Name: _____Answer Key_____ Date: _____ Section: _____

Write T/F in the blank next to the question:

1.
__T__ Each strategy is a separate class that follows a common interface; We can change the strategy used without changing the code that uses it

2.
__F__ Different strategies have their own interface that they follow, making them hard to interchange within the application

3.
__F__ The Strategy pattern is a design approach used to enable an object to switch between different algorithms or strategies at compile time

4.
__F__ The Strategy pattern is useful for situations where there isn't multiple ways to perform a task, and the choice of method depends on the circumstances at runtime

5.
__T__ Different compression tools like ZIP or RAR is an example of the Strategy pattern

6.
__F__ A con for Strategy pattern is that it makes unit testing harder

7.
__T__ The client chooses a concrete strategy and passes it to the context

8.
__T__ It is easy to expand and implement different strategies for a common interface

9.
__T__ Strategies provide different ways of executing the same thing

10.
__F__ The Strategy pattern leads to a hierarchy of classes that makes things unnecessary complex