**COSC 4351**

**Name: Hugh Hoang**

800 points

**UML MVC Class Diagram Model to C#**, **Ruby**, **Python**, and **Php5** \*.

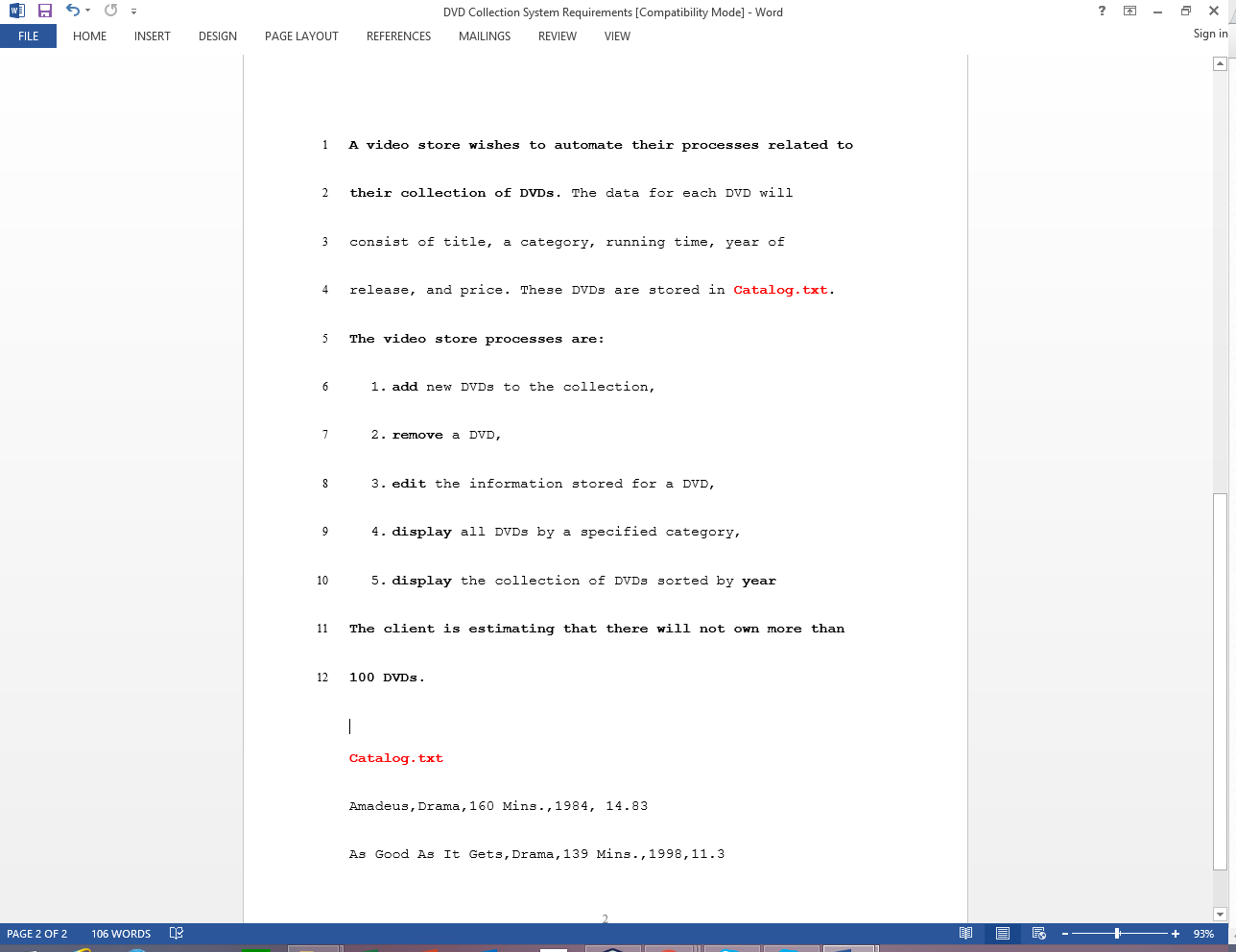
Enhance your **UML MVC Class Diagram** Model submitted for the UML Tutorial (add pseudocode for **main**, **showSelection**, UC**RestoreCatalog**, **restoreCatalog**, UC**AddDVD**, **addDVD**, UC**DisplayDVDByCategory**, **displayDVDByCategory,** UC**SaveCatalog**, **saveCatalog)** and convert it to the following Classes in **C#**, **Ruby**, **Python**, and **Php5**\*.

Please attach your new UML MVC Class Diagram to BB Assignment.

You must use an IDE.

If not MVC Classes you will not get any points.

**DVD Collection Application Requirements:**



1. (200 points)

**Given the above DVD Catalog Application above, convert the UML MVC Class Diagram Model to C# Classes and only implement the restore Catalog (using the given Catalog.txt), add DVD, display DVDs by category and save Catalog (using the given Catalog.txt).**

**(Must use an IDE, without it only half).**

**Answer:**

**Code here for each .cs file (one Class per file).**

**Full screen screenshot (output window, source window with first few lines of comments showing your name and Spring 2021)**

using System;

using System.IO;

//Hugh Hoang 1833106 Spring 2021

class DVD

{

string title, category, runtime, yor, price;

// Constructor that takes no arguments:

public DVD()

{

}

// Constructor that takes one argument:

public DVD(string intitle, string incategory, string inruntime, string inyor, string inprice)

{

title = intitle;

category = incategory;

runtime = inruntime;

yor = inyor;

price = inprice;

}

}

class DVDCatalog

{

public static void Main()

{

DVD[] currentcatalog = new DVD[100];

showSelection();

string option = Console.ReadLine();

if (option == "1")

{

using (StreamWriter w = File.AppendText("Catalog.txt"))

{

restoreCatalog(currentcatalog,w);

}

}

else if (option == "2")

{

using (StreamWriter w = File.AppendText("Catalog.txt"))

{

Console.WriteLine("New DVD's title name?");

string title = Console.ReadLine();

Console.WriteLine("New DVD's category name?");

string category = Console.ReadLine();

Console.WriteLine("New DVD's running time?");

string runtime = Console.ReadLine();

Console.WriteLine("New DVD's year of release?");

string yor = Console.ReadLine();

Console.WriteLine("New DVD's price?");

string price = Console.ReadLine();

string text = title + "," + category + "," + runtime + "," + yor + "," + price;

addDVD(text,w);

}

}

else if (option == "3")

{

using (StreamReader r = File.OpenText("Catalog.txt"))

{

Console.WriteLine("Display by what category?");

string categoryname = Console.ReadLine();

displayCategory(categoryname, r);

}

}

else if (option == "4")

{

using (StreamReader r = File.OpenText("Catalog.txt"))

{

currentcatalog= saveCatalog(currentcatalog, r);

}

}

else

{

Console.WriteLine($"Not valid option");

}

}

public static void showSelection()

{

Console.WriteLine("Welcome to DVD Catalog. Pick an option");

Console.WriteLine("1.restore Catalog");

Console.WriteLine("2.add DVD");

Console.WriteLine("3.display DVDs by category");

Console.WriteLine("4.save Catalog");

}

public static void restoreCatalog(DVD[] currentcatalog,TextWriter w)

{

//clears

System.IO.File.WriteAllText("Catalog.txt", string.Empty);

for(int i = 0; i < currentcatalog.Length; i++)

{

w.WriteLine(currentcatalog[i]);

}

}

public static void addDVD(string text, TextWriter w)

{

w.WriteLine(text);

}

public static void displayCategory(string categoryname, TextReader w)

{

string[] lines = File.ReadAllLines("Catalog.txt");

foreach (string line in lines)

{

int pFrom = line.IndexOf(",") + ",".Length;

int pTo = line.IndexOf(',', line.IndexOf(',') + 1);

String result = line.Substring(pFrom, pTo - pFrom);

if (result==categoryname)

{

Console.WriteLine(line);

}

}

}

public static DVD[] saveCatalog(DVD[] currentcatalog, TextReader w)

{

string[] lines = File.ReadAllLines("Catalog.txt");

int dvdnumber = 0;

foreach (string line in lines)

{

int indextitle = GetNthIndex(line, ',', 1);

int indexcategory = GetNthIndex(line, ',', 2);

int indexruntime = GetNthIndex(line, ',', 3);

int indexyor = GetNthIndex(line, ',', 4);

String title = line.Substring(0, indextitle);

String category = line.Substring(indextitle + 1, indexcategory - indextitle - 1);

String runtime = line.Substring(indexcategory + 1, indexruntime - indexcategory - 1);

String yor = line.Substring(indexruntime + 1, indexyor - indexruntime - 1);

String price = line.Substring(indexyor + 1, line.Length - indexyor - 1);

DVD currDVD = new DVD(title,category,runtime,yor,price);

currentcatalog[dvdnumber] = currDVD;

dvdnumber+= 1;

}

return currentcatalog;

}

public static int GetNthIndex(string s, char t, int n)

{

int count = 0;

for (int i = 0; i < s.Length; i++)

{

if (s[i] == t)

{

count++;

if (count == n)

{

return i;

}

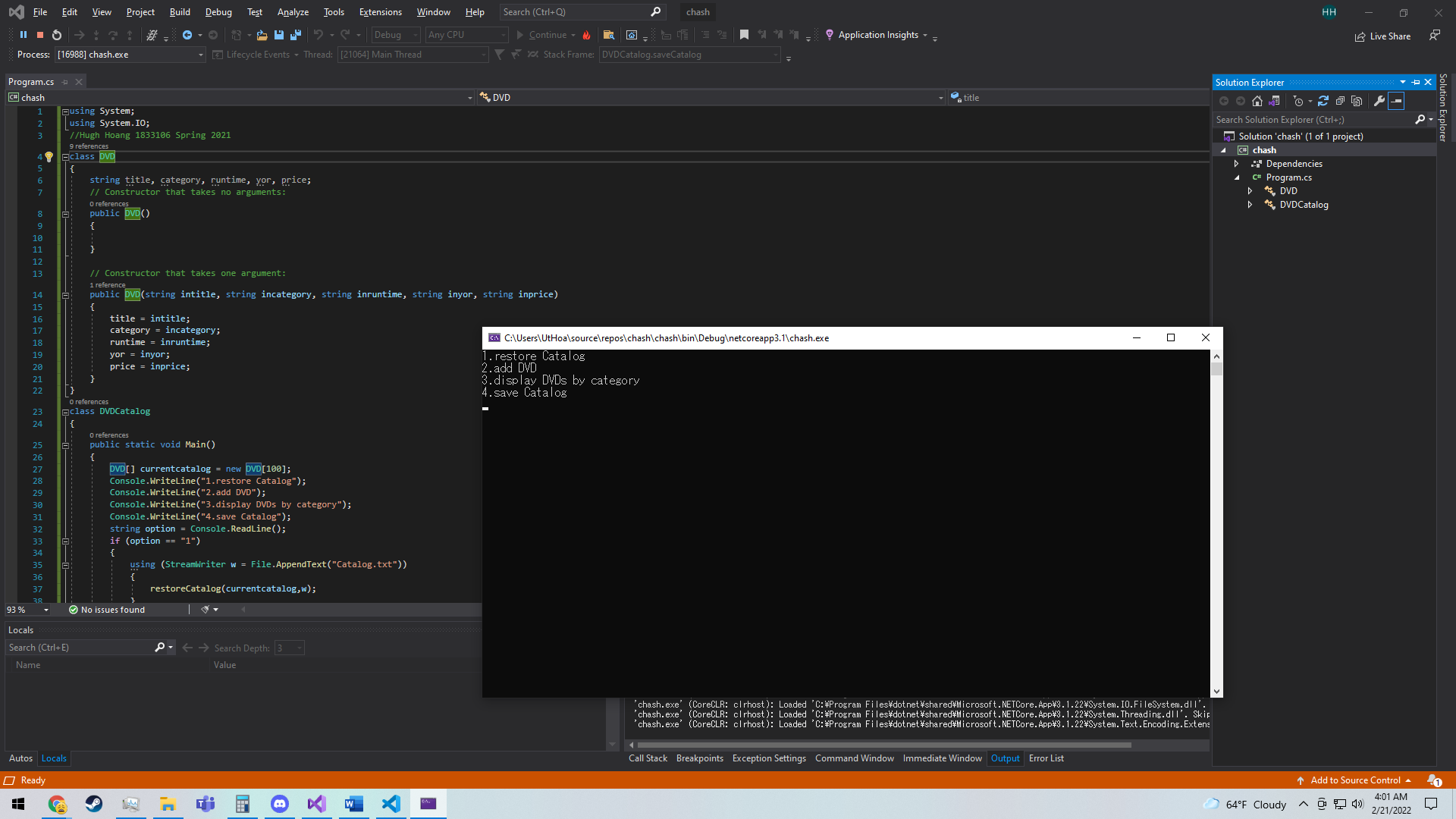
}

}

return -1;

}

}



2. (200 points)

**Given the above DVD Catalog Application above, convert the UML MVC Class Diagram Model to Ruby Classes and only implement the restore Catalog (using the given Catalog.txt), add DVD, display DVDs by category and save Catalog.**

**(Must use an IDE, without it only half).**

**Answer:**

**Code here for each .rb file (one Class per file).**

**Full screen screenshot (output window, source window with first few lines of comments showing your name and Spring 2021)**

# Hugh Hoang 1833106 Spring 2021

class DVD

{

    def initialize(title, category runtime, yor, price)

        @cust\_title = title

        @cust\_category = category

        @cust\_runtime = runtime

        @yor = inyor;

        @price = inprice;

     end

}

currentcatalog = Array.new(100)

showSelection()

option = gets.chomp

if (option == "1")

    restoreCatalog(currentcatalog);

elsif (option == "2")

    puts("New DVD's title name?");

    title = gets.chomp

    puts("New DVD's category name?");

    category = gets.chomp

    puts("New DVD's running time?");

    runtime = gets.chomp

    puts("New DVD's year of release?");

    yor = gets.chomp

    puts("New DVD's price?");

    price = gets.chomp

    text = title + "," + category + "," + runtime + "," + yor + "," + price;

    addDVD(text);

elsif (option == "3")

    puts("Display by what category?");

    categoryname = gets.chomp

    displayCategory(categoryname);

elsif (option == "4")

    currentcatalog= saveCatalog(currentcatalog);

else

    puts("Not valid option");

    def showSelection()

        puts("Welcome to DVD Catalog. Pick an option");

        puts("1.restore Catalog");

        puts("2.add DVD");

        puts("3.display DVDs by category");

        puts("4.save Catalog");

    def restoreCatalog(DVD[] currentcatalog)

        for i in currentcatalog do

            File.write("Catalog.txt", currentcatalog[i], mode: "a")

    def addDVD(string text)

        File.write("Catalog.txt", text, mode: "a")

    def displayCategory(string categoryname)

        File.readlines("Catalog.txt").each do |line|

            for line in lines:

                startindexcat=index\_of\_nth\_occorunce(line, ',', 1)

                endindexcat=index\_of\_nth\_occorunce(line, ',', 2)

                category = line[startindexcat + 1:endindexcat]

                if category==categoryname:

                    puts(line)

    def saveCatalog(DVD[] currentcatalog, TextReader w)

        string[] lines = File.ReadAllLines("Catalog.txt");

        int dvdnumber = 0;

        foreach (string line in lines)

            indextitle = index\_of\_nth\_occorunce(line, ',', 1);

            indexcategory = index\_of\_nth\_occorunce(line, ',', 2);

            indexruntime = index\_of\_nth\_occorunce(line, ',', 3);

            indexyor = index\_of\_nth\_occorunce(line, ',', 4);

            title = value[0..indextitle]

            category = value[indextitle + 1..indexcategory-1]

            runtime = value[indexcategory+1..indexruntime-1]

            yor = value[indexruntime+1..indexyor-1]

            price = value[indexyor+1..line.Length-1]

            DVD currDVD = new DVD(title,category,runtime,yor,price);

            currentcatalog[dvdnumber] = currDVD;

            dvdnumber+= 1;

        return currentcatalog;

    def index\_of\_nth\_occorunce(string, element, nth\_occurunce)

        count = 0

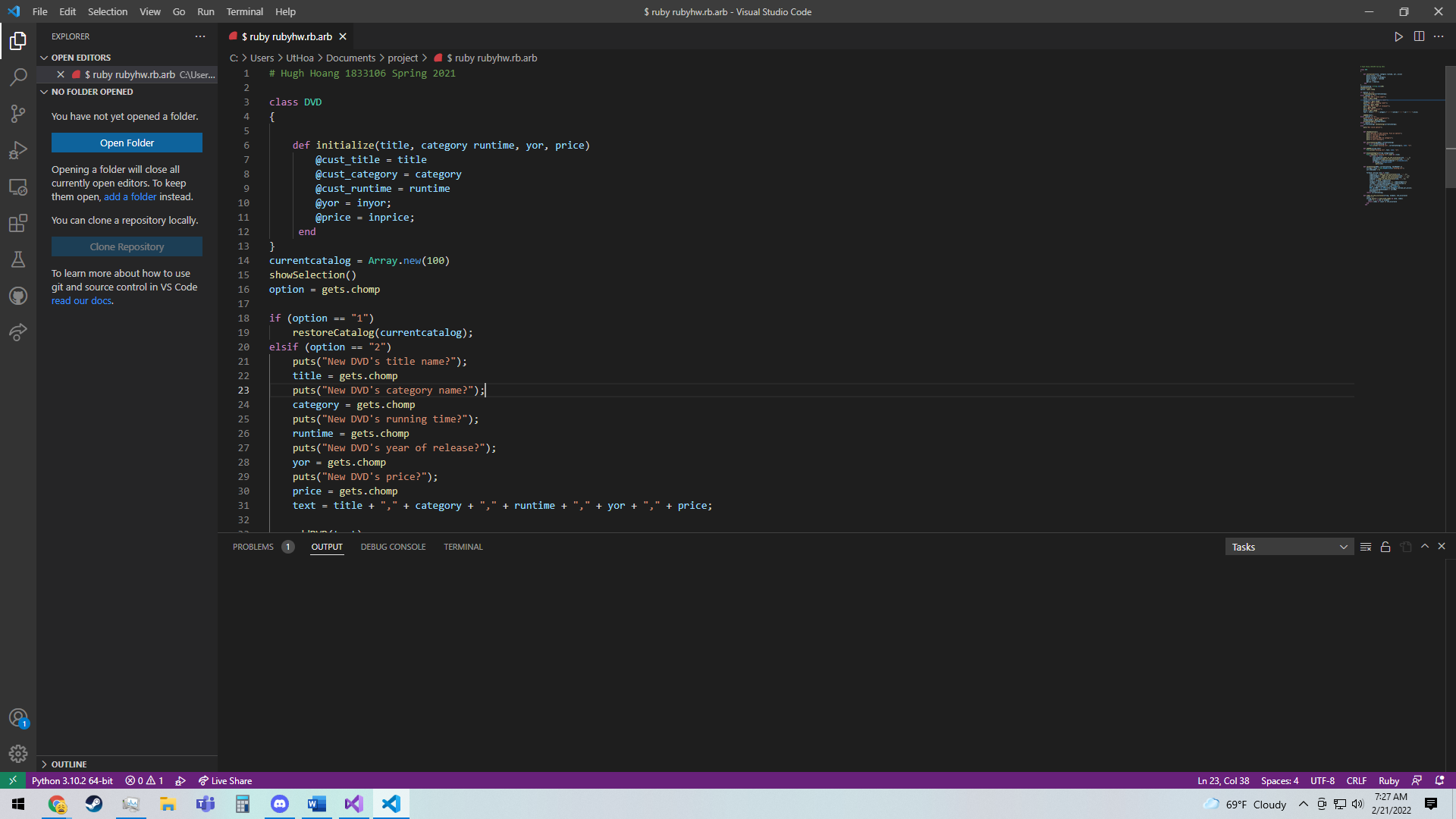
        string.split("").each\_with\_index do |elm, index|

          count += 1 if elm == element

          return index if count == nth\_occurunce

        end

      end



3. (200 points)

**Given the above DVD Catalog Application above, convert the UML MVC Class Diagram Model to Python Classes and only implement the restore Catalog (using the given Catalog.txt), add DVD, display DVDs by category and save Catalog.**

**(Must use an IDE, without it only half).**

**Answer:**

**Code here for each .py file (one Class per file).**

**Full screen screenshot (output window, source window with first few lines of comments showing your name and Spring 2021)**

#Hugh Hoang 1833106 Spring 2021

class DVD:

    def \_\_init\_\_(self, name, age,intitle, incategory, inruntime, inyor, inprice):

        self.title = intitle

        self.category = incategory

        self.runtime = inruntime

        self.yor = inyor

        self.price = inprice

def restoreCatalog(currentcatalog):

    file1 = open("Catalog.txt", "w")  # write mode

    for x in currentcatalog:

        file1.write("x \n")

    file1.close()

def addDVD():

    file1 = open("Catalog.txt", "a")  # append mode

    title = input("New DVD's title name?")

    category = input("ew DVD's category name?")

    runtime = input("New DVD's running time?")

    yor = input("New DVD's year of release?")

    price = input("New DVD's price?")

    text = title + "," + category + "," + runtime + "," + yor + "," + price

    file1.write(text+ "\n")

def displaycategory(categoryname):

    file1 = open("myfile.txt", "r")

    lines = file1.readlines()

    for line in lines:

        startindexcat=GetNthIndex(line, ',', 1)

        endindexcat=GetNthIndex(line, ',', 2)

        category = line[startindexcat + 1:endindexcat]

        if category==categoryname:

            file1.write(line+ "\n")

def savecatalog(category):

        file1 = open("myfile.txt", "r")

        lines = file1.readlines()

        dvdnumber=0

        for line in lines:

            indextitle = GetNthIndex(line, ',', 1)

            indexcategory = GetNthIndex(line, ',', 2)

            indexruntime = GetNthIndex(line, ',', 3)

            indexyor = GetNthIndex(line, ',', 4)

            title = line[0:indextitle]

            category = line[indextitle + 1:indexcategory - 1]

            runtime = line[indexcategory + 1:indexruntime - 1]

            yor = line[indexruntime + 1:indexyor - 1]

            price = line[indexyor + 1:line.length]

            currDVD = DVD(title,category,runtime,yor,price)

            currentcatalog[dvdnumber] = currDVD

            dvdnumber+= 1

def GetNthIndex(x, y, n):

        start = x.find(y)

        while start >= 0 and n > 1:

            start = x.find(y, start+len(y))

            n -= 1

        return start

currentcatalog=[]

print("1.restore Catalog")

print("2.add DVD")

print("3.display DVDs by category")

print("4.save Catalog")

option = input()

if option == "1":

    restoreCatalog(currentcatalog)

elif option == "2":

    addDVD()

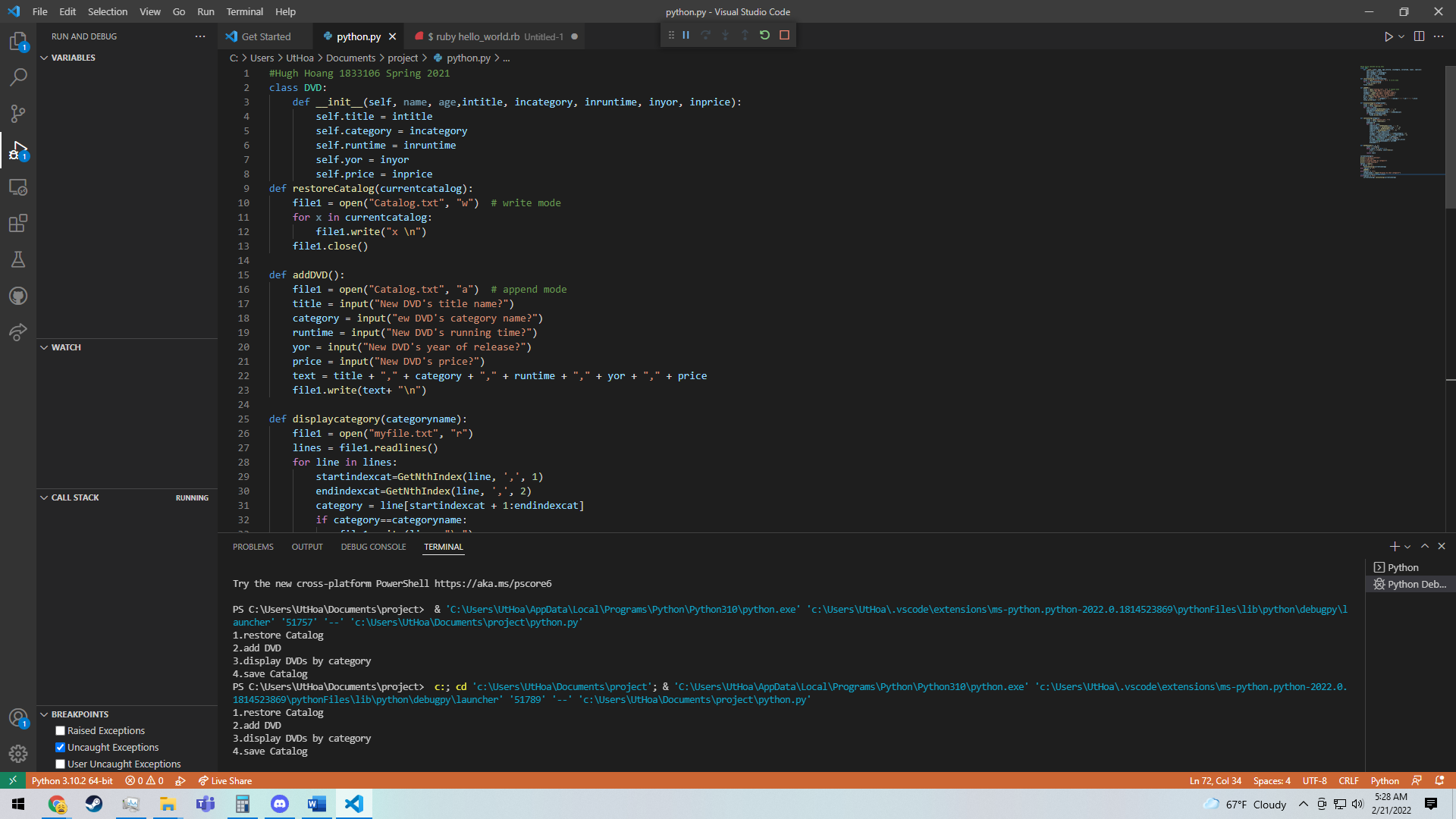
elif(option == "3"):

    categoryname = input("Display by what category?")

    displaycategory(categoryname)

elif(option == "4"):

    currentcatalog= savecatalog(currentcatalog)



4. (200 points)

**Given the above DVD Catalog Application above, convert the UML MVC Class Diagram Model to Php Classes and only implement the restore Catalog (using the given Catalog.txt), add DVD, display DVDs by category and save Catalog.**

**(Must use an IDE, without it only half).**

**Answer:**

**Code here for each .php file (one Class per file).**

**Full screen screenshot (output window, source window with first few lines of comments showing your name and Spring 2021)**

<?php

//Hugh Hoang 1833106 Spring 2021

class DVD {

    public $title;

    public $category;

    public $runtime;

    public $yor;

    public $price;

    function set\_title($title) {

        $this->title = $title;

      }

      function set\_category($category) {

        $this->category = $category;

      }

      function set\_runtime($runtime) {

        $this->runtime = $runtime;

      }

      function set\_yor($yor) {

        $this->yor = $yor;

      }

      function set\_price($price) {

        $this->price = $price;

      }

}

$foo = new DVD;

?>

<?php

//Hugh Hoang 1833106 Spring 2021

class DVDCatalog{

    public $dvds = array();

    function showSelection()

    {

        print("Welcome to DVD Catalog. Pick an option");

        print("1.restore Catalog");

        print("2.add DVD");

        print("3.display DVDs by category");

        print("4.save Catalog");

    }

function restoreCatalog($currentcatalog)

    {

        $myfile = fopen("Catalog.txt", "w");

        for($i = 0; $i <= currentcatalog.Length; $i++) {

            fwrite($myfile, $currentcatalog[i]);

        }

    }

    function addDVD($text)

    {

        $myfile = fopen("Catalog.txt", "w");

        $title = readline("New DVD's title name?");

        $category = readline("New DVD's category name?");

        $runtime = readline("New DVD's running time?");

        $yor = readline("New DVD's year of release?");

        $price = readline("New DVD's price?");

        $text = title + "," + category + "," + runtime + "," + yor + "," + price;

        fwrite($myfile, $currentcatalog[i]);

    }

    function displayCategory($categoryname)

    {

        $myfile = fopen("Catalog.txt", "w");

        echo fread($myfile,filesize("Catalog.txt.txt"));

        for($i = 0; $i <= currentcatalog.Length; $i++) {

            {

                if(currentcatalog[i]==categoryname){

                    print(myfile);

                }

            }

        }

    }

    function saveCatalog($currentcatalog)

    {

        $myfile = fopen("Catalog.txt", "w");

        $dvdnumber = 0;

        for($i = 0; $i <= currentcatalog.Length; $i++) {

            $indextitle = strposnth(line, ',', 1);

            $indexcategory = strposnth(line, ',', 2);

            $indexruntime = strposnth(line, ',', 3);

            $indexyor = strposnth(line, ',', 4);

            $title = substr(0, indextitle);

            $category = substr(indextitle + 1, indexcategory - indextitle - 1);

            $runtime = substr(indexcategory + 1, indexruntime - indexcategory - 1);

            $yor = substr(indexruntime + 1, indexyor - indexruntime - 1);

            $price = substr(indexyor + 1, line.Length - indexyor - 1);

            currDVD = new DVD(title,category,runtime,yor,price);

            currentcatalog[dvdnumber] = currDVD;

            dvdnumber+= 1;

        }

        return currentcatalog;

    }

    function strposnth($haystack, $needle, $nth=1, $insenstive=0)

    {

       if ($insenstive) {

           $haystack=strtolower($haystack);

           $needle=strtolower($needle);

       $count=substr\_count($haystack,$needle);

      if ($count<1 || $nth > $count) return false;

      for($i=0,$pos=0,$len=0;$i<$nth;$i++)

       {

           $pos=strpos($haystack,$needle,$pos+$len);

           if ($i==0) $len=strlen($needle);

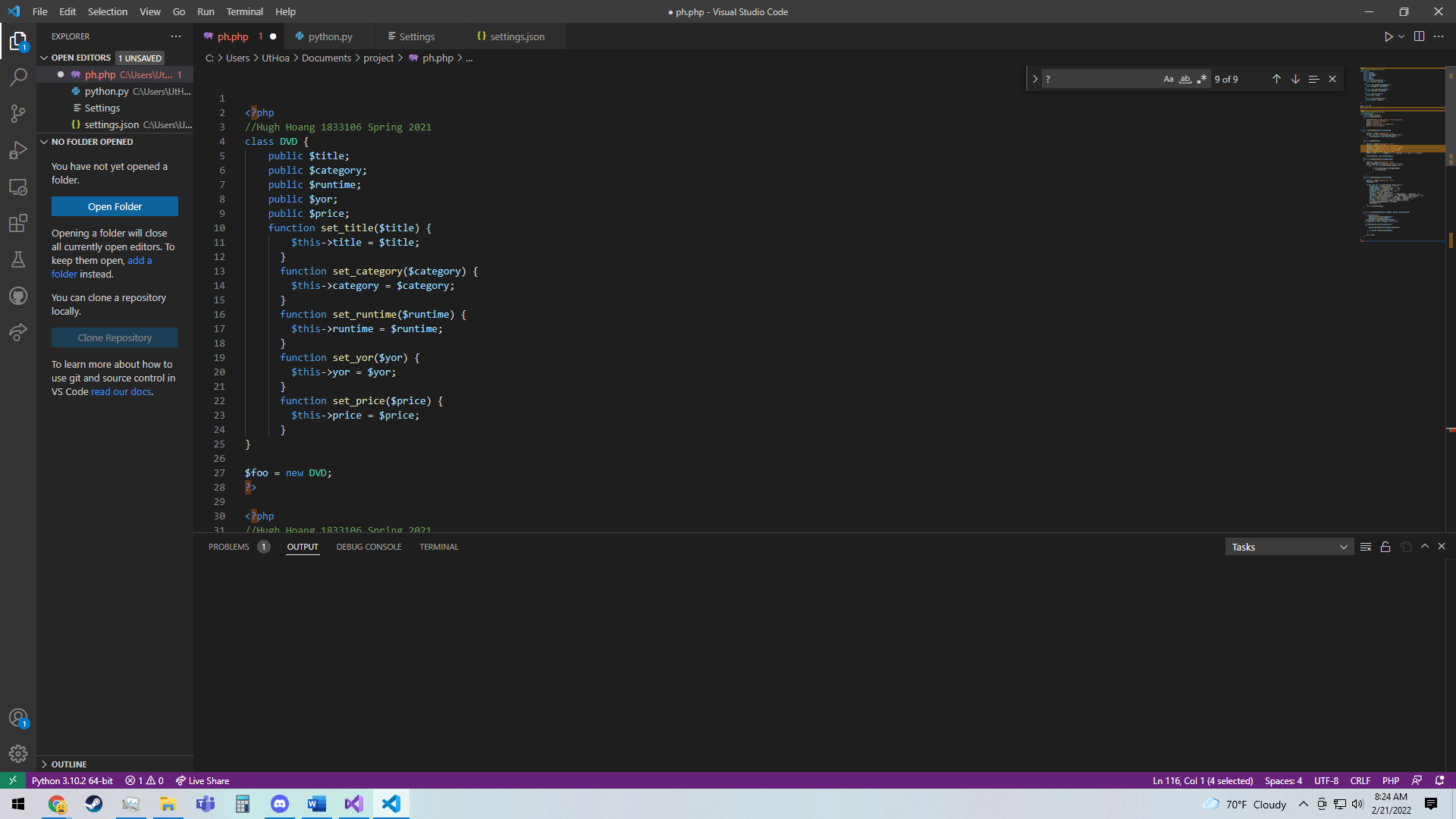
         }

       return $pos;

    }

}

//?>

**NOTES: Ruby, Python, Php are supported in NetBeans 6.9.1 IDE (or newer versions by importing plugins). Any other IDEs are OK.**

**NOTES: C# is supported by Visual Studio IDE.**

**\* Java is supported in NetBeans IDE. It is ok if one of your UML to OO Language is Java.**