Classes Test 1

Look at the following class definition.

public class MyClass  
 {  
 private int num;  
  
 public MyClass(int n)  
 {  
 num = n;  
 }  
 public int getNum()  
 {  
 return num;  
 }  
 public void setNum(int n)  
 {  
 num = n;  
 }

public String toString()  
 {  
 return "Num = " + num;  
 }  
 }

1. Which of the following is the constructor method for MyClass?
2. MyClass(int n)
3. getNum()
4. setNum(int n)
5. MyClass does not have a constructor.
6. Which of the following is the accessor method for MyClass?
7. MyClass(int n)
8. getNum()
9. setNum(int n)
10. MyClass does not have an accessor method.
11. Which of the following is the mutator method for myClass?
12. MyClass(int n)
13. getNum()
14. setNum(int n)
15. MyClass does not have a mutator method.
16. Which of the following correctly instantiates a MyClass object and initializes its instance variable a value of 5?
17. num = 5
18. MyClass obj = new MyClass(5)
19. MyClass obj = 5
20. obj.num = 5
21. Assume that a MyClass object named obj exists. Which of the following will correctly print the value of obj’s instance variable from a client class?
22. System.out.println(num)
23. System.out.println(obj.num)
24. System.out.println(obj.getNum())
25. More than one of these will.
26. Assume that a MyClass object named obj exists. Which of the following will correctly modify the value of obj’s instance variable from a client class?
27. num = 5
28. obj.num = 5
29. obj.setNum(5)
30. More than one of these will.
31. Assume that a MyClass object named obj exists and its instance variable num has been assigned the value 5. Which of the following will correctly print the instance variable num from a client class in the following format: Num = 5
32. System.out.println("Num = " + obj.getNum());
33. System.out.println(obj.toString());
34. System.out.println(obj);
35. All of the above
36. Suppose you wanted to add a new method to MyClass named **square** that returned the square of the instance variable num. Which of the following can replace /\* return type \*/ and /\* expression \*/ so that the method works as intended.

public / \* return type \*/ square()  
{  
 /\* expression \*/  
}

/\* return type \*/ /\* expression \*/

1. void return num \* num;
2. int return num \* num;
3. void System.out.println(num \* num);
4. int System.out.println(num \* num);