COMP303 A4

1. I applied composite design pattern for this part. Each simple filter, like LanguageFilter, is a leaf. The conjunction and disconjunction are composites. To create a composite, the client needs to pass in a list of elements of the type WatchListFilter. The class FirstEpisodeFilter returns true for every first episode, and returns false for other watchable objects.
2. The class EpisodeManager takes in a TVShow as input, and the class would manage the episodes inside this TVShow. The method setEpisode takes in a sequential number and clone the corresponding episode as a prototype. The method create() would clone an episode from prototype and return it. Since an new episode is created, the client should provide a new path. The episode returned would have the same information but different path from the original one.
3. Blackbox testing:

   testFirstEpisodeFilter()   testPSFilterStrategy()   testConjunctionOfFilter()

   testDisconjunctionOfFilter()   testCreate()

   Whitebox testing:

   testCloneEpisode()