This n8n workflow automates the process of fetching X post details from RSS feeds, cleaning the data, filtering posts from the last hour, categorizing them by specific X accounts, and appending the results to a Google Sheet. Below is a concise overview of its functionality:

1. Trigger: The "Schedule Trigger" node (scheduleTrigger) initiates the workflow hourly, ensuring regular updates.
2. Fetch X Account List: The "X account list" node (googleSheets) retrieves X accounts and their RSS feed URLs from a Google Sheet (sheet ID: 2080153648, document ID: 1Nn\_jXTmAC79U9Wdsve\_GZ1vQ9r6Sm48paRZbdH79HAQ).
3. Outer Loop: The "OuterLoop" node (splitInBatches) iterates over each X account to process their RSS feeds individually.
4. Read Existing Posts: The "post read" node (googleSheets) fetches existing posts from the "post" sheet (sheet ID: 1995156084) in the same Google Sheet, likely for reference or deduplication.
5. Read RSS Feed: The "RSS Read" node (rssFeedRead) pulls posts from the RSS feed URL of the current X account, sourced from the "X account list" node.
6. Clean Post Content: The "Code" node processes RSS items using JavaScript:
   * Cleans post content by removing HTML tags, scripts, and decoding entities via the extractTweetText function.
   * Validates and converts publication dates to ISO format.
   * Extracts X account name, post content, and date, using title or snippet as fallbacks if content is missing.
   * Outputs JSON with fields: Date and Time, x account, and post.
7. Filter Last Hour Posts: The "last 1 hour" node (code) filters posts to include only those from the last hour, based on the Date and Time field.
8. Switch by Account: The "Switch" node (switch) routes posts to different branches based on X account:
   * Output 0: @coingecko
   * Output 1: @CoinMarketCap
   * Output 2: @WatcherGuru
9. Inner Loops and Google Sheets Append:
   * Each branch uses an "innerloop" node (innerloop2, innerloop, innerloop1 for @coingecko, @CoinMarketCap, @WatcherGuru, respectively, using splitInBatches) to process posts individually.
   * Corresponding "Google Sheets" nodes (Google Sheets2, Google Sheets1, Google Sheets) append posts to the "post" sheet, mapping Date and Time, x account, and post, with Date and Time used to avoid duplicates.
   * "Wait" nodes (Wait2, Wait, Wait1) add a 1-second delay after each append to manage Google Sheets API rate limits.
10. Loop Back: After processing, inner loops return to the "OuterLoop" to handle the next X account’s RSS feed until all accounts are processed.

Key Features:

* Automation: Runs hourly via the schedule trigger.
* Data Source: Fetches posts from RSS feeds listed in a Google Sheet.
* Data Cleaning: Strips HTML and decodes entities for clean text.
* Time Filter: Processes only the last hour’s posts.
* Account Routing: Separates posts by @coingecko, @CoinMarketCap, and @WatcherGuru.
* Output: Stores data in a Google Sheet for analysis.
* Rate Limiting: Uses wait nodes to ensure stable API interactions.

Notes:

* Requires valid RSS feed URLs in the "x account" sheet and Google Sheets OAuth2 credentials (ID: HnNAuXH7E59P2ff6).
* The workflow is inactive (active: false) and may need activation for scheduled runs.
* Assumes consistent RSS feed fields (pubDate, creator, content) for reliable processing.
* The Google Sheet (1Nn\_jXTmAC79U9Wdsve\_GZ1vQ9r6Sm48paRZbdH79HAQ) has two sheets: one for accounts and one for posts.

This workflow efficiently collects, processes, and stores X posts from RSS feeds, tailored for specific accounts and recent activity.