AI Tweet Agent — Workflows Documentation

Last updated: 2025-08-26

# Overview

Two n8n workflows: a Poster that publishes approved tweets from Airtable every morning, and a Generator that drafts tweets using an AI model and stores them for review.

# Architecture

Workflow A — Tweet Poster: Schedule → Airtable Search (Approved & Pending) → Create Tweet (X) → Update Airtable

Workflow B — Tweet Generator: Schedule → Message a Model (Perplexity) → Code (JSON clean/parse) → Create Airtable Record

# Prerequisites

• n8n (cloud or self-hosted)  
• Airtable base with a 'Tweets' table  
• X (Twitter) OAuth2 credentials  
• Perplexity (or other LLM) API key

# Airtable Schema

Fields:  
- Name (string)  
- Tweet (long text)  
- Status (Pending Review | Approved | Rejected)  
- Posted (Pending | Posted)  
- Created At (date)

# Workflow A — Tweet Poster

Filter formula for Airtable Search:

AND({Status} = 'Approved', {Posted} = 'Pending')

Posting step uses the X node with text mapped to the Tweet field, then updates the Airtable record to Posted.

# Workflow B — Tweet Generator

Prompt the model to return EXACTLY a JSON array of 5 strings (tweets under 280 chars).

Use the following Code Node to clean/parse and map to Airtable:

// Handle Perplexity output  
let output = $json;  
  
if (Array.isArray(output)) {  
 output = output[0];  
}  
  
let content = output?.choices?.[0]?.message?.content || "";  
  
// Remove ```json ... ``` wrappers  
content = content.replace(/```json|```/g, "").trim();  
  
let tweets;  
try {  
 tweets = JSON.parse(content);  
} catch (e) {  
 throw new Error("Failed to parse tweets JSON: " + e.message + "\nContent: " + content);  
}  
  
return tweets.map(t => ({  
 json: {  
 Name: t.substring(0, 50),  
 Tweet: t,  
 Status: "Pending Review",  
 Posted: "Pending",  
 Created\_At: new Date().toISOString()  
 }  
}));

## Field Mapping Tips

Name → ={{ $json.Tweet.slice(0, 50) }}  
Tweet → ={{ $json.Tweet }}  
Status → ={{ $json.Status }}  
Created At → ={{ $json.Created\_At.substring(0,10) }}

# External Services

Airtable (store & review), X API (posting), Perplexity (LLM for draft generation).

# Testing

1) Run Generator → confirm records in Airtable  
2) Approve a subset  
3) Run Poster → confirm X post and Posted status update

# Production Notes

Add retries and error handling; respect API limits; log run IDs; trim to 280 chars; align cron to IST.