

1. b

Reason: Compiler will look for the main function to start the program and it will fail as no main function is there.

2. b

Reason: Instance was not created for the class but tried to make a call to the instance method

3. a

Reason: Executed properly as the instance was created in the main method and after that the instance method was invoked.

4. b

Reason: Private method of a class can't be invoked from another class even if an insurance is created containing that private method.

5. b

Rason: Program looks for myMethod method definition inside AnotherClass which is unavailable and gives compile time error.

6. b

Reason: Assuming both the classes are in the same package and will have package level access. So, each class instance will be able to call their methods having public and package level (or default) access modifiers.

7. b

Reason: As a is a local variable inside main, other instance methods of that class don't have access to that local variable.

8. a

Reason: As the input is positive integer and in every function call it's been reduced by 1, it will be 0 or 1 after a few cycles. Then it will just print "hello" and terminate the program to break the cyclic call.

9. c

Reason: As float and double value as some precision limitation, the behavior varies while we add or subtract the value to/from another value. Thats which even though adding 0.1 with 0.7 and subtracting 0.1 from 0.9 should give the equality by theory, that is not the case and it gets equal after 3 iteration as per the given calculation and logic.