**AI-LAB – ASSIGNMENT#1**

**USMAN JAVED**

**ROLL NO: Su92-BSAIM-F24-002**

**SECTION: BSAI-3A**

# Lab 3 Task: Model-Based Reflex Agent

This document explains the implementation and working of a Model-Based Heater Agent. The agent is designed to control a heater based on the room temperature while remembering its previous action. This prevents unnecessary switching of the heater ON or OFF.

## Python Code

class HeaterAgent:

    def \_\_init\_\_(self):

        self.last\_action = "OFF"

    def decide\_action(self, current\_temp):

        if current\_temp < 20 and self.last\_action == "OFF":

            self.last\_action = "ON"

            return "Turn ON"

        elif current\_temp > 22 and self.last\_action == "ON":

            self.last\_action = "OFF"

            return "Turn OFF"

        else:

            return "Do nothing"

agent = HeaterAgent()

print(f"Action needed at 19°C: {agent.decide\_action(19)}")

print(f"Action needed at 20.5°C: {agent.decide\_action(20.5)}")

print(f"Action needed at 23°C: {agent.decide\_action(23)}")

## Step-by-Step Explanation

A **model-based reflex agent** uses memory to make smarter decisions than a simple reflex agent. It remembers its **last action** to avoid unnecessary changes.

1. **Sensing**: The agent reads the current room temperature from a sensor.
2. **Memory (Model)**: It checks its internal memory, which stores the **last action** it took (either 'Heater ON' or 'Heater OFF').
3. **Logic**: It decides on the next action by combining the current temperature with its memory.
4. **Action**: It will only turn the heater ON if it's too cold **AND** its memory says the heater is currently OFF.
5. **Action**: It will only turn the heater OFF if it's too hot **AND** its memory says the heater is currently ON.
6. **Update**: After acting, it updates its memory to reflect the new state, preventing rapid on-off cycling.