**ASSIGNMENT - 2**

# 

# 

# **CS2V61-KNOWLEDGE ENGINEERING**

**TOPIC:** RECIPE RECOMMENDATION SYSTEM

SUBMITTED BY,

SYEDALI FATHIMA S

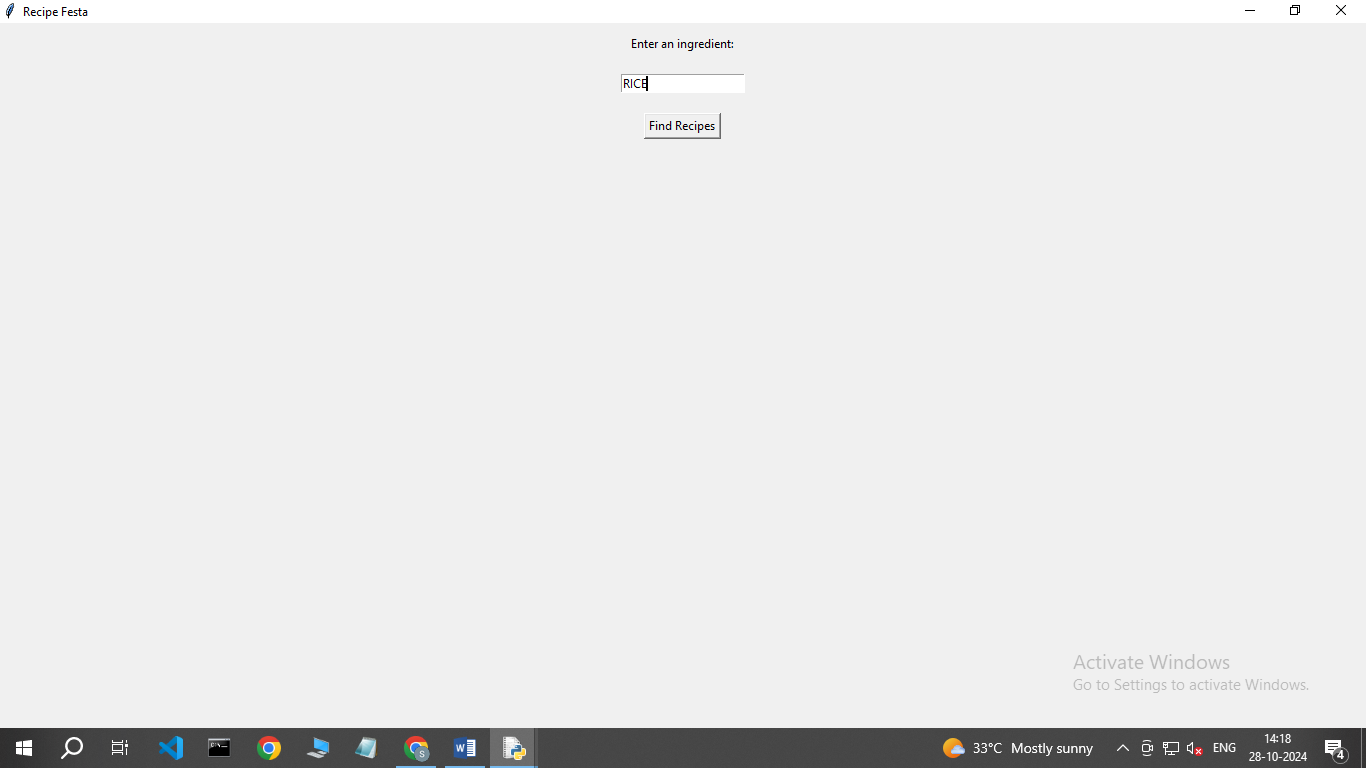
III CSE D

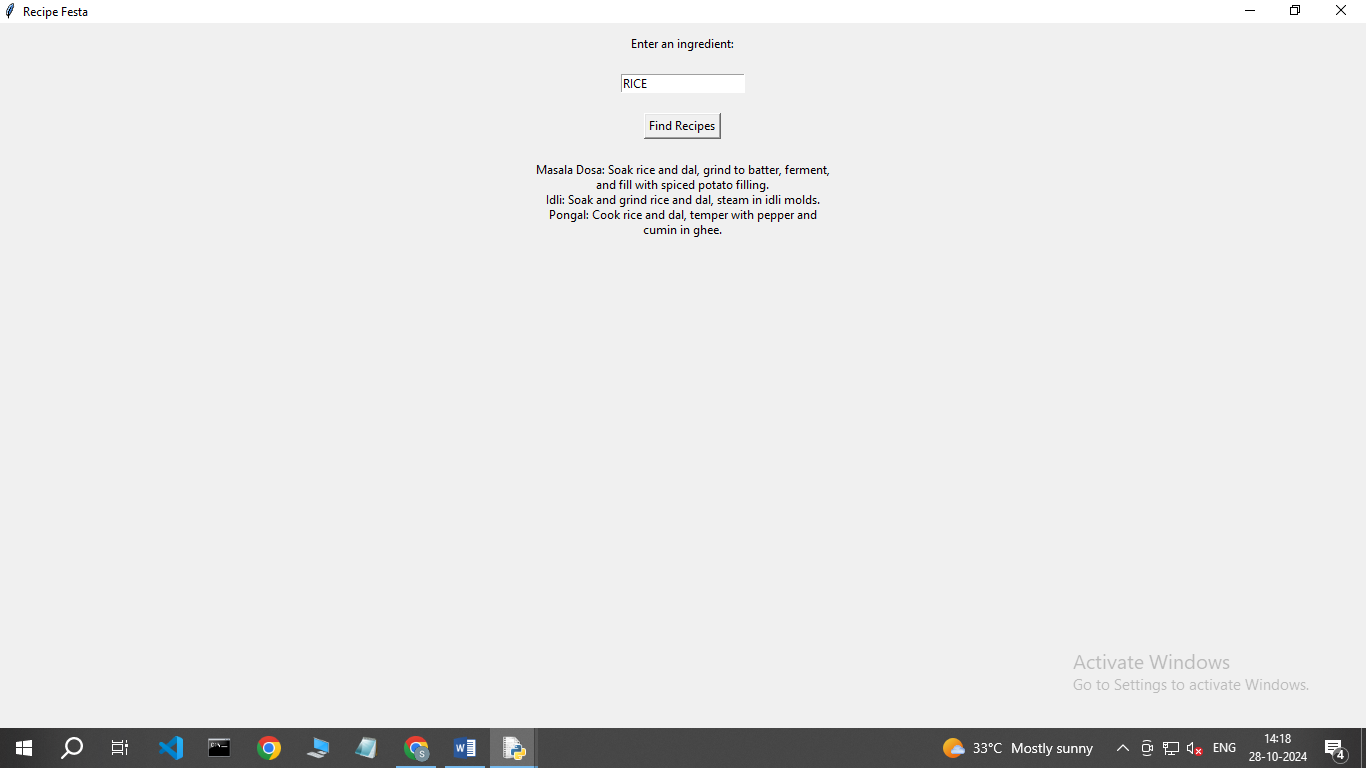
92132213228

PROGRAM:

import tkinter as tk  
from tkinter import messagebox  
  
# Expert System Logic  
class Recipe:  
    def \_\_init\_\_(self, name, ingredients, instructions):  
        [self.name](http://self.name/) = name  
        self.ingredients = ingredients  
        self.instructions = instructions  
  
class RecipeExpertSystem:  
    def \_\_init\_\_(self):  
        self.recipes = []  
  
    def add\_recipe(self, recipe):  
        self.recipes.append(recipe)  
  
    def find\_recipe(self, ingredient):  
        results = []  
        for recipe in self.recipes:  
            if ingredient in recipe.ingredients:  
                results.append(recipe)  
        return results  
  
# Initialize the expert system  
recipe\_system = RecipeExpertSystem()  
  
# Sample South Indian recipes  
recipe\_system.add\_recipe(Recipe("Masala Dosa",  
    ["rice", "urad dal", "fenugreek seeds", "potatoes", "onion", "green chilies"],  
    "Soak rice and dal, grind to batter, ferment, and fill with spiced potato filling."))  
     
recipe\_system.add\_recipe(Recipe("Sambar",  
    ["toor dal", "mixed vegetables", "sambar powder", "turmeric powder", "onion", "tomatoes"],  
    "Cook dal and vegetables together, add spices, and simmer."))  
  
recipe\_system.add\_recipe(Recipe("Coconut Chutney",  
    ["grated coconut", "green chilies", "ginger"],  
    "Blend coconut with chilies and ginger, then temper with mustard seeds."))  
  
recipe\_system.add\_recipe(Recipe("Idli",  
    ["rice", "urad dal"],  
    "Soak and grind rice and dal, steam in idli molds."))  
  
recipe\_system.add\_recipe(Recipe("Pongal",  
    ["rice", "moong dal", "black pepper", "cumin seeds"],  
    "Cook rice and dal, temper with pepper and cumin in ghee."))  
  
# User Interface  
class RecipeFestaApp:  
    def \_\_init\_\_(self, master, recipe\_system):  
        self.master = master  
        self.recipe\_system = recipe\_system  
        self.master.title("Recipe Festa")  
         
        self.label = tk.Label(master, text="Enter an ingredient:")  
        self.label.pack(pady=10)  
         
        self.ingredient\_entry = tk.Entry(master)  
        self.ingredient\_entry.pack(pady=10)  
         
        self.search\_button = tk.Button(master, text="Find Recipes", command=self.find\_recipes)  
        self.search\_button.pack(pady=10)  
  
        self.result\_label = tk.Label(master, text="", wraplength=300)  
        self.result\_label.pack(pady=10)  
  
    def find\_recipes(self):  
        ingredient = self.ingredient\_entry.get().lower()  
        if not ingredient:  
            messagebox.showwarning("Input Error", "Please enter an ingredient.")  
            return  
  
        recipes\_found = self.recipe\_system.find\_recipe(ingredient)  
        if recipes\_found:  
            results = "\n".join([f"{[recipe.name](http://recipe.name/)}: {recipe.instructions}" for recipe in recipes\_found])  
        else:  
            results = "No recipes found for this ingredient."  
         
        self.result\_label.config(text=results)  
  
# Create the main window  
if \_\_name\_\_ == "\_\_main\_\_":  
    root = tk.Tk()  
    app = RecipeFestaApp(root, recipe\_system)  
    root.mainloop()

OUTPUT:





CONCLUSION:

This Recipe Festa program provides an interactive way to discover South Indian recipes based on ingredients. Users can easily input an ingredient and receive relevant recipe suggestions, enhancing their cooking experience. The simple yet effective interface makes it accessible for anyone looking to explore new dishes.