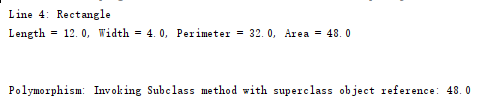
Lab 3: Abstract Classes, Interface, Composition in Java1. Abstract methods and Abstract Classes

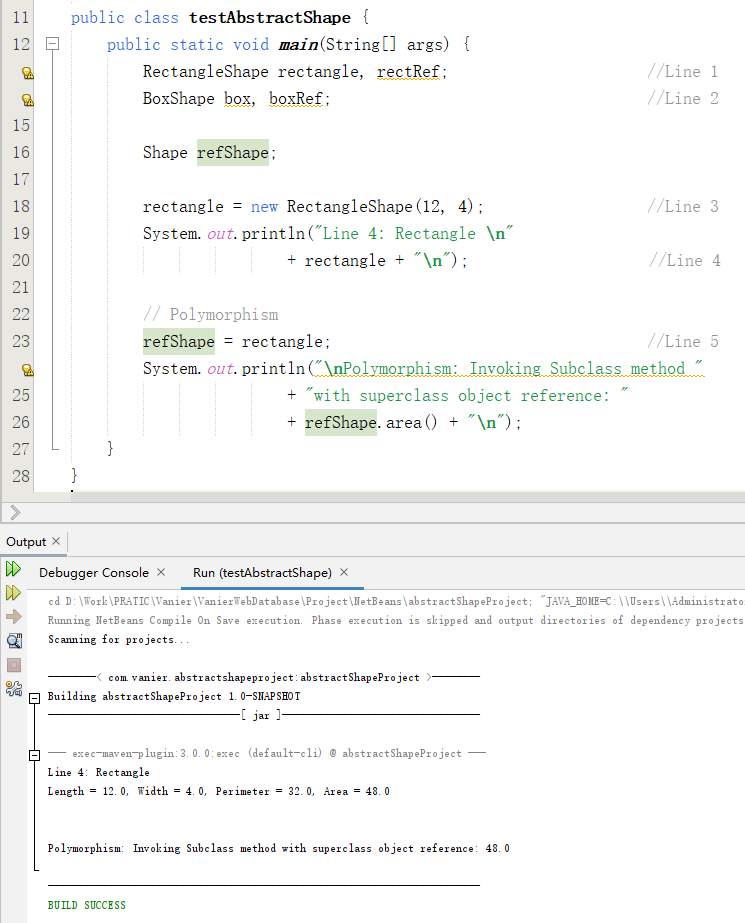
Shape.java

1. **public** **abstract** **class** Shape {
2. //area is overloading method in BoxShape and RectangleShape
3. **public** **abstract** **double** area();
4. }

testAbstractShape.java

1. **public** **class** testAbstractShape {
2. **public** **static** **void** main(String[] args) {
3. RectangleShape rectangle, rectRef;                      //Line 1
4. BoxShape box, boxRef;                                   //Line 2
6. Shape refShape;
8. rectangle = **new** RectangleShape(12, 4);                  //Line 3
9. System.out.println("Line 4: Rectangle \n"
10. + rectangle + "\n");                    //Line 4
11. // Polymorphism
12. refShape = rectangle;                                   //Line 5
13. System.out.println("\nPolymorphism: Invoking Subclass method "
14. + "with superclass object reference: "
15. + refShape.area() + "\n");
16. }
17. }





2. Interface Classes