# Enhance getMovies Function for Filtering and Sorting Support

## Description:

Enhance the getMovies service function to support filtering and sorting by receiving an object as a parameter. The new parameter structure should be { filters: { page: number, genre: number | null, releaseSort: 'asc' | 'desc' | null } }.

This enhancement enables more precise data fetching from the API by allowing users to specify page numbers, genre filters, and sorting orders.

## Acceptance Criteria:

1. Function Signature:
   1. Enhance the existing getMovies function to accept an object parameter.
2. Parameter Structure:
   1. The parameter should have a property named filters which is an object with the following properties:
      1. page: A property of type number representing the page number.
      2. genre: A property of type number or null representing the genre filter. If null, no genre filtering is applied.
      3. releaseSort: A property of type string ('asc', 'desc', or null) representing the sorting order. If null, no sorting is applied.
3. API Request Modification:
   1. Adjust the API request within the getMovies function to include filtering and sorting based on the provided parameters.
4. Default Values:
   1. Implement default values for genre and releaseSort in case they are not provided. For example, set default values of null for both.
5. Unit Testing:
   1. Write unit tests for the modified getMovies function to ensure it correctly handles filtering and sorting scenarios.
   2. Include test cases for providing different page numbers, genre filters, and sorting orders.

## Definition of Done:

1. The getMovies function is enhanced to accept an object parameter with the structure { filters: { page: number, genre: number | null, releaseSort: 'asc' | 'desc' | null } }.
2. The API request within the function is adjusted to include filtering and sorting based on the provided parameters.
3. Default values for genre and releaseSort are implemented.
4. Unit tests for the modified getMovies function pass successfully, covering various filtering and sorting scenarios.