

DBGenre		
Method	Input	Case
GetGenres	valid id	success scenario
GetGenres	invalid id (negative id)	fail scenario
GetGenres	non existing id	fail scenario
AddGenre	valid file id, string of genre already existing in DB	success scenario
AddGenre	valid file id, string of genre not existing in DB	success scenario
AddGenre	invalid file id, string	fail scenario
AddGenre	nonexisting file id, string	fail scenario
AddAllGenres	valid file id, empty genre array	success scenario
AddAllGenres	valid file id, array with one genre string existing in database	success scenario
AddAllGenres	valid file id, array with one genre string not existing in database	success scenario
AddAllGenres	valid file id, array with multiple genre strings not existing in database	success scenario
AddAllGenres	valid file id, array with multiple genre strings existing in database	success scenario
AddAllGenres	valid file id, array with multiple genre strings , some existing in database s	success scenario
AddAllGenres	invalid file id, array	fail scenario
AddAllGenres	non existing id, array	fail scenario
ClearFileGenres	valid id	success scenario
ClearFileGenres	invalid id (negative id)	fail scenario
ClearFileGenres	non existing id	fail scenario
DBUser		
Method	Input	Case
UpdateUser	User with same values as already saved	success scenario
UpdateUser	User with changed name	success scenario
UpdateUser	User with changed username	fail scenario
UpdateUser	User with invalid id	fail scenario
UpdateUser	User with nonexistent id	fail scenario
DeleteUser	valid id	success scenario
DeleteUser	invalid id (negative id)	fail scenario
DeleteUser	non existing id	fail scenario
RateFile	valid userId , and fileId, rating, nonexistent user/file combination	success scenario
RateFile	valid userId , and fileId, rating, existing user/file combination	success scenario
RateFile	invalid userId, valid fileId, rating	fail scenario
RateFile	valid userId, invalid fileId, rating	fail scenario
GetRating	valid userId, valid fileId, user has rated this file	success scenario
GetRating	valid userId, valid fileId, user has not rated this file	success scenario
GetRating	invalid userId, valid fileId	fail scenario
GetRating	valid userId , invalid fileId	fail scenario
PromotetoAdmin	valid promoterId, valid promoteId, promoter is admin	success scenario
PromotetoAdmin	valid promoterId, valid promoteId, promoter is not admin	fail scenario
PromotetoAdmin	invalid promoterId, valid promoteId	fail scenario
PromotetoAdmin	valid promoterId, invalid promoteId, promoter is admin	fail scenario
DemoteAdmin	valid demoterId, valid demoteId, demoter is admin	success scenario
DemoteAdmin	valid demoterId, valid demoteId, demoter is not admin	fail scenario
DemoteAdmin	invalid demoterId, valid demoteId	fail scenario
DemoteAdmin	valid demoterId, invalid demoteId, demoter is admin	fail scenario
Deposit	positive amount, valid id	success scenario
Deposit	negative amount, valid id	success scenario
Deposit	zero, valid id	success scenario
Deposit	int, invalid id	fail scenario
Login	valid username password combination	success scenario
Login	invalid username password combination	fail scenario
Login	non existing username, password	fail scenario
AddUser	user with valid data	success scenario
AddUser	user with invalid data ( i.e. null in username )	fail scenario
GetUser	valid userId	success scenario
GetUser	invalid userId	fail scenario
GetIsAdmin	valid userId	success scenario
GetIsAdmin	invalid userId	fail scenario
GetPurchases	valid userId, user has purchased and rented something	success scenario
GetPurchases	valid userId, user has not purchased or rented something	success scenario
GetPurchases	invalid userId	fail scenario
GetSongs	valid userId, user has purchased and rented songs	success scenario
GetSongs	valid userId, user has not purchased or rented a song	success scenario
GetSongs	invalid userId	fail scenario
GetMovies	valid userId, user has purchased and rented movies	success scenario
GetMovies	valid userId, user has not purchased or rented a movie	success scenario
GetMovies	invalid userId	fail scenario
DBMovie		
Method	Input	Case
UpdateMovie	valid adminId(is admin), Movie with valid data	success scenario
UpdateMovie	invalid adminId, Movie with valid data	fail scenario
UpdateMovie	"valid" adminId(is not admin), Movie with valid data	fail scenario
UpdateMovie	valid adminId(is admin), Movie with invalid data (i.e. Null in title)	fail scenario
GetMovie	valid movieId	success scenario
GetMovie	invalid movieId	fail scenario
GetMovie	non existing movieId	fail scenario
PurchaseMovie	valid userId, valid movieId, sufficient funds	success scenario
PurchaseMovie	valid userId, valid movieId, insufficient funds	fail scenario
PurchaseMovie	invalid userId, movieId, int	fail scenario
PurchaseMovie	userId, invalid movieId, int	fail scenario
RentMovie	valid userId, valid movieId, sufficient funds	success scenario
RentMovie	valid userId, valid movieId, insufficient funds	fail scenario
RentMovie	invalid userId, movieId, int	fail scenario
RentMovie	userId, invalid movieId, int	fail scenario
DeleteMovie	valid adminId(is admin), valid movieId	success scenario
DeleteMovie	"valid" adminId(is not admin), movieId	fail scenario
DeleteMovie	invalid adminId, movieId	fail scenario
DeleteMovie	adminId, invalid movieId	fail scenario
CreateMovie	valid adminId(is admin), valid tmpId, genre array, Movie with valid data	success scenario
CreateMovie	"valid" adminId(is not admin), tmpId, genre array, Movie	fail scenario
CreateMovie	adminId, valid tmpId, genre array, Movie with invalid data (i.e. Null in title)	fail scenario
CreateMovie	Invalid adminId, valid tmpId, genre array, Movie	fail scenario
CreateMovie	adminId, invalid tmpId, genre array, Movie	fail scenario
FilterMovies	valid filter with matches	success scenario
FilterMovies	valid filter without matches	success scenario
FilterMovies	empty string	success scenario
FilterMovies	null	fail scenario
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 filter without matches	success scenario
FilterSongs	empty string	success scenario
FilterSongs	null	fail scenario
DBSFileHandler		
Method	Input	Case
GetNumberOfRatings	valid fileId	success scenario
GetNumberOfRatings	invalid fileId	fail scenario
GetRating	valid fileId	success scenario
GetRating	invalid fileId	fail scenario
DownloadFile	valid userId and fileId combination	success scenario
DownloadFile	invalid userId and fileId combination	fail scenario
DownloadFile	userId, invalid fileId	fail scenario
DownloadFile	invalid userId, fileId	fail scenario
UploadFile	valid StreamFile	success scenario
UploadFile	invalid StreamFile	fail scenario



