Slip 15

15.1

// Subsystem 1: DVD Player

class DVDPlayer {

public void on() {

System.out.println("DVD Player is ON");

}

public void play(String movie) {

System.out.println("Playing movie: " + movie);

}

public void off() {

System.out.println("DVD Player is OFF");

}

}

// Subsystem 2: Projector

class Projector {

public void on() {

System.out.println("Projector is ON");

}

public void setInput(DVDPlayer dvdPlayer) {

System.out.println("Setting input to DVD Player");

}

public void off() {

System.out.println("Projector is OFF");

}

}

// Subsystem 3: Lights

class Lights {

public void dim() {

System.out.println("Dimming the lights");

}

public void brighten() {

System.out.println("Brightening the lights");

}

}

// Facade: HomeTheaterFacade

class HomeTheaterFacade {

private DVDPlayer dvdPlayer;

private Projector projector;

private Lights lights;

public HomeTheaterFacade(DVDPlayer dvdPlayer, Projector projector, Lights lights) {

this.dvdPlayer = dvdPlayer;

this.projector = projector;

this.lights = lights;

}

public void watchMovie(String movie) {

System.out.println("Get ready to watch a movie!");

lights.dim();

projector.on();

projector.setInput(dvdPlayer);

dvdPlayer.on();

dvdPlayer.play(movie);

}

public void endMovie() {

System.out.println("Shutting down the home theater");

dvdPlayer.off();

projector.off();

lights.brighten();

}

}

// Client code to test the Facade Design Pattern

public class FacadePatternExample {

public static void main(String[] args) {

DVDPlayer dvdPlayer = new DVDPlayer();

Projector projector = new Projector();

Lights lights = new Lights();

HomeTheaterFacade homeTheater = new HomeTheaterFacade(dvdPlayer, projector, lights);

// Watch a movie

homeTheater.watchMovie("Inception");

// End the movie

homeTheater.endMovie();

}

}

15.2

#Write a python program to make categorial values in numeric format

import pandas as pd

df=pd.read\_

csv('PlayTennis.csv')

print(df)

from sklearn.preprocessing import LabelEncoder

le=LabelEncoder()

label=le.fit\_transform(df['Play Tennis'])

print(label)

df.drop("Play Tennis",axis=1, inplace=True)

df["Play Tennis"]=label

print(df

15.3

Module.js

// modules.js

// Function to return today's date and time

function getCurrentDateTime() {

const currentDate = new Date();

return currentDate.toLocaleString();

}

// Export the function to make it available externally

module.exports = {

getCurrentDateTime: getCurrentDateTime

};

Server.js

// server.js

// Import the modules.js module

const myModule = require('./modules');

// Import the built-in http module

const http = require('http');

// Create a local server

const server = http.createServer((req, res) => {

res.writeHead(200, {'Content-Type': 'text/plain'});

// Use the function from modules.js to get the current date and time

const dateTime = myModule.getCurrentDateTime();

res.end(`Current Date and Time: ${dateTime}`);

});

// Set the server to listen on port 3000

const PORT = 3000;

server.listen(PORT, () => {

console.log(`Server running at http://localhost:${PORT}/`);

});