

```
1 #include <iostream>
2 #include "Shapes.h"
3
4
5 int main() {
6     // Trying to declare this will give an error
7     // because it's not allowed to instantiate an
8     // object of an abstract class.
9
10    Rectangle rect(3 , 4);
11    Square q(3);
12    Circle c(3);
13
14    std::cout << "Rectangle: [" << rect.height() << " , " << rect.width() <<
15        "]" << std::endl;
16    std::cout << "Square : [" << q.height() << " , " << q.width() <<
17        "]" << std::endl;
18    std::cout << "Circle : [" << c.height() << " , " << c.width() <<
19        "]" << std::endl;
20
21    std::cout << "Area: " << '\n';
22    std::cout << "+ Rectangle : " << rect.area() << std::endl;
23    std::cout << "+ Square : " << q.area() << std::endl;
24    std::cout << "+ Circle : " << c.area() << '\n' << std::endl;
25
26    std::cout << "Perimeter: " << '\n';
27    std::cout << "+ Rectangle : " << rect.perimeter() << std::endl;
28    std::cout << "+ Square : " << q.perimeter() << std::endl;
29    std::cout << "+ Circle : " << c.perimeter() << '\n' << std::endl;
30
31    std::cout << "Height and width: " << '\n';
32    std::cout << "+ Rectangle : " << rect.height() << " , " << rect.width()
33        << std::endl;
34    std::cout << "+ Square : " << q.height() << " , " << q.width()
35        << std::endl;
36    std::cout << "+ Circle : " << c.height() << " , " << c.width()
37        << '\n' << std::endl;
38
39    // Test rotation
40    Rectangle r(2 , 7);
41    std::cout << "Rectangle:" << std::endl;
42    std::cout << "[" << r.height() << " , " << r.width() << "]" rotation -> ";
43
44    r.rotate();
45
46    std::cout << "[" << r.height() << " , " << r.width() << "]" << std::endl;
47
48    std::cout << "Square:" << std::endl;
49    std::cout << "[" << q.height() << " , " << q.width() << "]" rotation -> ";
50
51    q.rotate();
52
53    std::cout << "[" << q.height() << " , " << q.width() << "]" << std::endl;
54
55
```

```
56
57     std::cout << "Circle:" << std::endl;
58     std::cout << "[" << c.height() << " , " << c.width() << "]" rotation -> ";
59
60     c.rotate();
61
62     std::cout << "[" << c.height() << " , " << c.width() << "]" << std::endl;
63
64     return 0;
65 }
```