#Author : Authors of Learn & Practice AI by Projects

#Purpose : FastAPI application to serve the user request for IRIS model's prediction request.

#import libraries

from fastapi import FastAPI

import uvicorn

import joblib

import json

#declare FastAPI instance

fastAPIInstance = FastAPI()

joblib\_file = "Iris\_Classifer.jbl"

classification\_model = joblib.load(joblib\_file)

@fastAPIInstance.get('/')

def home():

#returns default string for home url access

return '<h1>Classification Model(FastAPI based) is running successfully!!!</p>'

@fastAPIInstance.get('/predict-class')

def predict\_class(sepal\_length:float, sepal\_width:float, petal\_length:float, petal\_width:float):

#print the arguments' value on the command prompt

print("sepal\_length : {} , sepal\_width: {}, patal\_length : {}, petal\_width : {}".format(sepal\_length, sepal\_width, petal\_length, petal\_width))

#predict the class for the given arguemnt

result = classification\_model.predict([[sepal\_length, sepal\_width, petal\_length, petal\_width]])

# print the predicted result(json format)

print(json.dumps({"predicted-result": result[0]}))

# return the predicted result(json format)

return (json.dumps({"predicted-result": result[0]}))