Tên: Phạm Dương Minh Nhật

Mã sinh viên: 19IT182

# **EXERCISE1**

using System.ComponentModel.DataAnnotations;

namespace OOP3

{

class Student

{

private int SID;

private string TenSV;

private string Khoa;

private double Diem;

public void DisplayStudent()

{

Console.WriteLine("MSSV: {0}", this.SID);

Console.WriteLine("Ten SV: {0}", this.TenSV);

Console.WriteLine("Khoa: {0}", this.Khoa);

Console.WriteLine("Diem TB: {0}", this.Diem);

}

public Student(Student st)

{

SID = st.SID;

TenSV = st.TenSV;

Khoa = st.Khoa;

Diem = st.Diem;

}

public Student(int id, string name, string fac, double mark)

{

SID = id;

TenSV = name;

Khoa = fac;

Diem = mark;

}

public Student()

{

SID = 1;

TenSV = "Minh Nhat";

Khoa = "KHMT";

Diem = 8.5;

}

public int StudentID

{

get { return SID; }

set { SID = value; }

}

public string Name

{

get { return TenSV; }

set { TenSV = value; }

}

public string Faculty

{

get { return Khoa; }

set { Khoa = value; }

}

public double Mark

{

get { return Diem; }

set { Diem = value; }

}

}

class Program

{

public static void Main()

{

Student[] DSSV;

int n;

Console.Write("Nhap so luong SV:");

n = Convert.ToInt32(Console.ReadLine());

DSSV = new Student[n];

Console.WriteLine("\n ====NHAP DS SINH VIEN====");

for (int i = 0; i < n; i++)

{

DSSV[i] = new Student();

Console.Write("Nhap MaSV {0}:", i + 1);

DSSV[i].StudentID = Convert.ToInt32(Console.ReadLine());

Console.Write("Ho ten SV:");

DSSV[i].Name = Console.ReadLine();

Console.Write("Nhap khoa:");

DSSV[i].Faculty = Console.ReadLine();

Console.Write("Nhap Diem TB:");

DSSV[i].Mark = Convert.ToDouble(Console.ReadLine());

}

Console.WriteLine("\n ====XUAT DS SINH VIEN====");

foreach (Student sv in DSSV)

{

sv.DisplayStudent();

}

Console.ReadLine();

}

}

}

Text

Description automatically generated with low confidence

# **EXERCISE2**

using System.ComponentModel.DataAnnotations;

using System.Diagnostics.Metrics;

using System.Numerics;

namespace OOP3

{

class Student

{

public void DisplayStudent()

{

Console.WriteLine("MSSV: {0}", this.StudentID);

Console.WriteLine("Ten SV: {0}", this.Name);

Console.WriteLine("Khoa: {0}", this.Faculty);

Console.WriteLine("Diem TB: {0}", this.Mark);

}

public Student(Student st)

{

StudentID = st.StudentID;

Name = st.Name;

Faculty = st.Faculty;

Mark = st.Mark;

}

public Student(int id, string name, string fac, double mark)

{

StudentID = id;

Name = name;

Faculty = fac;

Mark = mark;

}

public Student()

{

StudentID = 1;

Name = "Minh Nhat";

Faculty = "KHMT";

Mark = 8.5;

}

public int StudentID

{

get; set;

}

public string Name

{

get; set;

}

public string Faculty

{

get; set;

}

public double Mark

{

get; set;

}

}

class Program

{

public static void Main()

{

while (true)

{

Console.WriteLine("Welcome to Minh Nhat EXERCISES OOP2 ABSTRACT");

Console.WriteLine("1.Nhap 1 Sinh Vien");

Console.WriteLine("2.Nhap Danh Sach Sinh Vien");

int chooseFunc = Convert.ToInt32(Console.ReadLine());

if (chooseFunc > 3 || chooseFunc < 1)

{

Console.WriteLine("Please re-Enter Function");

chooseFunc = Convert.ToInt32(Console.ReadLine());

} else

{

switch (chooseFunc)

{

case 1:

Xuat1SinhVien(Nhap1SV());

break;

case 2:

Console.WriteLine("Nhap so luong sinh vien");

int n = Convert.ToInt32(Console.ReadLine());

XuatDS(NhapDS(n));

break;

}

}

}

static Student Nhap1SV()

{

Student student = new Student();

student = new Student();

Console.Write("Nhap MaSV {0}:");

student.StudentID = Convert.ToInt32(Console.ReadLine());

Console.Write("Ho ten SV:");

student.Name = Console.ReadLine();

Console.Write("Nhap khoa:");

student.Faculty = Console.ReadLine();

Console.Write("Nhap Diem TB:");

student.Mark = Convert.ToDouble(Console.ReadLine());

return student;

}

static Array NhapDS(int n)

{

Student[] DSSV;

DSSV = new Student[n];

Console.WriteLine("\n ====NHAP DS SINH VIEN====");

for (int i = 0; i < n; i++)

{

DSSV[i] = new Student();

Console.Write("Nhap MaSV {0}:", i + 1);

DSSV[i].StudentID = Convert.ToInt32(Console.ReadLine());

Console.Write("Ho ten SV:");

DSSV[i].Name = Console.ReadLine();

Console.Write("Nhap khoa:");

DSSV[i].Faculty = Console.ReadLine();

Console.Write("Nhap Diem TB:");

DSSV[i].Mark = Convert.ToDouble(Console.ReadLine());

}

return DSSV;

}

static void Xuat1SinhVien(Student student)

{

Console.WriteLine("Sinh Vien Da Nhap: ");

Console.WriteLine("MSSV: {0}", student.StudentID);

Console.WriteLine("Ten SV: {0}", student.Name);

Console.WriteLine("Khoa: {0}", student.Faculty);

Console.WriteLine("Diem TB: {0}", student.Mark);

}

static void XuatDS(Array DSSV)

{

Console.WriteLine("Sinh Vien Da Nhap: ");

foreach (Student student in DSSV)

{

student.DisplayStudent();

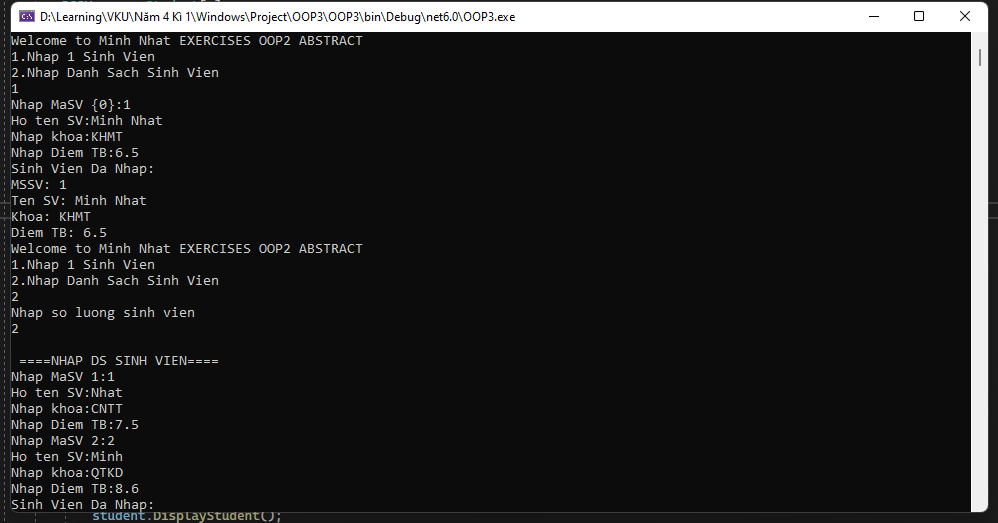
}

}

}

}

}



Text

Description automatically generated

# **EXERCISE3**

## **LIST**

using System.ComponentModel.DataAnnotations;

using System.Diagnostics.Metrics;

using System.Numerics;

namespace OOP3

{

class Program

{

public static void Main()

{

while (true)

{

Console.WriteLine("Welcome to Minh Nhat EXERCISES OOP2 ABSTRACT");

Console.WriteLine("1.Nhap 1 Sinh Vien");

Console.WriteLine("2.Nhap Danh Sach Sinh Vien");

int chooseFunc = Convert.ToInt32(Console.ReadLine());

if (chooseFunc > 3 || chooseFunc < 1)

{

Console.WriteLine("Please re-Enter Function");

chooseFunc = Convert.ToInt32(Console.ReadLine());

} else

{

switch (chooseFunc)

{

case 1:

Xuat1SinhVien(Nhap1SV());

break;

case 2:

Console.WriteLine("Nhap so luong sinh vien");

int n = Convert.ToInt32(Console.ReadLine());

XuatDS(NhapDS(n));

break;

}

}

}

static Student Nhap1SV()

{

Student student = new Student();

student = new Student();

Console.Write("Nhap MaSV:");

student.StudentID = Convert.ToInt32(Console.ReadLine());

Console.Write("Ho ten SV:");

student.Name = Console.ReadLine();

Console.Write("Nhap khoa:");

student.Faculty = Console.ReadLine();

Console.Write("Nhap Diem TB:");

student.Mark = Convert.ToDouble(Console.ReadLine());

return student;

}

static List<Student> NhapDS(int n)

{

Student[] DSSV;

DSSV = new Student[n];

Console.WriteLine("\n ====NHAP DS SINH VIEN====");

for (int i = 0; i < n; i++)

{

DSSV[i] = new Student();

Console.Write("Nhap MaSV {0}:", i + 1);

DSSV[i].StudentID = Convert.ToInt32(Console.ReadLine());

Console.Write("Ho ten SV:");

DSSV[i].Name = Console.ReadLine();

Console.Write("Nhap khoa:");

DSSV[i].Faculty = Console.ReadLine();

Console.Write("Nhap Diem TB:");

DSSV[i].Mark = Convert.ToDouble(Console.ReadLine());

}

var ListStudent = new List<Student>();

ListStudent.AddRange(DSSV);

return ListStudent;

}

static void Xuat1SinhVien(Student student)

{

Console.WriteLine("Sinh Vien Da Nhap: ");

Console.WriteLine("MSSV: {0}", student.StudentID);

Console.WriteLine("Ten SV: {0}", student.Name);

Console.WriteLine("Khoa: {0}", student.Faculty);

Console.WriteLine("Diem TB: {0}", student.Mark);

}

static void XuatDS(List<Student> DSSV)

{

Console.WriteLine("Sinh Vien Da Nhap: ");

foreach (Student student in DSSV)

{

student.DisplayStudent();

}

}

}

}

class People

{

public string Name

{

get; set;

}

public People(People peo)

{

Name = peo.Name;

}

public People(string name)

{

Name = name;

}

public People()

{

Name = "Minh Nhat";

}

}

class Student:People

{

public void DisplayStudent()

{

Console.WriteLine("MSSV: {0}", this.StudentID);

Console.WriteLine("Ten SV: {0}", this.Name);

Console.WriteLine("Khoa: {0}", this.Faculty);

Console.WriteLine("Diem TB: {0}", this.Mark);

}

public Student(Student st)

{

StudentID = st.StudentID;

Name = st.Name;

Faculty = st.Faculty;

Mark = st.Mark;

}

public Student(int id, string name, string fac, double mark)

{

StudentID = id;

Name = name;

Faculty = fac;

Mark = mark;

}

public Student()

{

StudentID = 1;

Name = "Minh Nhat";

Faculty = "KHMT";

Mark = 8.5;

}

public int StudentID

{

get; set;

}

public string Faculty

{

get; set;

}

public double Mark

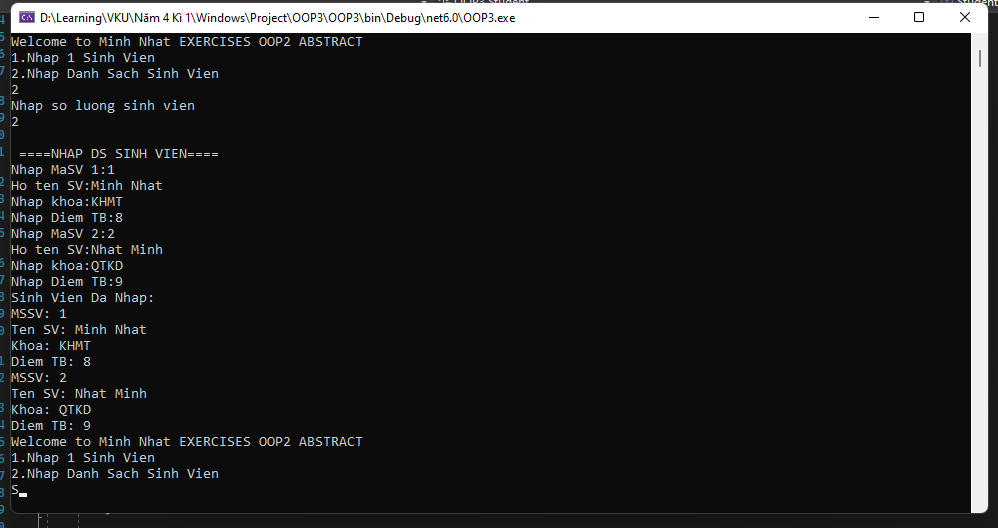
{

get; set;

}

}

}



## **ARRAYLIST**

class Program

{

public static void Main()

{

while (true)

{

Console.WriteLine("Welcome to Minh Nhat EXERCISES OOP2 ABSTRACT");

Console.WriteLine("1.Nhap 1 Sinh Vien");

Console.WriteLine("2.Nhap Danh Sach Sinh Vien");

int chooseFunc = Convert.ToInt32(Console.ReadLine());

if (chooseFunc > 3 || chooseFunc < 1)

{

Console.WriteLine("Please re-Enter Function");

chooseFunc = Convert.ToInt32(Console.ReadLine());

} else

{

switch (chooseFunc)

{

case 1:

Xuat1SinhVien(Nhap1SV());

break;

case 2:

Console.WriteLine("Nhap so luong sinh vien");

int n = Convert.ToInt32(Console.ReadLine());

XuatDS(NhapDS(n));

break;

}

}

}

static Student Nhap1SV()

{

Student student = new Student();

student = new Student();

Console.Write("Nhap MaSV:");

student.StudentID = Convert.ToInt32(Console.ReadLine());

Console.Write("Ho ten SV:");

student.Name = Console.ReadLine();

Console.Write("Nhap khoa:");

student.Faculty = Console.ReadLine();

Console.Write("Nhap Diem TB:");

student.Mark = Convert.ToDouble(Console.ReadLine());

return student;

}

static ArrayList NhapDS(int n)

{

Student[] DSSV;

DSSV = new Student[n];

Console.WriteLine("\n ====NHAP DS SINH VIEN====");

for (int i = 0; i < n; i++)

{

DSSV[i] = new Student();

Console.Write("Nhap MaSV {0}:", i + 1);

DSSV[i].StudentID = Convert.ToInt32(Console.ReadLine());

Console.Write("Ho ten SV:");

DSSV[i].Name = Console.ReadLine();

Console.Write("Nhap khoa:");

DSSV[i].Faculty = Console.ReadLine();

Console.Write("Nhap Diem TB:");

DSSV[i].Mark = Convert.ToDouble(Console.ReadLine());

}

var ArrayListStudent = new ArrayList();

ArrayListStudent.AddRange(DSSV);

return ArrayListStudent;

}

static void Xuat1SinhVien(Student student)

{

Console.WriteLine("Sinh Vien Da Nhap: ");

Console.WriteLine("MSSV: {0}", student.StudentID);

Console.WriteLine("Ten SV: {0}", student.Name);

Console.WriteLine("Khoa: {0}", student.Faculty);

Console.WriteLine("Diem TB: {0}", student.Mark);

}

static void XuatDS(ArrayList DSSV)

{

Console.WriteLine("Sinh Vien Da Nhap: ");

foreach (Student student in DSSV)

{

student.DisplayStudent();

}

}

}

}

Text

Description automatically generated

# **EXERCISE4**

using System;

using System.Collections;

using System.ComponentModel.DataAnnotations;

using System.Diagnostics.Metrics;

using System.Numerics;

using System.Xml.Linq;

namespace OOP3

{

class Program

{

public static void Main()

{

while (true)

{

Console.WriteLine("Welcome to Minh Nhat EXERCISES OOP3");

Console.WriteLine("1.Nhap Danh Sach Nhan Vien");

Console.WriteLine("2.Nhap Danh Sach Nha Quan Li");

Console.WriteLine("3.Nhap Danh Sach Nha Khoa Hoc");

int chooseFunc = Convert.ToInt32(Console.ReadLine());

double luong = 10000;

int bacLuongQL = 150;

int bacLuongKH = 170;

if (chooseFunc > 3 || chooseFunc < 1)

{

Console.WriteLine("Please re-Enter Function");

chooseFunc = Convert.ToInt32(Console.ReadLine());

} else

{

switch (chooseFunc)

{

case 1:

Console.WriteLine("Nhap so luong nhan vien");

int n = Convert.ToInt32(Console.ReadLine());

XuatDSNV(NhapDSNV(n, luong));

break;

case 2:

Console.WriteLine("Nhap so luong nha quan li");

int q = Convert.ToInt32(Console.ReadLine());

XuatDSQL(NhapDSNQL(q, bacLuongQL));

break;

case 3:

Console.WriteLine("Nhap so luong nha khoa hoc");

int k = Convert.ToInt32(Console.ReadLine());

XuatDSKH(NhapDSNKH(k, bacLuongKH));

break;

}

}

}

static ArrayList NhapDSNV(int n, double luong)

{

NhanVien[] DSNV;

DSNV = new NhanVien[n];

Console.WriteLine("\n ====Nhap DS Nhan Vien====");

for (int i = 0; i < n; i++)

{

DSNV[i] = new NhanVien();

Console.Write("Nhap Ten NV:");

DSNV[i].HoTen = Console.ReadLine();

Console.Write("Nam sinh NV:");

DSNV[i].NamSinh = Console.ReadLine();

Console.Write("Bang Cap NV:");

DSNV[i].BangCap = Console.ReadLine();

DSNV[i].Luong = luong;

}

var ArrayListNV = new ArrayList();

ArrayListNV.AddRange(DSNV);

return ArrayListNV;

}

static void XuatDSNV(ArrayList DSNV)

{

Console.WriteLine("Nhan Vien Da Nhap: ");

double tongLuong = 0;

foreach (NhanVien nv in DSNV)

{

nv.DisplayNV();

tongLuong = tongLuong + nv.Luong;

}

Console.WriteLine("Tong Luong Chi Tra Cho Nhan Vien: {0} ", tongLuong);

}

static ArrayList NhapDSNQL(int n, int bacLuongQL)

{

NhaQuanLi[] DSQL;

DSQL = new NhaQuanLi[n];

Console.WriteLine("\n ====Nhap DS Quan Li====");

for (int i = 0; i < n; i++)

{

DSQL[i] = new NhaQuanLi();

Console.Write("Nhap Ten QL:");

DSQL[i].HoTen = Console.ReadLine();

Console.Write("Nam sinh QL:");

DSQL[i].NamSinh = Console.ReadLine();

Console.Write("Bang Cap QL:");

DSQL[i].BangCap = Console.ReadLine();

Console.Write("Chuc Vu QL:");

DSQL[i].ChucVu = Console.ReadLine();

Console.Write("Ngay Cong QL:");

DSQL[i].NgayCong = Convert.ToInt32(Console.ReadLine());

DSQL[i].BacLuong = bacLuongQL;

DSQL[i].Luong = Convert.ToDouble(DSQL[i].NgayCong \* DSQL[i].BacLuong);

}

var ArrayListQL = new ArrayList();

ArrayListQL.AddRange(DSQL);

return ArrayListQL;

}

static void XuatDSQL(ArrayList DSQL)

{

Console.WriteLine("Quan Li Da Nhap: ");

double tongLuong = 0;

foreach (NhaQuanLi ql in DSQL)

{

ql.DisplayNQL();

tongLuong = tongLuong + ql.Luong;

}

Console.WriteLine("Tong Luong Chi Tra Cho Quan Li: {0} ", tongLuong);

}

static ArrayList NhapDSNKH(int n, int bacLuongKH)

{

NhaKhoaHoc[] DSKH;

DSKH = new NhaKhoaHoc[n];

Console.WriteLine("\n ====Nhap DS Nha Khoa Hoc====");

for (int i = 0; i < n; i++)

{

DSKH[i] = new NhaKhoaHoc();

Console.Write("Nhap Ten KH:");

DSKH[i].HoTen = Console.ReadLine();

Console.Write("Nam sinh KH:");

DSKH[i].NamSinh = Console.ReadLine();

Console.Write("Bang Cap KH:");

DSKH[i].BangCap = Console.ReadLine();

Console.Write("Chuc Vu KH:");

DSKH[i].ChucVu = Console.ReadLine();

Console.Write("So bai bao nha KH da dang:");

DSKH[i].SoBaiBao = Convert.ToInt32(Console.ReadLine());

Console.Write("Ngay Cong KH:");

DSKH[i].NgayCong = Convert.ToInt32(Console.ReadLine());

DSKH[i].BacLuong = bacLuongKH;

DSKH[i].Luong = Convert.ToDouble(DSKH[i].NgayCong \* DSKH[i].BacLuong);

}

var ArrayListKH = new ArrayList();

ArrayListKH.AddRange(DSKH);

return ArrayListKH;

}

static void XuatDSKH(ArrayList DSKH)

{

Console.WriteLine("Nha Khoa Hoc Da Nhap: ");

double tongLuong = 0;

foreach (NhaKhoaHoc ql in DSKH)

{

ql.DisplayNKH();

tongLuong = tongLuong + ql.Luong;

}

Console.WriteLine("Tong Luong Chi Tra Cho Nha Khoa Hoc: {0} ", tongLuong);

}

}

}

class NhanVien

{

public string HoTen

{

get; set;

}

public string NamSinh

{

get; set;

}

public string BangCap

{

get; set;

}

public double Luong

{

get; set;

}

public NhanVien(NhanVien nv)

{

HoTen = nv.HoTen;

NamSinh = nv.NamSinh;

BangCap = nv.BangCap;

Luong = nv.Luong;

}

public NhanVien(string hoTen, string namSinh, string bangCap, double luong)

{

HoTen = hoTen;

NamSinh = namSinh;

BangCap = bangCap;

Luong = luong;

}

public NhanVien()

{

}

public void DisplayNV()

{

Console.WriteLine("Ho Ten Nhan Vien {0}", this.HoTen);

Console.WriteLine("Nam Sinh {0}", this.NamSinh);

Console.WriteLine("Bang Cap {0}", this.BangCap);

Console.WriteLine("Luong {0}", this.Luong);

}

}

class NhaQuanLi : NhanVien

{

public string ChucVu

{

get; set;

}

public int NgayCong

{

get; set;

}

public int BacLuong

{

get; set;

}

public NhaQuanLi(string hoTen, string namSinh, string bangCap, string chucVu, int ngayCong, int bacLuong) : base(hoTen, namSinh, bangCap, 2000)

{

this.ChucVu = chucVu;

this.NgayCong = ngayCong;

this.BacLuong = bacLuong;

/\*this.Luong = Convert.ToDouble(ngayCong \* bacLuong);\*/

}

public NhaQuanLi()

{

}

public void DisplayNQL()

{

Console.WriteLine("Ho Ten {0}", this.HoTen);

Console.WriteLine("Nam Sinh {0}", this.NamSinh);

Console.WriteLine("Bang Cap {0}", this.BangCap);

Console.WriteLine("Chuc Vu {0}", this.ChucVu);

Console.WriteLine("Ngay Cong {0}", this.NgayCong);

Console.WriteLine("Bac Luong {0}", this.BacLuong);

Console.WriteLine("Luong {0}", this.Luong);

}

}

class NhaKhoaHoc:NhaQuanLi

{

public int SoBaiBao

{

get; set;

}

public NhaKhoaHoc(string hoTen, string namSinh, string bangCap, string chucVu, int ngayCong, int bacLuong, int soBaiBao) : base(hoTen, namSinh, bangCap, chucVu, ngayCong, bacLuong)

{

this.SoBaiBao = soBaiBao;

}

public NhaKhoaHoc()

{

}

public void DisplayNKH()

{

Console.WriteLine("Ho Ten {0}", this.HoTen);

Console.WriteLine("Nam Sinh {0}", this.NamSinh);

Console.WriteLine("Bang Cap {0}", this.BangCap);

Console.WriteLine("Chuc Vu {0}", this.ChucVu);

Console.WriteLine("So bai bao {0}", this.SoBaiBao);

Console.WriteLine("Ngay Cong {0}", this.NgayCong);

Console.WriteLine("Bac Luong {0}", this.BacLuong);

Console.WriteLine("Luong {0}", this.Luong);

}

}

}

Text

Description automatically generated

Text

Description automatically generated with medium confidence

Text

Description automatically generated