|  |  |  |
| --- | --- | --- |
| RAZDA Co. | | |
| **Filename: [discount\_manager.py]** | | |
| **Summary***:*  This file manages product discounts in a centralized way, enabling application, removal, expiration checks, and discount calculations. It uses a database to store discount information and employs caching for performance. It follows the Singleton pattern to ensure a single instance and avoids redundant processing by caching active discounts and periodically updating the cache. | | |
| ***Processes*** | | |
| * **Initialize Singleton Instance** | **Implements the Singleton pattern in \_\_new\_\_ to create only one instance of DiscountManager, ensuring efficient resource use and centralized discount management.** | |
| * **Database Connection & Table Creation** | **\_connect\_db establishes a connection to the discounts.db database, while \_create\_tables ensures necessary tables exist to store discounts.** | |
| * **Load Active Discounts** | **load\_active\_discounts loads currently valid discounts into \_cache, updating it every 30 minutes. This improves performance by reducing database calls.** | |
| * **Apply Discount** | **apply\_discount allows applying a percentage or fixed amount discount to a product, storing details in the cache and database.** | |
| * **Insert/Update Discount** | **\_insert\_or\_update\_discount handles inserting or updating discount records in the database, with parameters for discount type, duration, and conditions.** | |
| * **Remove Discount** | **remove\_discount deletes a specific discount from both the database and cache, ensuring accurate records and removing invalid discounts.** | |
| * **Check & Expire Discounts** | **check\_and\_expire\_discounts identifies expired discounts and removes them from the system, maintaining valid discounts only.** | |
| * **Calculate Discounted Price** | **calculate\_discounted\_price calculates the final discounted price for a product by applying any active discount to the original price.** | |
| * **Get Discount** | **get\_discount retrieves a discount from the cache, reloading the cache if expired, ensuring up-to-date discount information.** | |
| * **List Active Discounts** | **list\_active\_discounts returns a list of all active discounts for easy admin viewing.** | |
| * **Apply Scheduled Discounts** | **apply\_scheduled\_discounts is intended for running pre-configured discounts automatically, triggered by external scheduling tools like cron.** | |
| * **Notify Admin:** | **notify\_admin is a placeholder for notifying administrators of critical issues or updates related to discount operations.** | |
| **Files it Gets Information From:** | | **Files it Sends too:** |
| * **Environment Variables**: Loads DB\_PATH for database path configuration. | * **Discounts Database**: Stores and retrieves discount details in discounts.db, managing active, expired, and updated discounts. | |
| * **discounts.db Database**: Contains product discount data, including discount amount, percentage, and expiration information. | * **Logs Directory**: Logs all activities, errors, and status updates to discount\_manager.log for tracking. | |
| **Expected input into file:** | | **Expected output from file:** |
| * **Discount Requests**: apply\_discount receives inputs for product ID, discount percent, fixed discount, condition, and duration. * **Database Records**: Existing discount records for accurate discount retrievals. * **Admin Alerts**: Notifications for critical events, such as database issues or high-volume discount applications. | | * **Discounted Prices**: calculate\_discounted\_price returns the new price with applied discount, ready for display or order processing. * **Expired Discounts**: check\_and\_expire\_discounts outputs removal of expired discounts from the system, updating cache and database. * **Admin Interface**: list\_active\_discounts supplies discount information for display in the admin panel. |
| **Things that need to be taking place:** | | |
| |  | | --- | | **- Database Initialization: Ensure discounts.db exists and is accessible; \_create\_tables will create necessary tables if missing.** |  |  | | --- | | **- Cache Consistency: Maintain cache expiry times to ensure accurate and up-to-date discount data, as cache refreshes every 30 minutes.** |  |  | | --- | | **- Scheduled Discount Logic: Implement apply\_scheduled\_discounts based on business logic, with appropriate external scheduling triggers.** |  |  | | --- | | **- Error Handling: Log all exceptions during database operations, discount applications, and admin notifications for streamlined troubleshooting.** |  |  | | --- | | **- Admin Notifications: Implement notify\_admin for real-time alerts when critical errors or high-priority events occur, enhancing response times to discount-related issues.** |  |  | | --- | |  | | | |
| Edit log (update each time you make changes to doc or file). | | |
| * Oliver Smith (Razda Admin) Nov 8, 2024: | | |