QN-8 Interview Scheduling Automation

**Problem Statement**:

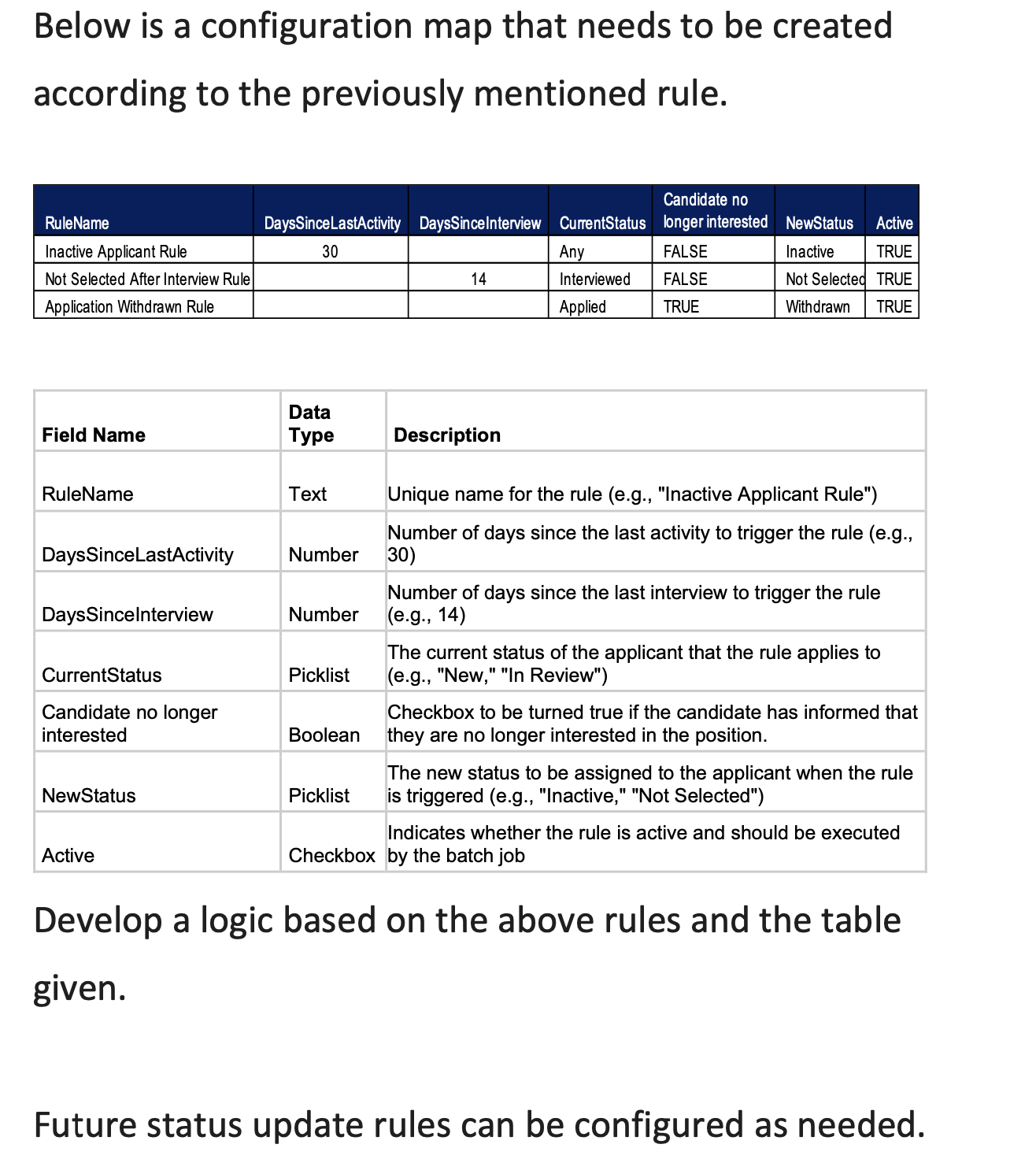
We have a large volume of job applicants in our system, and their statuses need to be updated regularly based on various criteria. For example, applicants who haven't been updated in 30 days should be moved to an "Inactive" status, and those who have been interviewed but not advanced should be marked as "Not Selected." Currently, this process is manual and time-consuming, leading to inconsistencies and outdated applicant information. We need an automated batch job that can efficiently update applicant statuses based on predefined rules. This will help us maintain accurate records, improve reporting, and ensure a streamlined recruitment process.

**Solution Design**:

Functional Requirements:

The batch job should update the "Status" field on the Applicant object based on the following rules:

* Applicants with no activity (e.g., no status changes, interviews, or communications) for 30 days should be marked as "Inactive."
* Applicants who have been interviewed but haven't progressed to the next stage within 14 days of the interview should be marked as "Not Selected."
* Applicants who have communicated that they are no longer interested, and has been recorded as a checkbox on candidate record, their status needs to be updated to “Withdrawn”
* Below is a Configuration map that’s needs to be created according to previously mentioned rule.



Technical Requirements:

* Batch Apex: The solution should be implemented using Batch Apex to handle large volumes of applicant records efficiently.
* The batch job should be scheduled to run on a regular Scheduling: The batch job should be scheduled to run on a regular basis (e.g., nightly or weekly) to keep applicant statuses up-to-date.
* Query Optimization: The batch job should use efficient SOQL queries to retrieve and process applicant records.
* Error Handling: The batch job should include robust error handling mechanisms to manage exceptions and prevent data corruption.

Additional Considerations:

* Configuration: The batch job should be configurable to allow administrators to modify the status update rules and schedule as needed. (Use custom Metadata for the configuration).
* Performance: The batch job should be optimized for performance to minimize execution time and resource consumption.
* Testing: Thorough testing should be conducted to ensure the batch job functions correctly and meets all requirements.

**Solution Process**:  
  
Step1:

Have created one custom meta-date named “Rule” and added all the associated fields which are mentioned in requirements. In Below, It’s the Rule meta-data Snap and it’s manage records.(Snap01 & Snap02)

 Snap01  
  
A screenshot of a computer

Description automatically generated Snap02

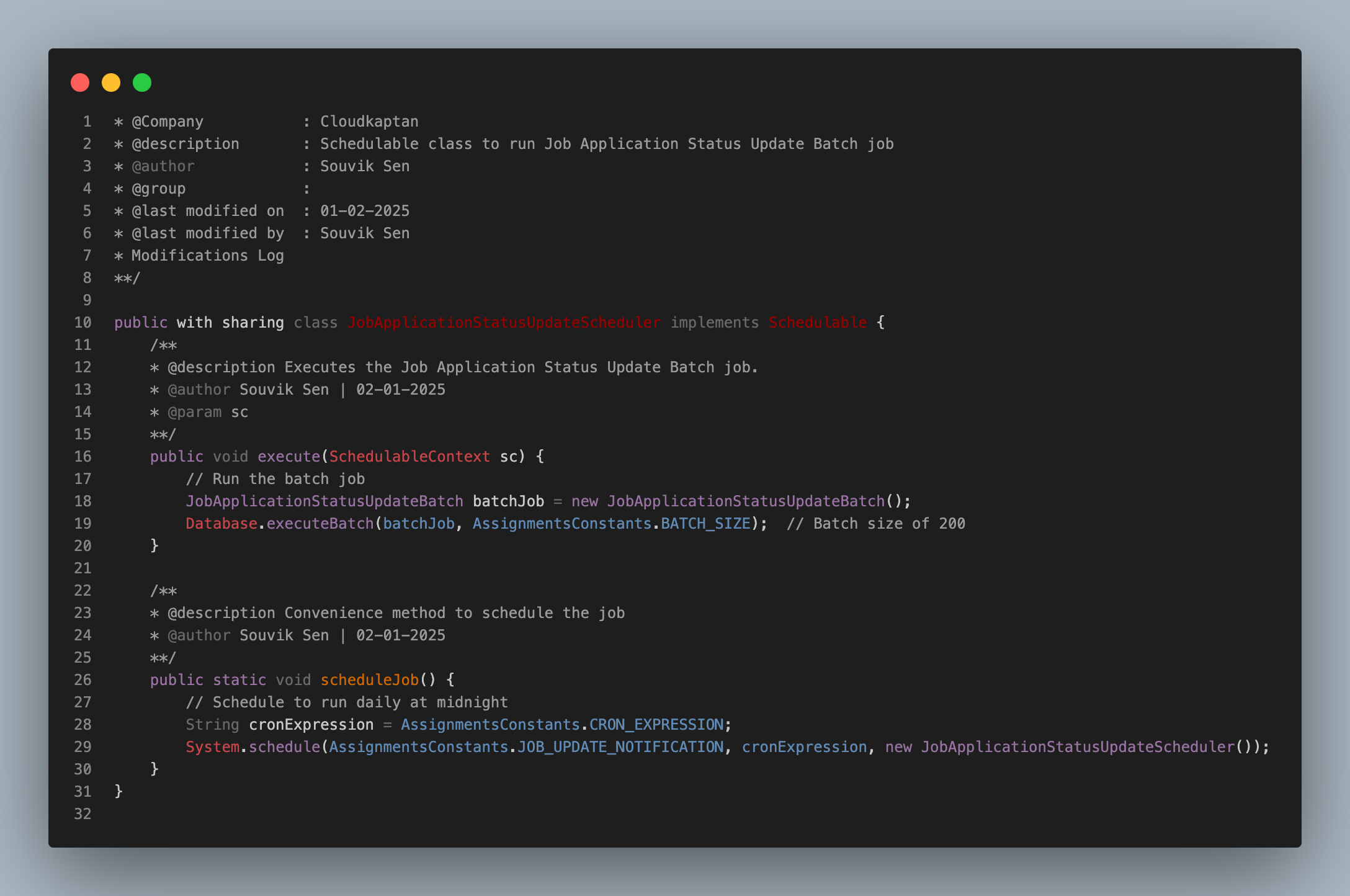
Step 2:

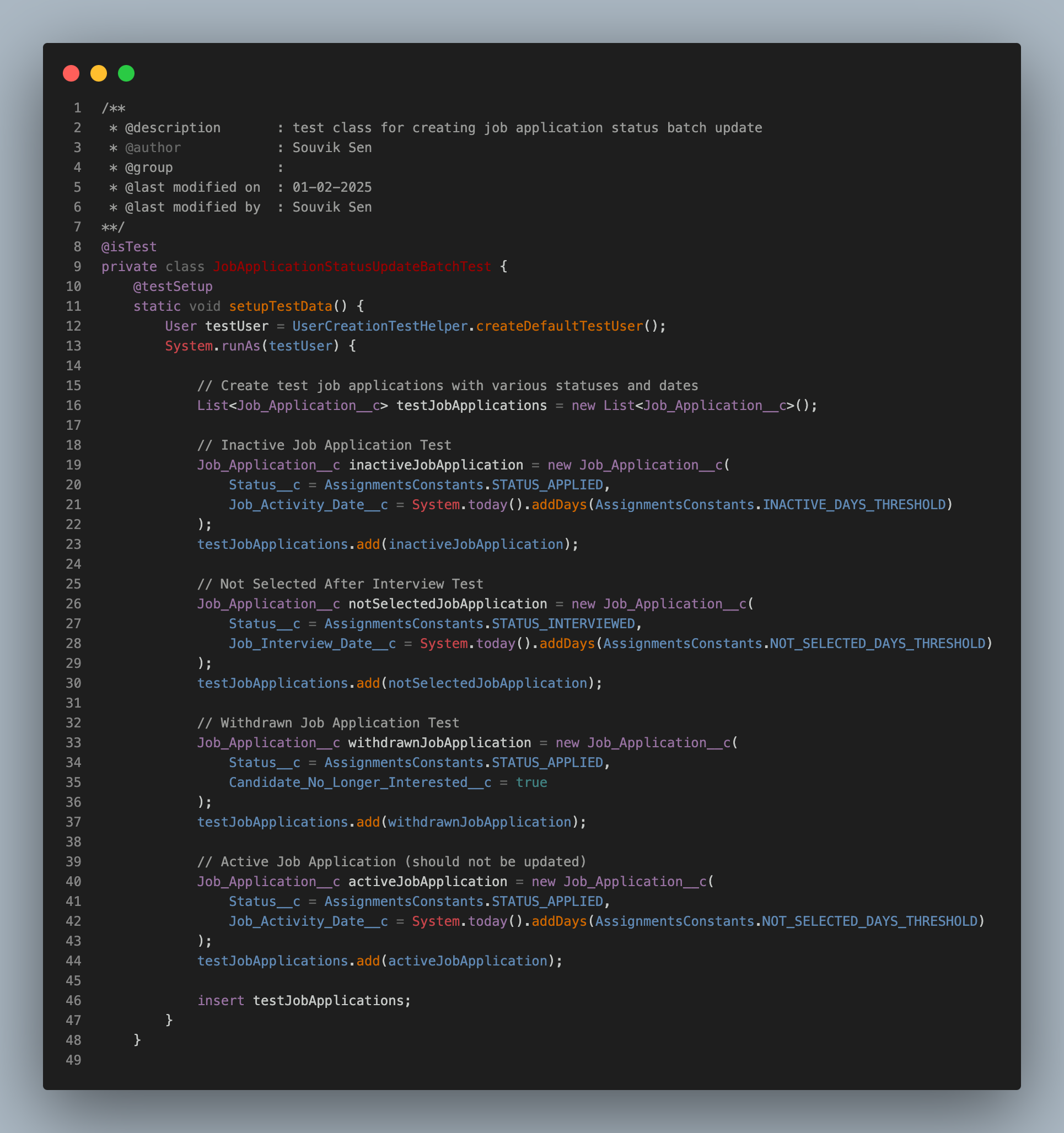
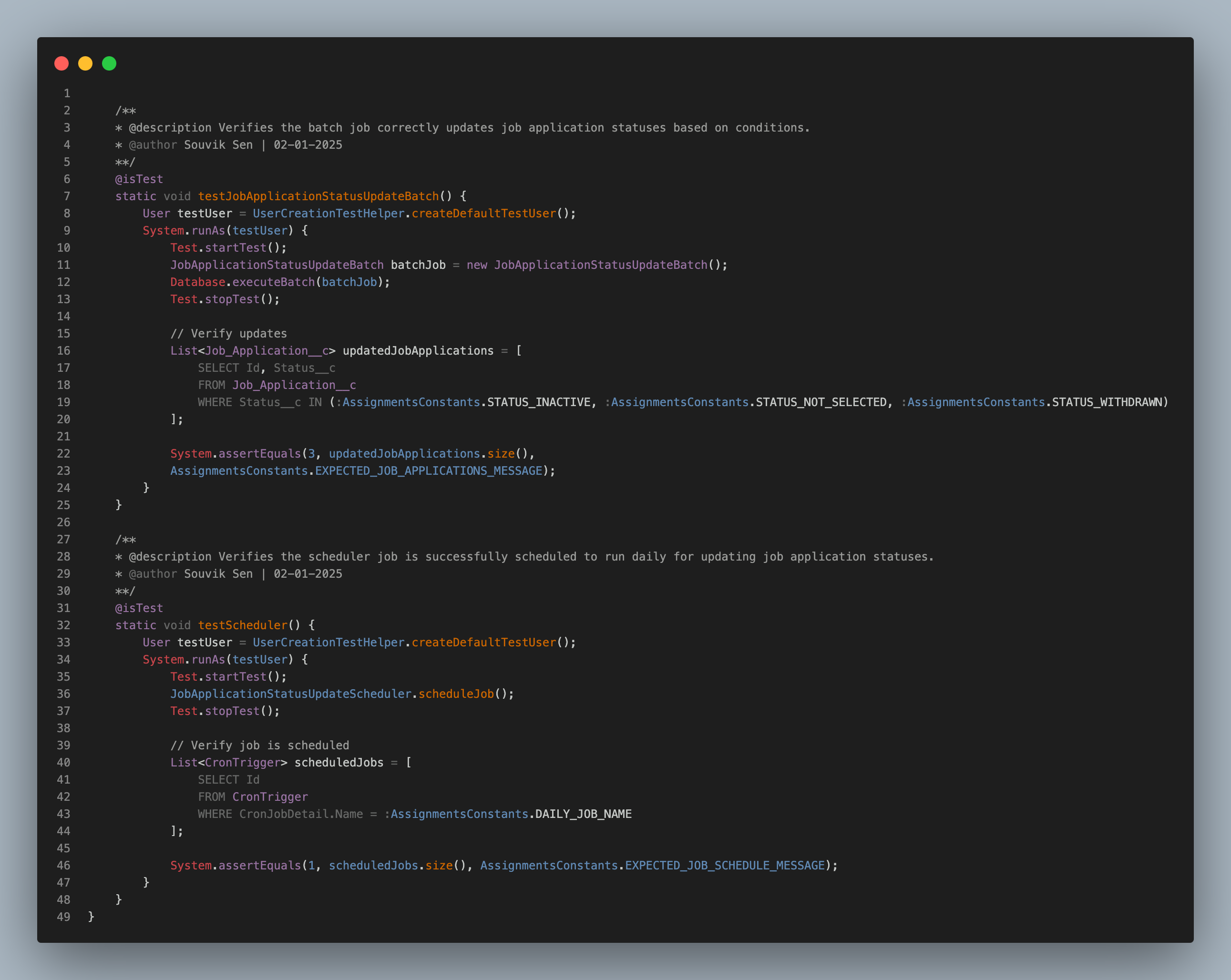
This Class defines a data model for job application status update rules.

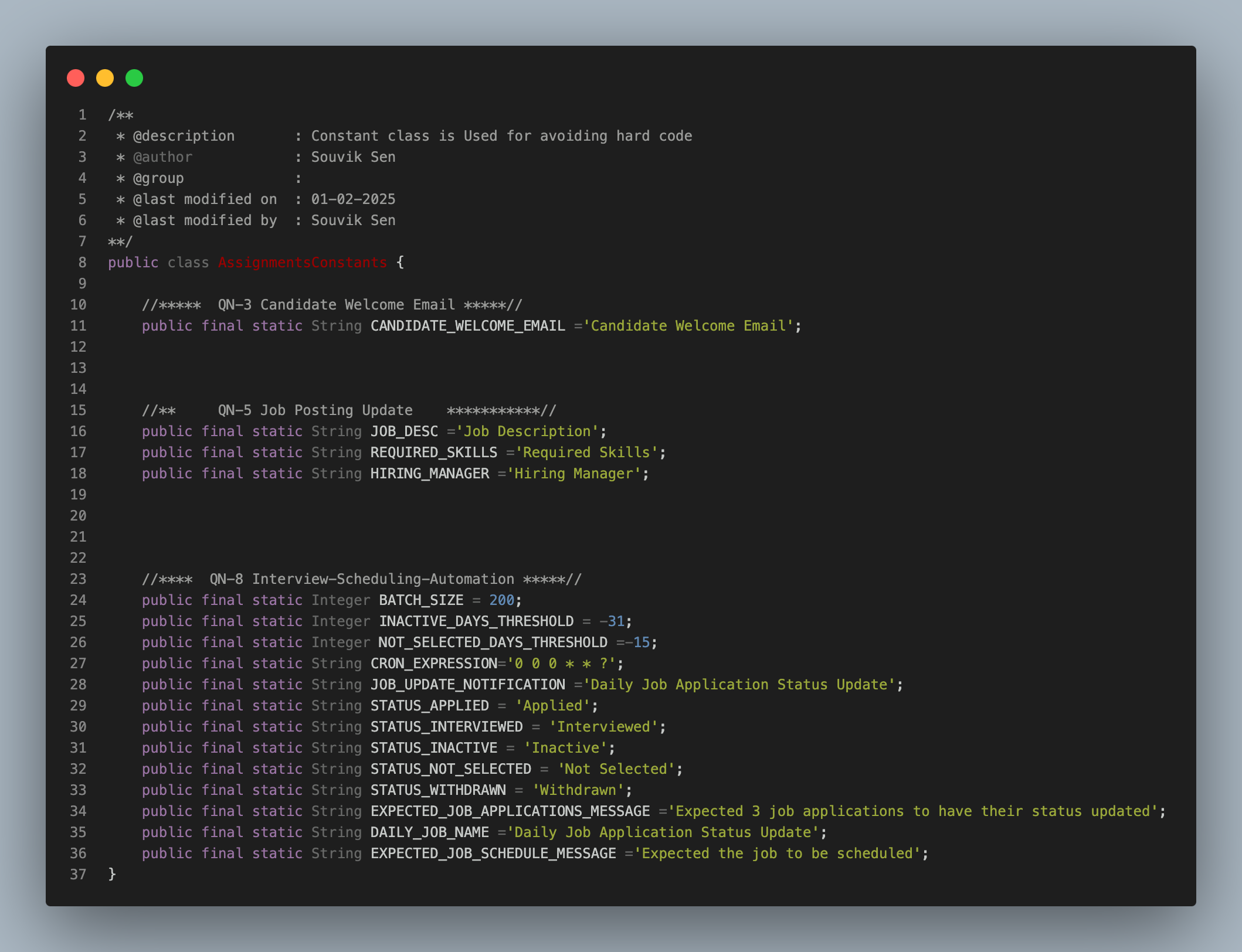


Step 3:  
Batch Apex class for update Job Application Statuses based on Config rule  




Step 4: Schedulable class to run Job application status update batch job  


Step 5: Test Class   
  
  
  
  
  
  
  
  
  
  
  


Step 6: Constant Class  
  


Output Result:  
  
  
  
A screenshot of a computer

Description automatically generated