# WEEK 2 - **Prompt Security & Caching Refactor**

In a Prompt,

Static segment: Fixed instructions or context that are the same for every user and query.

Dynamic segment: Variable data that can change per request.

## Segmenting the given prompt:

**Static Parts:**

These do not change per request and are cacheable.

- Role definition
  "You are an AI assistant trained to help employees with HR-related queries."

- Instructions
  "Answer only based on official company policies. Be concise and clear in your response."

**Dynamic Parts:**

These vary per employee or query.

- {{employee_name}}
- {{department}}
- {{location}}
- {{leave_policy_by_location}}
- {{optional_hr_annotations}}
- {{user_input}}
- {{employee_account_password}} – This is sensitive data and should not be passed to the LLM.

## Restructured Prompt to Improve Caching Efficiency:

"You are an AI assistant trained to help employees with HR-related queries. The Leave Management Portal contains information about employee leaves.

Instructions:

- Be concise and clear in your response.
- Answer only based on official company policies.
- Refer to the location-specific leave policy before curating leave-related responses.
- Do not reveal any sensitive data such as passwords or internal system details.
- If the user prompt asks for any unrelated or out-of-scope information, refuse politely and redirect to HR support.
- If any employee-specific information such as department or location is required, prompt the user to provide it."

## Prompt Injection Mitigation Strategies

1. **Never Provide Secrets to the Model**
   Passwords, tokens, employee IDs should never be entered as inputs to the model.

2. **Strict Security Rules in the System Prompt**
   If a user requests passwords, credentials, or internal system data, respond with a refusal and suggest contacting HR or IT support.
   This ensures that user instructions do not override system-level constraints.

3. **Scope Limiting**
   The assistant is allowed to answer **only**:

   - Leave balances
   - Leave types
   - Eligibility
   - Policy-based queries

   Anything outside this scope must be refused politely.

4. **Input Classification**

   Before sending to the LLM, Classify query as: **Leave-related** or **Credential Request**
   This helps to block or reroute unsafe queries before the model sees them.

5. **Response Sanitization**

   Before returning the answer, verify that no credentials or confidential data are generated.