

LAB – 121
LABORATORIO DE INF – 121
“PROGRAMACION ORIENTADA A OBJETOS
CLASES, OBJETOS, METODOS”
Gestión II/2016

Docente: Lic. Marcelo Aruquipa
Aux. Fabio Laura
Aux. Pamela Choque

PARTE I IMPEMENTACION DE CLASES E INSTANCIACION

1. ESTRUCTURA DE UNA CLASE EN JAVA Y C#

Una clase en Java y C# debe tener la siguiente estructura: **YYY.java**

El nombre de la clase siempre en Mayúsculas y en singular

El nombre de los métodos:

<i>Java</i>	<i>CSharp C#</i>
<i>Debe empezar en minúsculas</i>	<i>Debe comenzar con mayúsculas</i>

```
tipoAcceso class YYY
{
    //se definen los atributos
    tipoAcceso tipoDato nomAtributo;
    ...
    ...
    ...

    //se definen los metodos
    // método que no devuelve
    tipoAcceso void nomMetodoA() {
        ...
        ...
    }
    // método que devuelve
    tipoAcceso tipoDev nomMetodoB(tipoDato param1, tipoDato param2){
        ...
        ...
        return valor;
    }
}
```

2. TIPOS DE ACCESO

Tipos de Acceso que puede tener un miembro atributo o método:

- **public +**
Acceso a él desde cualquier clase o instancia sin importar el paquete o procedencia de ésta.
- **protected #**
Acceso desde la misma clase, clases del mismo paquete y clases que hereden de ella (incluso en diferentes paquetes).
- **private -**
Puede ser accedido únicamente por la misma clase es decir por sus métodos o constructores nada más.

3. THIS

El **this** es una palabra reservada del lenguaje de programación

¿Donde se puede usar el this?

Uso de la palabra THIS sirve para hacer referencia a un método o propiedad del objeto actual.

¿Donde se puede usar el this?

Puede referirse a cualquier miembro del objeto actual desde dentro de un método de instancia o un constructor.

4. IMPLEMENTACION EN JAVA Y C#

En el siguiente ejemplo se muestra la interacción de objetos de una sola clase.

1. Ejemplo.

Implementar la clase Reloj. Cada objeto de esta clase representa una hora específica del día, almacenando las horas, minutos y segundos como enteros. Se ha de incluir un constructor, métodos de acceso y asignación, un método adelantar(int h, int m, int s) para adelantar la hora actual de un objeto existente, un método sincronizar(Reloj reloj) que sincronizará la hora actual de un objeto existente con otro objeto de tipo Reloj previamente instanciado y un método mostrar().

- Elabore el diagrama de clases.
- Implemente la clase Reloj.
- En el programa principal, demuestre con un ejemplo el uso de los métodos adelantar, sincronizar y mostrar.

SOLUCION.-

a)

RELOJ
-horas : int -minutos : int -segundos : int
+getters, setters() +adelantar() : void +sincronizar() : void

b)

Reloj.cs	Reloj.java
<pre>using System; namespace Guia1_LAB121 { public class Reloj{ private int horas; private int minutos; private int segundos; public Reloj(){ horas = 0; minutos = 0; segundos = 0; } public Reloj(int horas, int minutos, int segundos) { this.horas = horas; this.minutos = minutos; this.segundos = segundos; } public int Horas { get { return horas; } set { horas = value; } } } }</pre>	<pre>public class Reloj { private int horas; private int minutos; private int segundos; public Reloj() { horas = 0; minutos = 0; segundos = 0; } public Reloj(int horas, int minutos, int segundos) { this.horas = horas; this.minutos = minutos; this.segundos = segundos; } public int getHoras() {return horas; } public int getMinutos() { return minutos; } public int getSegundos() { return segundos; } public void setHoras(int horas) { this.horas = horas; } }</pre>

<pre> public int Minutos { get { return minutos; } set { minutos = value; } } public int Segundos { get { return segundos; } set { segundos = value; } } } //c) public void adelantar(int h, int m, int s){ int acs, acm; acs = acm = 0; acs = (segundos+s)/60; segundos = (segundos+s)%60; acm = (minutos + m + acs)/60; minutos = (minutos + m + acs)%60; horas = (horas + h + acm)%24; } public void sincronizar(Reloj B){ horas = B.Horas; minutos = B.Minutos; segundos = B.Segundos; } } public void mostrar(){ Console.WriteLine("Mostrando datos del reloj..."); Console.WriteLine("horas: "+horas); Console.WriteLine("minutos : "+minutos); Console.WriteLine("segundos: "+segundos); } } } </pre>	<pre> public void setMinutos(int minutos) { this.minutos = minutos; } public void setSegundos(int segundos) { this.segundos = segundos; } } //c) public void adelantar(int h, int m, int s) { int acs, acm; acs = acm = 0; acs = (segundos + s) / 60; segundos = (segundos + s) % 60; acm = (minutos + m + acs) / 60; minutos = (minutos + m + acs) % 60; horas = (horas + h + acm) % 24; } public void sincronizar(Reloj B) { horas = B.gethoras(); minutos = B.getminutos(); segundos = B.getsegundos(); } public void mostrar() { System.out.println("Mostrando datos del reloj..."); System.out.println("horas: " + horas); System.out.println("minutos : " + minutos); System.out.println("segundos: " + segundos); } } </pre>
<p style="text-align: center;">Program.cs</p>	<p style="text-align: center;">Main.java</p>
<pre> using System; namespace Guia1_LAB121 { public class Program { public static void Main() { Reloj r1 = new Reloj(20,10,0); Reloj r2 = new Reloj(14,30,0); Console.WriteLine("Reloj 1"); r1.mostrar(); Console.WriteLine("\nReloj 2"); r2.mostrar(); Console.Write("\n Adelantar al reloj1 0h, 2m, 3s\n"); int xh = 0, xm = 2, xs = 3; r1.adelantar(xh,xm,xs); r1.mostrar(); Console.Write("\n Sincronizar Reloj1 con reloj2\n"); r1.sincronizar(r2); r1.mostrar(); Console.ReadKey(true); } } } </pre>	<pre> class Main { public static void main(String[] args) { Reloj r1 = new Reloj(20, 10, 0); Reloj r2 = new Reloj(14, 30, 0); System.out.println("Reloj 1"); r1.mostrar(); System.out.println("\nReloj 2"); r2.mostrar(); System.out.print("\n Adelantar al reloj1 0h, 2m, 3s\n"); int xh = 0, xm = 2, xs = 3; r1.adelantar(xh, xm, xs); r1.mostrar(); System.out.print("\n Sincronizar Reloj1 con reloj2\n"); r1.sincronizar(r2); r1.mostrar(); } } </pre>
<p style="text-align: center;">LA CAPTURA DE PANTALLA DE EJECUCION EN JAVA</p>	

```

Console
<terminated> Main [Java Application] C:\Program Files\Java\jre7\bin\javaw.exe (09/09/2016 09:26:27)

Reloj 1
Mostrando datos del reloj...
horas: 20
minutos : 10
segundos: 0

Reloj 2
Mostrando datos del reloj...
horas: 14
minutos : 30
segundos: 0

Adelantar al reloj1 0h, 2m, 3s

Mostrando datos del reloj...
horas: 20
minutos : 12
segundos: 3

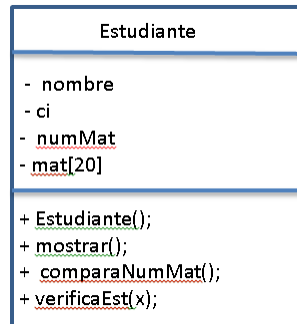
Sincronizar el reloj1 con el reloj2

Mostrando datos del reloj...
horas: 14
minutos : 30
segundos: 0

```

2. Ejemplo.

Implementar la Clase estudiante y resolver:



- Instanciar tres objetos de tipo estudiante de diferente manera para luego mostrar sus datos.
- Comparar si es que el estudiante 1 y el estudiante 2 llevan la misma cantidad de materias.
- Dada una materia x buscar a los estudiantes que la estén cursando, si es así mostrar sus datos.

SOLUCION.-

Estudiante.cs	Estudiante.java
<pre> using System; namespace Guia1_LAB121 { public class Estudiante { private string nombre; private int ci; private int numMat; private string [] mat = new String [20]; //A) CONSTRUCTOR POR DEFECTO public Estudiante() { this.nombre = "Roberto Flores"; this.ci = 64795; this.numMat = 2; this.mat[1] = "Inf-121"; this.mat[2] = "Lab-111"; } //A) CONSTRUCTORES CON ARGUMENTOS public Estudiante(string nom) </pre>	<pre> public class Estudiante { private String nombre; private int ci; private int numMat; private String mat [] = new String [20]; //A) CONSTRUCTOR POR DEFECTO public Estudiante() { this.nombre = "Juan Perez"; this.ci = 15937; this.numMat = 3; this.mat[1] = "Inf-111"; this.mat[2] = "Lab-121"; this.mat[3] = "Mat-125"; } //A) CONSTRUCTORES CON ARGUMENTOS public Estudiante(String nom){ this.nombre = nom; this.ci = 36478; this.numMat = 3; this.mat[1] = "Inf-121"; this.mat[2] = "Lab-121"; this.mat[3] = "Mat-115"; } } </pre>

<pre> { this.nombre = nom; this.ci = 16845; this.numMat = 3; this.mat[1] = "Inf-121"; this.mat[2] = "Lab-111"; this.mat[3] = "Mat-114"; } public Estudiante(string nom, int ci) { this.nombre = nom; this.ci = ci; this.numMat = 4; this.mat[1] = "Mat-115"; this.mat[2] = "Lab-121"; this.mat[3] = "Mat-124"; this.mat[4] = "Inf-113"; } //A) METODO MOSTRAR public void mostrar() { Console.WriteLine("\n DATOS DEL ESTUDIANTE"); Console.WriteLine("NOMBRE: "+this.nombre); Console.WriteLine("CI: "+this.ci); Console.WriteLine("NRO MATERIAS INS: "+this.numMat); for(int i=1 ; i <= this.numMat ; i++){ Console.WriteLine(i+"a "+this.mat[i]); } } //B) COMPARAR SI ES QUE EL ESTUDIANTE 1 Y EL ESTUDIANTE 2 LLEVAN LA MISMA CANTIDAD DE MATERIAS. public void comparaNumMat(Estudiante e2) { if(this.numMat == e2.NumMat) Console.WriteLine("EL ESTUDIANTE "+this.nombre+" LLEVA EL MISMO NUMERO DE MATERIAS QUE " + e2.Nombre); else Console.WriteLine("AMBOS ESTUDIANTES NO LLEVAN LA MISMA CANTIDAD DE MATERIAS"); } } //C) DADA UNA MATERIA X BUSCAR A LOS QUE LA ESTÉN //CURSANDO, SI ES ASÍ MOSTRAR EL/LOS SUS DATOS. public bool verificaEst(string x) { bool sw = false; for(int i=1 ; i<= this.numMat ; i++) { if(this.mat[i] == x){ Console.WriteLine("DATOS DEL ESTUDIANTE QUE CURSA LA MATERIA"+x); Console.WriteLine("NOMBRE "+this.nombre); Console.WriteLine("CI => "+this.ci); sw = true; break; } } return sw; } //GETTERS public string Nombre { get { return nombre; } } public int Ci { get { return ci; } } public int NumMat { get { return numMat; } } } } </pre>	<pre> } public Estudiante(String nom, int ci){ this.nombre = nom; this.ci = ci; this.numMat = 4; this.mat[1] = "Mat-115"; this.mat[2] = "Lab-121"; this.mat[3] = "Mat-124"; this.mat[4] = "Inf-113"; } //A) METODO MOSTRAR public void mostrar(){ System.out.println("\n*** DATOS DEL ESTUDIANTE ***"); System.out.println("NOMBRE: "+this.nombre); System.out.println("CI: "+this.ci); System.out.println("NUMERO DE MATERIAS INSCRITAS: "+this.numMat); for(int i=1 ; i <= this.numMat ; i++){ System.out.println(i+"a "+this.mat[i]); } } //B) COMPARAR SI ES QUE EL ESTUDIANTE 1 Y EL ESTUDIANTE // 2 LLEVAN LA MISMA CANTIDAD DE MATERIAS. public void comparaNumMat(Estudiante e2) { if(this.numMat == e2.getNumMat()){ System.out.println("EL ESTUDIANTE "+this.nombre+" LLEVA EL MISMO NUMERO DE MATERIAS QUE "+ e2.getNombre()); } } } //C) DADA UNA MATERIA X BUSCAR A LOS QUE LA ESTÉN CURSANDO, SI ES ASÍ MOSTRAR EL/LOS SUS DATOS. public boolean verificaEst(String x) { boolean sw = false; for(int i=1 ; i<= this.numMat ; i++){ if(this.mat[i] == x){ System.out.println("DATOS DEL ESTUDIANTE QUE CURSA LA MATERIA "+x); System.out.println("NOMBRE => "+this.nombre); System.out.println("CI => "+this.ci); sw = true; break; } } return sw; } } //GETTERS Y SETERRS public String getNombre() { return nombre; } public void setNombre(String nombre) { this.nombre = nombre; } public int getCi() { return ci; } public void setCi(int ci) { this.ci = ci; } public int getNumMat() { return numMat; } public void setNumMat(int numMat) { this.numMat = numMat; } public String[] getMat() { return mat; } public void setMat(String[] mat) { this.mat = mat; } } } </pre>
Program.cs	Main.java

```

using System;

namespace Guia1_LAB121
{
    public class Program
    {
        public static void Main() {
            //INSTANCIANDO 3 OBJETOS DE DIFERENTE MANERA,
            //MOSTRARLOS
            Estudiante e1 = new Estudiante();
            Estudiante e2 = new Estudiante("Julian Zevallos");
            Estudiante e3 = new Estudiante("Vanesa
            Linares",47861);
            Console.WriteLine("\n+++ INCISO A) +++");
            e1.mostrar();
            e2.mostrar();
            e3.mostrar();

            //verificar el numero de materias
            Console.WriteLine("\n+++ INCISO B)+++");
            e1.comparaNumMat(e2);

            //verifica la materia "x"
            Console.WriteLine("\n+++ INCISO C)+++");
            string x = "Inf-121";
            bool v1 = e1.verificaEst(x);
            bool v2 = e2.verificaEst(x);
            bool v3 = e3.verificaEst(x);
            if(!v1 && !v2 && !v3)
                Console.WriteLine("LA MATERIA " + x + " NO ES
            CURSADA POR NINGUN ESTUDIANTE");

            Console.ReadKey();
        }
    }
}

```

```

public class Principal {

    public static void main(String[] args) {
        //INSTANCIANDO 3 OBJETOS DE DIFERENTE MANERA MOSTRARLOS
        Estudiante e1 = new Estudiante();
        Estudiante e2 = new Estudiante("Analia Rios");
        Estudiante e3 = new Estudiante("Jorge Alanoca",94685);

        System.out.println("+++ INCISO A) +++");
        e1.mostrar();
        e2.mostrar();
        e3.mostrar();

        //verificar el numero de materias
        System.out.println("\n+++ INCISO B) +++");
        e1.comparaNumMat(e2);

        //verifica la materia "x"
        System.out.println("\n+++ INCISO C) +++");
        String x = "Inf-121";
        boolean v1 = e1.verifica(x);
        boolean v2 = e2.verifica(x);
        boolean v3 = e3.verifica(x);
        if(!v1 && !v2 && !v3){
            System.out.println("LA MATERIA "+x+" NO ES CURSADA
            POR NINGUN ESTUDIANTE");
        }
    }
}

```

LA CAPTURA DE PANTALLA DE EJECUCION EN JAVA

```

<terminated> Principal (6) [Java Application] C:\Program Files\Java\jre7\bin\javaw.exe (10/09/2016 22:38:08)
+++++++ INCISO A) ++++++++

*** DATOS DEL ESTUDIANTE ***
NOMBRE: Juan Perez
CI: 15937
NUMERO DE MATERIAS INSCRITAS: 3
1# Inf-111
2# Lab-121
3# Mat-125

*** DATOS DEL ESTUDIANTE ***
NOMBRE: Analia Rios
CI: 36478
NUMERO DE MATERIAS INSCRITAS: 3
1# Inf-121
2# Lab-121
3# Mat-115

*** DATOS DEL ESTUDIANTE ***
NOMBRE: Jorge Alanoca
CI: 94685
NUMERO DE MATERIAS INSCRITAS: 4
1# Mat-115
2# Lab-121
3# Mat-124
4# Inf-113

+++++++ INCISO B) ++++++++

EL ESTUDIANTE Juan Perez LLEVA EL MISMO NUMERO DE MATERIAS QUE Analia Rios

+++++++ INCISO C) ++++++++

DATOS DEL ESTUDIANTE QUE CURSA LA MATERIA Inf-121
NOMBRE => Analia Rios
CI => 36478

```

PARTE II PROBLEMA A RESOLVER

En el siguiente problema se muestra la interacción de objetos de diferente clase.

Implementar un P.O.O. para la clase Empresa y Empleado

Empleado	Empresa
- nom : string - anioIng : int - cargo : string	- nom : string - ne : int - emp[,] : string
+ Empleado() + Empleado(nom,anioIng) + Mostrar(): void + getters(), setters(a)	+ Empresa() + Empleado(nom,ne) + Mostrar(): void + BuscarEmpleado(string): bool + EdadEmpleado(string): int + getters(), setters(a)

- Instanciar 2 objetos Empresa y 3 Objetos empleado de diferente manera, mostrar datos
- Dado una empresa buscar al empleado de nombre x
- Dado un obj empresa mostrar la edad de empleado de nombre x
- Dado un obj empresa y un obj empleado mostrar la edad de empleado, si este trabaja en la empresa

SOLUCION.-

Clases implementadas en C#

Empleado.cs

```
using System;

namespace Guia1_LAB121
{
    // User: Marcelo Aruquipa
    public class Empleado{
        private string nom;
        private int anioIng;
        private string cargo;

        public Empleado (){
            this.nom = "Adrian";
            this.anioIng = 2000;
            this.cargo = "Gerente";
        }
        public Empleado(string nom){
            this.nom = nom;
            this.anioIng = 2005;
            this.cargo = "operador";
        }
        public Empleado(string nom, int anioIng){
            this.nom = nom;
            this.anioIng = anioIng;
            this.cargo = "asistente";
        }
        public Empleado(string nom,int anioIng,string cargo){
            this.nom=nom;
            this.anioIng=anioIng;
            this.cargo=cargo;
        }
        public void Mostrar(){
            Console.WriteLine("\n\n NomEmp= {0} \n nroEmp= {1} \n Cargo= {2} ",this.nom, this.anioIng, this.cargo);
        }
        public string Nom {
            get { return nom; }
            set { nom = value; }
        }
        public int AnioIng {
            get { return anioIng; }
            set { anioIng = value; }
        }
        public string Cargo {
            get { return cargo; }
            set { cargo = value; }
        }
    }
}
```

```

    }
}

```

Empresa.cs

```

using System;

namespace Guia1_LAB121
{
    public class Empresa
    {
        private string nom;
        private int ne;
        private string [,] emp = new string[50,4];

        public Empresa()
        {
            this.nom = "Entel";
            this.ne = 2;
            this.emp[0,0] = "Alan";    this.emp[0,1] = "1999";    this.emp[0,2] = "Gerente";
            this.emp[1,0] = "Rosa";    this.emp[1,1] = "2005";    this.emp[1,2] = "Secretaria";
        }

        public Empresa(string nom)
        {
            this.nom = nom;
            this.ne = 1;
            this.emp[0,0] = "Luigui"; this.emp[0,1] = "2008"; this.emp[0,2] = "Mensajero";
        }

        public Empresa(string nom, int ne){
            this.nom = nom;
            this.ne = ne;
            Console.WriteLine("\n Intr.  nomEmp, Año Nac, Cargo \n");
            for(int i=0; i<this.ne ; i++){
                this.emp[i,0] = Console.ReadLine();
                this.emp[i,1] = Console.ReadLine();
                this.emp[i,2] = Console.ReadLine();
            }
        }

        public void Mostrar(){
            Console.WriteLine("\n\n NomEmp : {0} \n nroEmp: {1} ",this.nom, this.ne);
            Console.WriteLine("\n Empleados: ");
            Console.WriteLine("\n Nombre    Año Nac    Cargo");

            for(int i=0; i<this.ne; i++){
                Console.WriteLine("\n {0}  {1}  {2} ", this.emp[i,0],this.emp[i,1], this.emp[i,2]);
            }
        }

        public bool BuscarEmpleado(string x){
            for(int i=0; i<this.ne; i++){
                if(this.emp[i,0] == x){
                    return true;
                }
            }
            return false;
        }

        public int EdadEmpleado(string nom){
            for(int i=0; i<this.ne; i++){
                if(this.emp[i,0] == nom)
                    return (DateTime.Now.Year - int.Parse(this.emp[i,1]));
            }
            return 0;
        }

        public string Nom {
            get { return nom; }
            set { nom = value; }
        }

        public int Ne {
            get { return ne; }
            set { ne = value; }
        }

        public string[,] Emp {

```



```

        get { return emp; }
        set { emp = value; }
    }
}

```

Principal.cs

```

namespace Guia1_LAB121
{
    public class Principal
    {
        public static void Main()
        {
            Empresa e1 = new Empresa();
            Empresa e2 = new Empresa("VIVA");

            Empleado t1 = new Empleado("Juan");
            Empleado t2 = new Empleado("Alan");
            Empleado t3 = new Empleado("Marco");

            Console.WriteLine("\n Datos de empresa e1");
            e1.Mostrar();
            Console.WriteLine("\n Datos de empresa e2");
            e2.Mostrar();

            Console.WriteLine("\n\n Busqueda de empleado");

            Console.WriteLine("\n\n" + e1.BuscarEmpleado("Pedro"));
            Console.WriteLine("\n\n" + e2.BuscarEmpleado("Luigui"));

            Console.WriteLine("\n\n Uso de propiedad get y set");
            Console.WriteLine("\n\n Nombre empresa e1: " + e1.Nom);

            //cambiamos de nombre
            e1.Nom = "YPFB";
            e1.Mostrar();

            Console.WriteLine("\n\n Dado un obj empresa mostrar la edad de empleado de nombre x");
            string nomEmpleado = "Alan";
            if(e1.BuscarEmpleado(nomEmpleado))
                Console.WriteLine("\n Edad = " + e1.EdadEmpleado(nomEmpleado));
            else
                Console.WriteLine("\n el Empleado no existe");
            Console.WriteLine("\n\n Dado un obj empresa y un obj empleado mostrar la edad de empleado");
            nomEmpleado = t3.Nom;
            if(e1.BuscarEmpleado(nomEmpleado))
                Console.WriteLine("\n Edad = " + e1.edadEmpleado(nomEmpleado));
            else
                Console.WriteLine("\n el Empleado no existe en la empresa");

            Console.ReadKey();
        }
    }
}

```

LA CAPTURA DE PANTALLA DE EJECUCION EN C#

```

C:\Users\Marcelo\Documents\SharpDevelop Projects\Guia1_LAB121\Guia1_LAB121\bin...

Datos de empresa e1
NomEmp : Entel
nroEmp: 2
Empleados:
Nombre  Año Nac  Cargo
Alan   1999  Gerente
Rosa   2005  Secretaria
Datos de empresa e2
NomEmp : UIUA
nroEmp: 1
Empleados:
Nombre  Año Nac  Cargo
Luigui 2008  Mensajero

Busqueda de empleado
False
True

Uso de propiedad get y set

Nombre empresa e1: Entel

NomEmp : YPFB
nroEmp: 2
Empleados:
Nombre  Año Nac  Cargo
Alan   1999  Gerente
Rosa   2005  Secretaria

Dado un obj empresa mostrar la edad de empleado de nombre x
Edad = 17

Dado un obj empresa y un obj empleado mostrar la edad de empleado
el Empleado no existe en la empresa

```

Clases implementadas en JAVA

Empleado.java

```

public class Empleado{
    private String nom;
    private int anioIng;
    private String cargo;

    public Empleado (){
        this.nom = "Adrian";
        this.anioIng = 2000;
        this.cargo = "Gerente";
    }
    public Empleado(String nom){
        this.nom = nom;
        this.anioIng = 2005;
        this.cargo = "operador";
    }
    public Empleado(String nom, int anioIng){
        this.nom = nom;
        this.anioIng = anioIng;
        this.cargo = "asistente";
    }
    public Empleado(String nom,int anioIng,String cargo){
        this.nom=nom;
        this.anioIng=anioIng;
        this.cargo=cargo;
    }
    public void mostrar(){
        System.out.printf("\n\n NomEmp= %s  \n nroEmp= %d \n Cargo= %s ",this.nom, this.anioIng, this.cargo);
    }

    public String getNom() {
        return nom;
    }
    public void setNom(String nom) {
        this.nom = nom;
    }
    public int getAnioIng() {
        return anioIng;
    }
}

```

```

    }
    public void setAnioIng(int anioIng) {
        this.anioIng = anioIng;
    }
    public String getCargo() {
        return cargo;
    }
    public void setCargo(String cargo) {
        this.cargo = cargo;
    }
}

```

Empresa.java

```

import java.util.Scanner;

public class Empresa
{
    private String nom;
    private int ne;
    private String [][]emp = new String[50][4];

    public Empresa()
    {
        this.nom = "Entel";
        this.ne = 2;
        this.emp[0][0] = "Alan"; this.emp[0][1] = "1999"; this.emp[0][2] = "Gerente";
        this.emp[1][0] = "Rosa"; this.emp[1][1] = "2005"; this.emp[1][2] = "Secretaria";
    }
    public Empresa(String nom)
    {
        this.nom = nom;
        this.ne = 1;
        this.emp[0][0] = "Luigui"; this.emp[0][1] = "2008"; this.emp[0][2] = "Mensajero";
    }
    public Empresa(String nom, int ne){
        Scanner lee = new Scanner(System.in);
        this.nom = nom;
        this.ne = ne;
        System.out.print("\n Intr.  nomEmp, Año Nac, Cargo \n");
        for(int i=0; i<this.ne ; i++){
            this.emp[i][0] = lee.next();
            this.emp[i][1] = lee.next();
            this.emp[i][2] = lee.next();
        }
    }
    public void mostrar(){
        System.out.printf("\n NomEmp : %s \n nroEmp: %s ",this.nom, this.ne);
        System.out.print("\n Empleados: ");
        System.out.print("\n Nombre    Año Nac    Cargo");
        for(int i=0; i<this.ne; i++)
            System.out.printf("\n %s    %s    %s", this.emp[i][0],this.emp[i][1], this.emp[i][2]);
    }
    public boolean buscarEmpleado(String x){
        for(int i=0; i<this.ne; i++){
            if(this.emp[i][0] == x)
                return true;
        }
        return false;
    }
    public int edadEmpleado(String nom){
        Calendar fecha = Calendar.getInstance();
        for(int i=0; i<this.ne; i++){
            if(this.emp[i][0] == nom)
                return (fecha.get(Calendar.YEAR) - Integer.parseInt(this.emp[i][1]));
        }
        return 0;
    }
    public String getNom() {
        return nom;
    }
}

```

```

    public void setNom(String nom) {
        this.nom = nom;
    }
    public int getNe() {
        return ne;
    }
    public void setNe(int ne) {
        this.ne = ne;
    }
    public String[][] getEmp() {
        return emp;
    }
    public void setEmp(String[][] emp) {
        this.emp = emp;
    }
}

```

Main.java

```

public class Main {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Empresa e1 = new Empresa();
        Empresa e2 = new Empresa("VIVA");

        Empleado t1 = new Empleado("Juan");
        Empleado t2 = new Empleado("Alan");
        Empleado t3 = new Empleado("Marco");

        System.out.print("\n Datos de empresa e1");
        e1.mostrar();
        System.out.print("\n\n Datos de empresa e2");
        e2.mostrar();

        System.out.print("\n\n Búsqueda de empleado");
        System.out.print("\n\n" + e1.buscarEmpleado("Pedro"));
        System.out.print("\n\n" + e2.buscarEmpleado("Luigui"));

        System.out.print("\n\n Uso de propiedad get y set");
        System.out.print("\n\n Nombre empresa e1: " + e1.getNom());

        //cambiamos de nombre
        e1.setNom("YPFB");
        e1.mostrar();

        System.out.print("\n\n Dado un obj empresa mostrar la edad de empleado de nombre x");
        String nomEmpleado = "Alan";
        if(e1.buscarEmpleado(nomEmpleado))
            System.out.print("\n Edad = " + e1.edadEmpleado(nomEmpleado));
        else
            System.out.print("\n el Empleado no existe");

        System.out.print("\n\n Dado un obj empresa y un obj empleado mostrar la edad de empleado");
        nomEmpleado = t3.getNom();
        if(e1.buscarEmpleado(nomEmpleado))
            System.out.print("\n Edad = " + e1.edadEmpleado(nomEmpleado));
        else
            System.out.print("\n el Empleado no existe en la empresa");
    }
}

```

LA CAPTURA DE PANTALLA DE EJECUCION EN JAVA

```

@ Javadoc Declaration Search Console History Tasks
<terminated> Main (24) [Java Application] C:\Program Files\Java\jre7\bin\javaw.exe (11/09/2016 20:08:17)

Datos de empresa e1
NomEmp : Entel
nroEmp: 2
Empleados:
Nombre    Año Nac    Cargo
Alan  1999  Gerente
Rosa   2005  Secretaria

Datos de empresa e2
NomEmp : VIVA
nroEmp: 1
Empleados:
Nombre    Año Nac    Cargo
Luigui  2008  Mensajero

Busqueda de empleado

false

true
Uso de propiedad get y set

Nombre empresa e1:  Entel
NomEmp : YPFB
nroEmp: 2
Empleados:
Nombre    Año Nac    Cargo
Alan  1999  Gerente
Rosa   2005  Secretaria

Dado un obj empresa mostrar la edad de empleado de nombre x
Edad = 17

Dado un obj empresa y un obj empleado mostrar la edad de empleado
el Empleado no existe en la empresa

```

EJERCICIOS.

- Ordenar los empleados de la empresa por orden alfabético.
- Ordenar los empleados por edad.
- Dado tres objetos empleado mostrarlos de manera ordenada por antigüedad.

OBSERVACIONES

- ✓ Se debe hacer notar que cuando no se define constructores, al instanciar objetos, estos son objetos vacíos y nulos.
- ✓ En estos dos ejemplos las clases tienen implementados sus constructores por defecto como parametrizados, por lo que no fue necesario hacer la lectura de datos por teclado.
- ✓ Se sugiere hacer de todas maneras la implementación de los métodos leer.
- ✓ Se utiliza la propiedad **DateTime** en C# obtención del año actual.
- ✓ Se utiliza la clase **Calendar** en JAVA para obtener el año actual.
- ✓ Se hizo el uso de los estilos de Programación de JAVA y C#.