tuong hinh tron c
8          Circle c = new Circle();
9          c.input();
10
11         //tao doi tuong hinh chu nhat r
12         Rectangle r = new Rectangle();
13         r.input();
14
15         //in thong tin cac doi tuong hinh tron c, hinh cn r
16         c.output();
17         r.output();
18     }
19 }
  
```

- Bấm Shift+F6 để biên dịch và chạy thử chương trình



```

Output - d06_abstract (run) x
run:
Nhap ban kinh hinh tron: 3
nhap chieu dai HCN: 4
nhap chieu rong HCN: 5

>> Hình Tron:
Dien tích: 28.26
Chu vi: 18.84

>> Hình Chu nhat:
Dien tích: 20.00
Chu vi: 18.00
  
```

Assignment

Write a Java application - **Inventory System** - to manage the list of computers with the specification as follows:

1	<p>Creates an abstract class named Computer in package Goods.</p> <ul style="list-style-type: none"> - Fields ID, name, price - Default constructor to initialize the above fields. - Method <ul style="list-style-type: none"> - void accept() : allow user input data into data fields. Made validation for : ID, name is not null. Price must be greater than zero. - void printInfo() : this is an abstract method used to print details of an product.
2	<p>Create class Laptop derives from Computer, in package Goods, consists of:</p> <ul style="list-style-type: none"> - Field webcam (y/n), QoH (số lượng tồn kho) - Constructors to initialize the all fields. - Override methods: <ul style="list-style-type: none"> - accept() : allow user to input details for a laptop: Recall method accept() of super class. - printInfo() : display details of an laptop. - toString() : return a string presenting all the details of a product as follows: id, name, webcam, price, QoH, amount (= price*QoH)
4	<p>Create class LaptopCatalog in package Goods for managing a collection of Computers</p> <ul style="list-style-type: none"> - Fields: max, cnt, cList – array of laptops No-arg constructor to initialize the all the fields. - Methods: <ul style="list-style-type: none"> - addLaptop() - add a new laptop into array - search (String laptopName) - search laptops by the name accepted by user. - displayAll() - display all computers. - remove(String laptopID): remove laptop by id
5	<p>Create main class Inventory in package Application that allows user to manage the laptops accepted into system through the menu system as follows:</p> <ol style="list-style-type: none"> 1. Add a new laptop 2. Search laptop by name 3. Display all laptops 4. Remove a laptop by id 5. Exit