The **Decorator pattern** **achieves a single objective of dynamically adding responsibilities to any object**. The decorator pattern applies when there is **a need to dynamically add as well as remove responsibilities** to a class, and **when sub-classing would be impossible due to the large number of subclasses** that could result.





