**RIPHAH INTERNATIONAL UNIVERSITY, ISLAMABAD**

****

**Lab 8**

**Bachelors of Computer science – 6th semester**

**Subject:** Artificial Intelligence Lab

**Submitted to:** Ma’am Ayesha

**Submitted by:** Munaza Malik

**Sap Id:** 44956

**Date:** 25th March, 2025

**Tasks**

**Question 01:**

Write a program for a simple reflex agent. The agent will act as a vacuum cleaner. In the first activity, we will create an environment for the agent.

* The environment is divided into 4 portions A,B,C and D.
* Then define two states for each portion.
* 0 indicates the cleaned state and 1 indicates the dirty state.
* We will initialize each portion with a random state that would be either 0 or 1.

**Code:**

import random

def initialize\_environment():

# Creating an environment with random states (0: Clean, 1: Dirty)

return {"A": random.randint(0, 1),

"B": random.randint(0, 1),

"C": random.randint(0, 1),

"D": random.randint(0, 1)}

def vacuum\_cleaner(environment):

for location, state in environment.items():

print(f"Vacuum is at {location}. State: {'Dirty' if state == 1 else 'Clean'}")

if state == 1:

print(f"Cleaning {location}...")

environment[location] = 0 # Clean the location

print(f"{location} is now clean.\n")

def main():

environment = initialize\_environment()

print("Initial Environment State:")

print(environment, "\n")

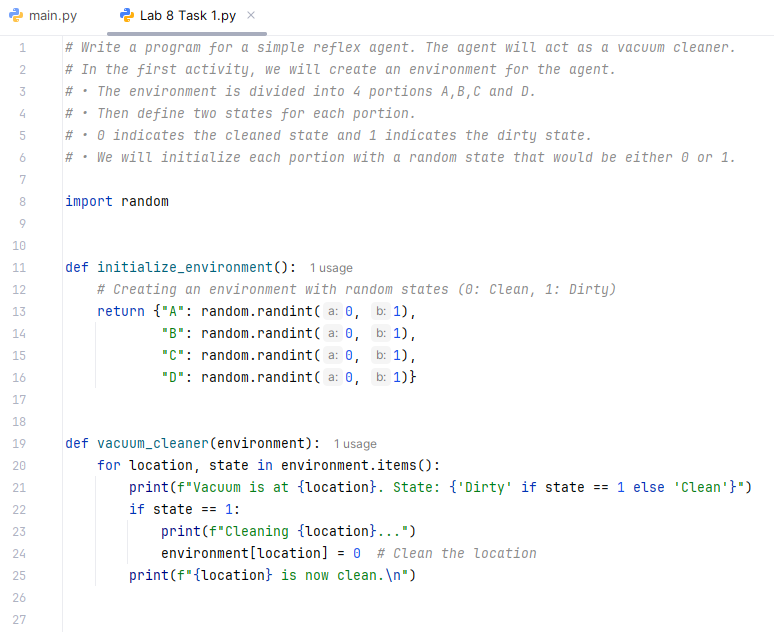
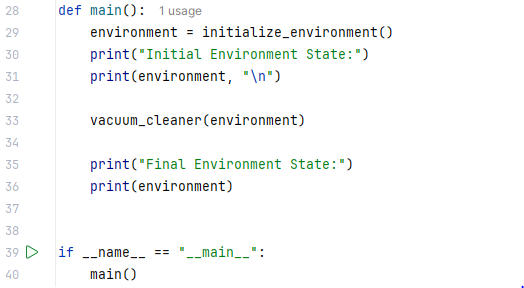
vacuum\_cleaner(environment)

print("Final Environment State:")

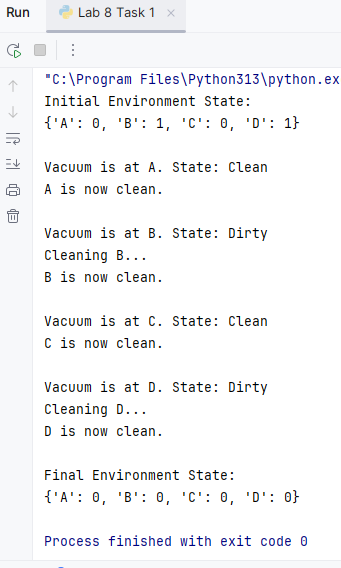
print(environment)

if \_\_name\_\_ == "\_\_main\_\_":

main()

**Output:**



**Question 02:**

Create a Simple Reflex Agent that:

* Observes traffic light color (red, yellow, green).
* Takes an action based on the light:

Red → Stop

Yellow → Slow down

Green → Move

**Code:**

import random

def get\_traffic\_light():

# Randomly selecting a traffic light color

return random.choice(["Red", "Yellow", "Green"])

def reflex\_agent(light\_color):

actions = {

"Red": "Stop",

"Yellow": "Slow down",

"Green": "Move"

}

return actions.get(light\_color, "Unknown action")

def main():

light\_color = get\_traffic\_light()

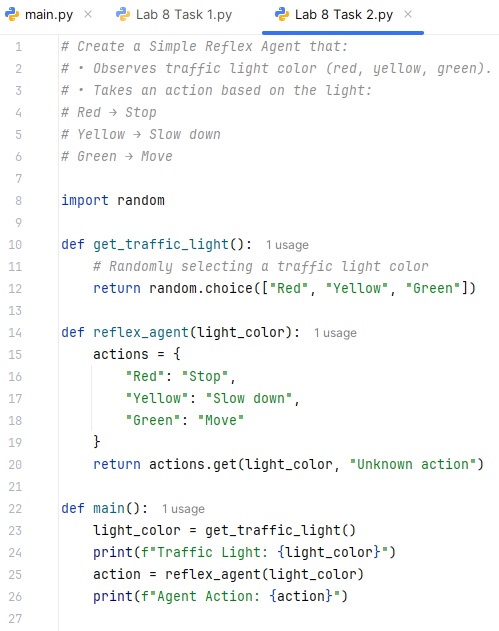
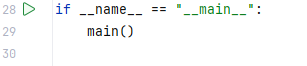
print(f"Traffic Light: {light\_color}")

action = reflex\_agent(light\_color)

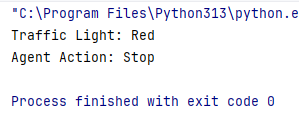
print(f"Agent Action: {action}")

if \_\_name\_\_ == "\_\_main\_\_":

main()

**Output:**



**Question 03:**

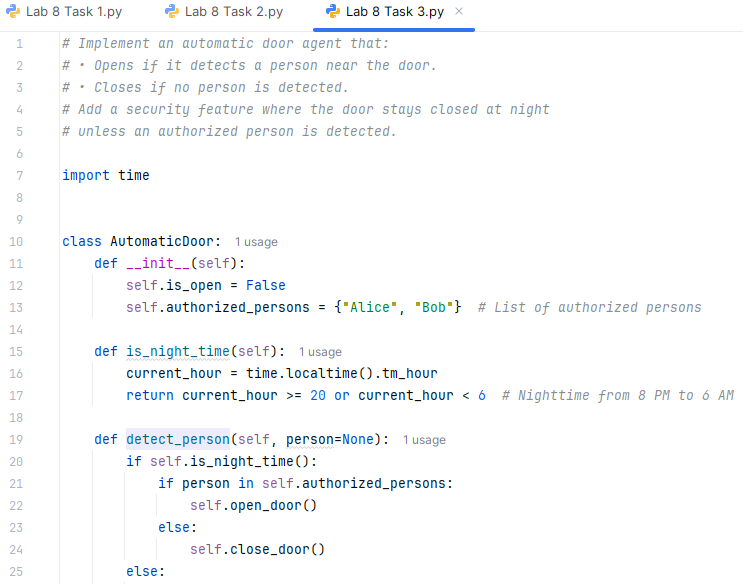
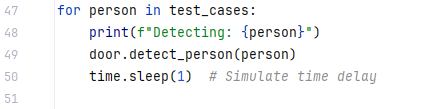
Implement an automatic door agent that:

* Opens if it detects a person near the door.
* Closes if no person is detected.

Add a security feature where the door stays closed at night unless an authorized person is detected.

**Code:**

import time  
  
  
class AutomaticDoor:  
 def \_\_init\_\_(self):  
 self.is\_open = False  
 self.authorized\_persons = {"Alice", "Bob"} *# List of authorized persons* def is\_night\_time(self):  
 current\_hour = time.localtime().tm\_hour  
 return current\_hour >= 20 or current\_hour < 6 *# Nighttime from 8 PM to 6 AM* def detect\_person(self, person=None):  
 if self.is\_night\_time():  
 if person in self.authorized\_persons:  
 self.open\_door()  
 else:  
 self.close\_door()  
 else:  
 if person:  
 self.open\_door()  
 else:  
 self.close\_door()  
  
 def open\_door(self):  
 if not self.is\_open:  
 self.is\_open = True  
 print("Door is now OPEN.")  
  
 def close\_door(self):  
 if self.is\_open:  
 self.is\_open = False  
 print("Door is now CLOSED.")  
  
  
*# Simulation*door = AutomaticDoor()  
  
test\_cases = [None, "Alice", "Bob", "Unknown"]  
  
for person in test\_cases:  
 print(f"Detecting: {person}")  
 door.detect\_person(person)  
 time.sleep(1) *# Simulate time delay*

**Output:**

