Academic Year	**2025 - 26**	**Experiment No.**	**7**
Course & Semester	**B.E. – Sem. VIII**	**Subject Name**	**Recommendation Systems Lab**
Experiment Type	**Software Performance**	**Subject Code**	**CSDOL8022**

Name of Student:		**Roll No.:**	
Date of Performance:		**Date of Submission:**	
LO Mapping	CSDOL8022.4 : Design, implement and analyze Knowledge-based recommendation systems.		

Indicator	**Poor**	**Average**	**Good**
Timeline Maintains submission deadline (3)	Submission not done (0)	One or More than One week late (1-2)	Maintains deadline (3)
Completion and Organization (3)	N/A	Document is just acceptable (1-2)	Completed whole document and neatly organized (3)
Program Performance (2)	Could not perform at all (0)	Implemented few parts (1)	Full implementation (2)
Knowledge In depth knowledge of the Experiment (2)	Unable to answer questions (0)	Unable to answer few questions (1)	Able to answer all questions (2)

Assessment Marks:

Timeline	
Completion and Organization	
Program Performance	
Knowledge	

Total: (Out of 10)

Teacher's Sign: Student Sign:

Experiment No. 7

- **Aim:**

  Case Study on Knowledge Based Recommender

- **Objective:** The objective of this problem statement is to understand Knowledge Based Recommender.

- **Theory:**

A **Knowledge-Based Recommender System (KBRS)** is a type of recommendation system that relies on **explicit domain knowledge** about how specific item features meet user needs. Instead of learning patterns from user interactions, it applies **rules, logic, and reasoning** to recommend items based on user preferences or stated requirements.

Key Features of KBRS:

1. **Domain Knowledge-Driven**:
   - Recommendations are based on detailed information about items and their attributes, as well as how they fulfill user requirements.
   - Domain experts or structured data often define this knowledge.
2. **User-Centric Input**:
   - Users explicitly state their preferences, goals, or constraints (e.g., "I need a laptop with 16GB RAM under $1000").
   - The system matches these preferences with items in its database.
3. **Rule-Based Reasoning**:
   - Uses predefined rules or algorithms to filter and rank items that meet user criteria.
   - Example: A car recommender system might prioritize fuel efficiency if the user specifies "low running costs."
4. **No Cold-Start Problem**:
   - Unlike collaborative filtering, KBRS doesn't rely on historical user data, so it works well for new users or items.
5. **Transparent Recommendations**:
   - Recommendations are explainable because the reasoning process follows logical rules that can be presented to users.

How It Works:

1. **Knowledge Acquisition**:
   - Collect and organize domain knowledge, often in the form of a knowledge base (e.g., product specifications, compatibility rules, or user needs).
2. **User Input**:
   - Obtain explicit input from users regarding their preferences, requirements, or constraints.
3. **Matching Process**:
   - Compare user input with the knowledge base using rules, constraints, or reasoning models.
4. **Recommendation Generation**:
   - Rank and recommend items that best align with user preferences.

Applications:

- **E-commerce**:
  - Suggesting products based on specific needs (e.g., choosing a smartphone with a particular screen size and battery capacity).
- **Travel**:
  - Recommending vacation packages based on budget, activities, and climate preferences.
- **Healthcare**:
  - Proposing treatments or medications based on patient symptoms and medical history.
- **Education**:
  - Recommending courses or programs tailored to a student's career goals.

- ## **Implementation:**

- ## **Conclusion:**

- ## **Reference:**