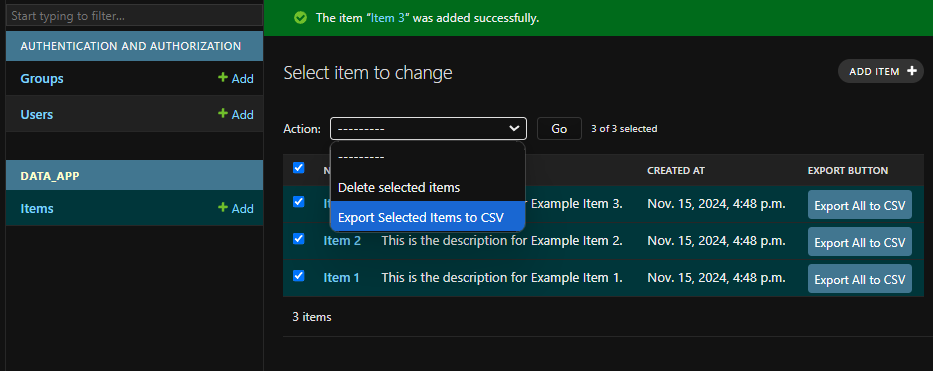
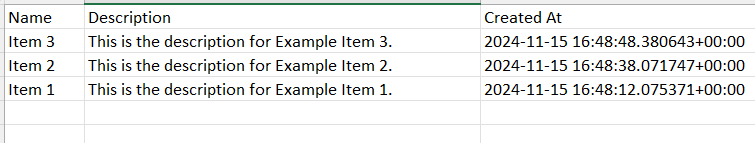
**Django Task-5**

1. **Task Description**

Create a function in views.py that can export the saved data in .csv format and that trigger button should be on superadmin of django long with Actions.

1. **Task Output Screenshot**





1. **Widget/Algorithm Used In Task**

**3.1 Widgets/Tools Used**

1. Django Admin Actions:  
   This feature allows adding custom options to the Actions dropdown menu in Django Admin. It was used to add an option for exporting selected items to a .csv file.
2. Custom Button in Django Admin Interface:  
   A custom button, labeled "Export All to CSV," was added to the admin list view. This button triggers the export of all records in the database to a .csv file. It uses Django's ModelAdmin customization to render the button in the list view.
3. Django HTTP Response:  
   The HttpResponse class is used to create and serve the .csv file as a downloadable attachment.

**3.2 Algorithm Used**

1. Exporting Selected Records:
   * When a user selects specific records and chooses the "Export Selected Items to CSV" action from the dropdown, the selected records are fetched from the database.
   * The data is written to a .csv file using Python's built-in csv module.
   * The file is then sent as an HTTP response to the user for download.
2. Exporting All Records:
   * Clicking the "Export All to CSV" button triggers a database query to retrieve all records in the Item model.
   * The retrieved data is written to a .csv file.
   * The file is returned as an HTTP response, allowing the user to download the entire dataset.
3. Integration with Django Admin:
   * The export functionality is integrated into the Django Admin interface using the ModelAdmin class.
   * The Actions dropdown and the custom export button enhance the user interface, providing both targeted and bulk export options.