DBSong

Method	Input	Case
UpdateSong	valid adminId(is admin), Song with valid data	success scenario
UpdateSong	invalid adminId, Song with valid data	fail scenario
UpdateSong	"valid" adminId(is not admin), Song with valid data	fail scenario
UpdateSong	valid adminId(is admin), Song with invalid data (i.e. Null in title)	fail scenario
GetSong	valid songId	success scenario
GetSong	invalid songId	fail scenario
GetSong	non existing songId	fail scenario
PurchaseSong	valid userId, valid songId, sufficient funds	success scenario
PurchaseSong	valid userId, valid songId, insufficient funds	fail scenario
PurchaseSong	invalid userId, songId, int	fail scenario
PurchaseSong	userId, invalid songId, int	fail scenario
RentSong	valid userId, valid songId, sufficient funds	success scenario
RentSong	valid userId, valid songId, insufficient funds	fail scenario
RentSong	invalid userId, songId, int	fail scenario
RentSong	userId, invalid songId, int	fail scenario
DeleteSong	valid adminId(is admin), valid songId	success scenario
DeleteSong	"valid" adminId(is not admin), songId	fail scenario
DeleteSong	invalid adminId, songId	fail scenario
DeleteSong	adminId, invalid songId	fail scenario
CreateSong	valid adminId(is admin), valid tmpId, genre array, Song with valid data	success scenario
CreateSong	"valid" adminId(is not admin), tmpId, genre array, Song	fail scenario
CreateSong	adminId, valid tmpId, genre array, Song with invalid data (i.e. Null in title)	fail scenario
CreateSong	Invalid adminId, valid tmpId, genre array, Song	fail scenario
CreateSong	adminId, invalid tmpId, genre array, Song	fail scenario
FilterSongs	valid filter with matches	success scenario
FilterSongs	valid fliter without matches	success scenario
FilterSongs	empty string	success scenario
FilterSongs	null	fail scenario