Programación 3

Orientación a objetos con **C#**

2. Tipos, arrays, strings

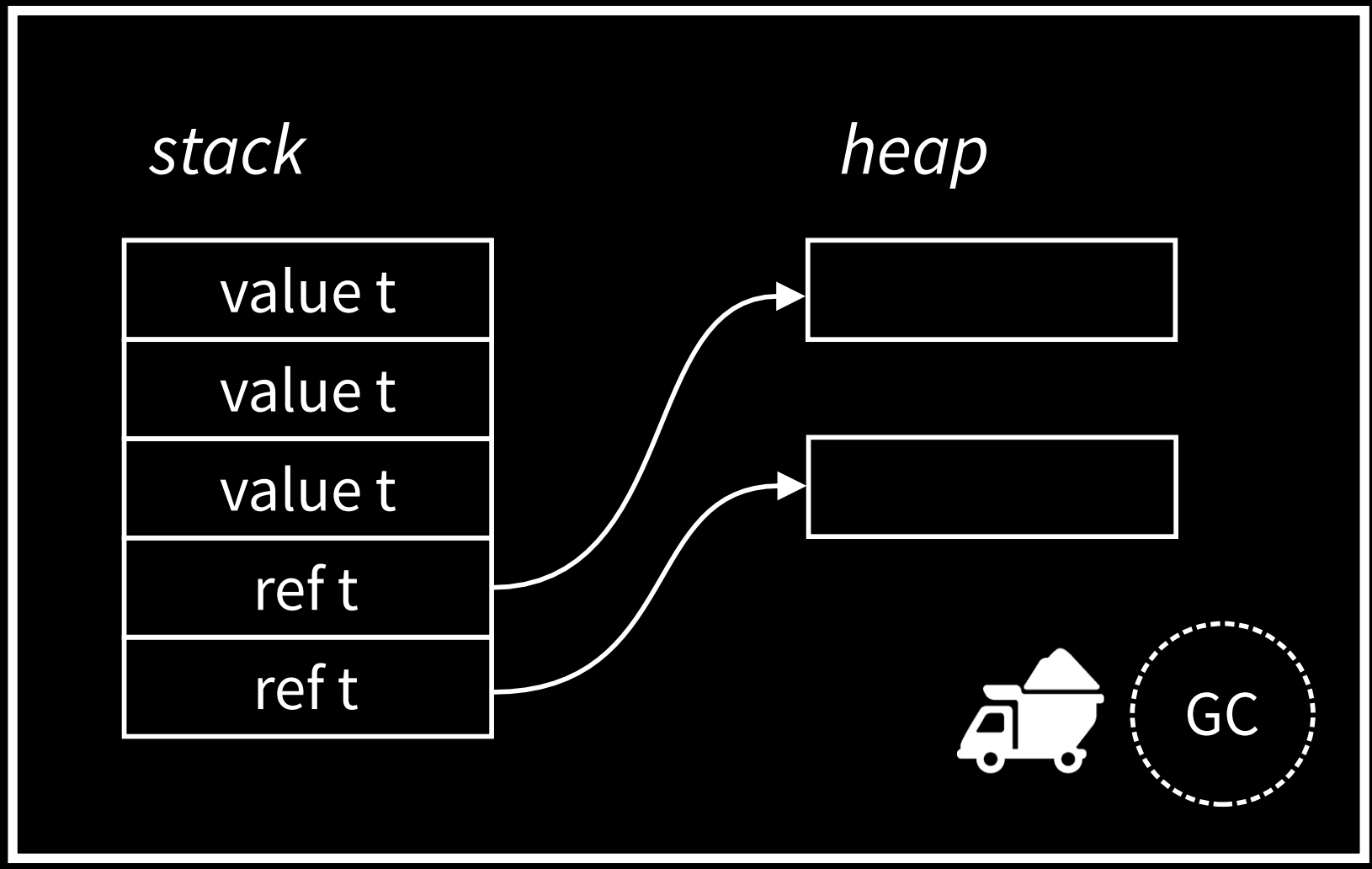
Mg. Mauro Gullino
maurogullino@gmail.com

bool	System.Boolean
byte	System.Byte
sbyte	System.SByte
char	System.Char
decimal	System.Decimal
double	System.Double
float	System.Single
short	System.Int16
ushort	System.UInt16
int	System.Int32
uint	System.UInt32
long	System.Int64
ulong	System.UInt64
string	System.String
object	System.Object

value types

reference types

CLR “managed data”



```
int hola;           //error var no asignada
Console.WriteLine(hola.GetType());
Console.ReadKey();


int hola = 6;
Console.WriteLine(hola.GetType());


int hola = new System.Int32();
Console.WriteLine(hola);
```

```
double hola = 6;  //conversión implícita
Console.WriteLine(hola.GetType());


int hola = 6d;   //error casting
Console.WriteLine(hola.GetType());
```

```
float hola = 3.5;          //error casting
Console.WriteLine(hola.GetType());


float hola = 3.5f;
Console.WriteLine(hola.GetType());


double x = 3 / 2;
Console.WriteLine(x.GetType());
Console.WriteLine(x);
```

```
int dato, dato2;

checked
{
      dato = Int32.MaxValue;
      dato2 = dato + 1;
}

Console.WriteLine(dato2);
```

```
bool prueba = true;
int prueba2 = 1;

if(prueba) Console.WriteLine(“prueba”);
if(prueba2) Console.WriteLine(“prueba2”); //error

if(prueba2==1) ....  //ok
```

Nullable Types

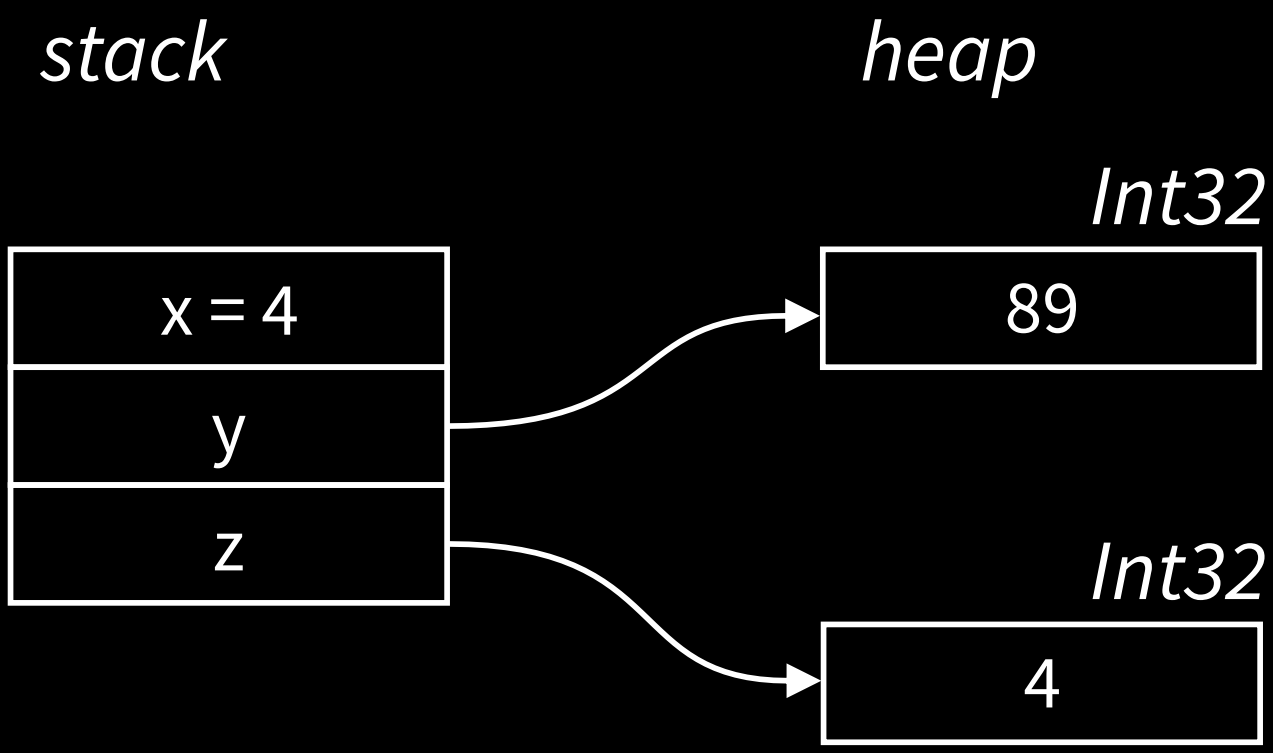
```
int? x = null;

if (x.HasValue) Console.WriteLine(x);
else Console.WriteLine("es nulo");

Console.WriteLine(x.GetValueOrDefault());


//solo se puede hacer con value types
```

Boxing & Unboxing

```
int x = 4;
object y = x;
```

boxing

```
object z = y;
Console.WriteLine(y.GetType());
y = 89;

Console.WriteLine(y);
Console.WriteLine(z);
```


stack
heap
Int32
x = 4
89
y
z
Int32
4

```
int x = 4, z;
object y = x;
y++;  //error

z = (int) y;
z++;
Console.WriteLine(z);
```

¿Por qué es importante?

```
float e = 3.1459f;
float d = e;

object o1 = d;
object o2 = e;

Console.WriteLine(d == e);   //value types
Console.WriteLine(o1 == o2); //ref types

Console.WriteLine(o1.Equals(o2));
```

Arrays

- son objetos *(reference type)*
- se necesita una variable para la referencia
- tamaño fijo y mismo tipo

```
int[] enteros, enteros2;
enteros = new int[10];
enteros2 = new int[5] { 0, 2, 4, 6, 8};

Console.WriteLine(enteros.GetType());

Console.WriteLine(enteros.Length);
```

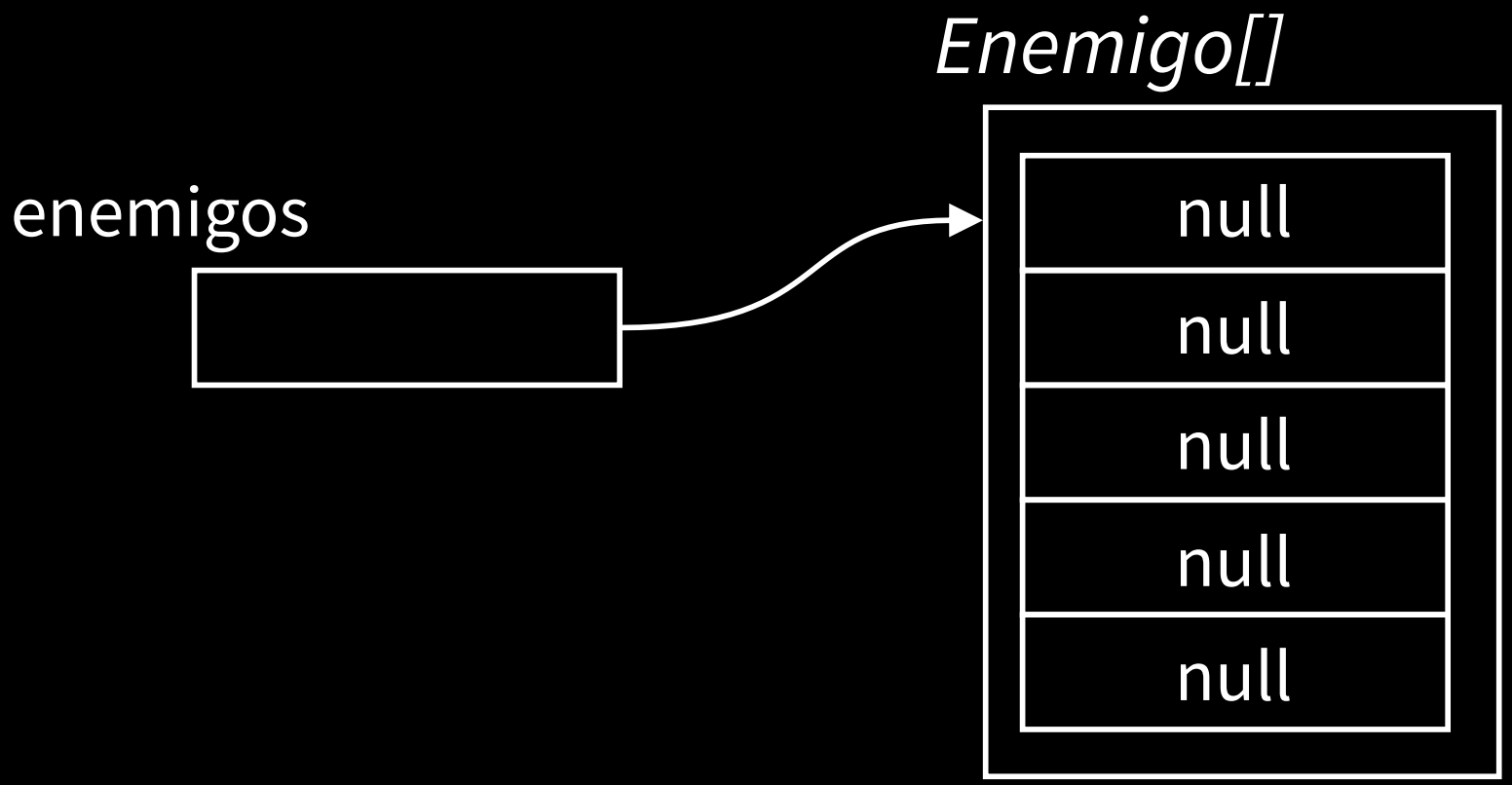
```
int[] enteros, enteros2;
enteros = new int[5] { 0, 2, 4, 6, 8 };
enteros2 = enteros;

Console.WriteLine(enteros2 == enteros);

enteros2[0] = 10;
Console.WriteLine(enteros[0]);

Console.WriteLine(enteros2 == enteros);
```

- chequeo de contornos en ejecución

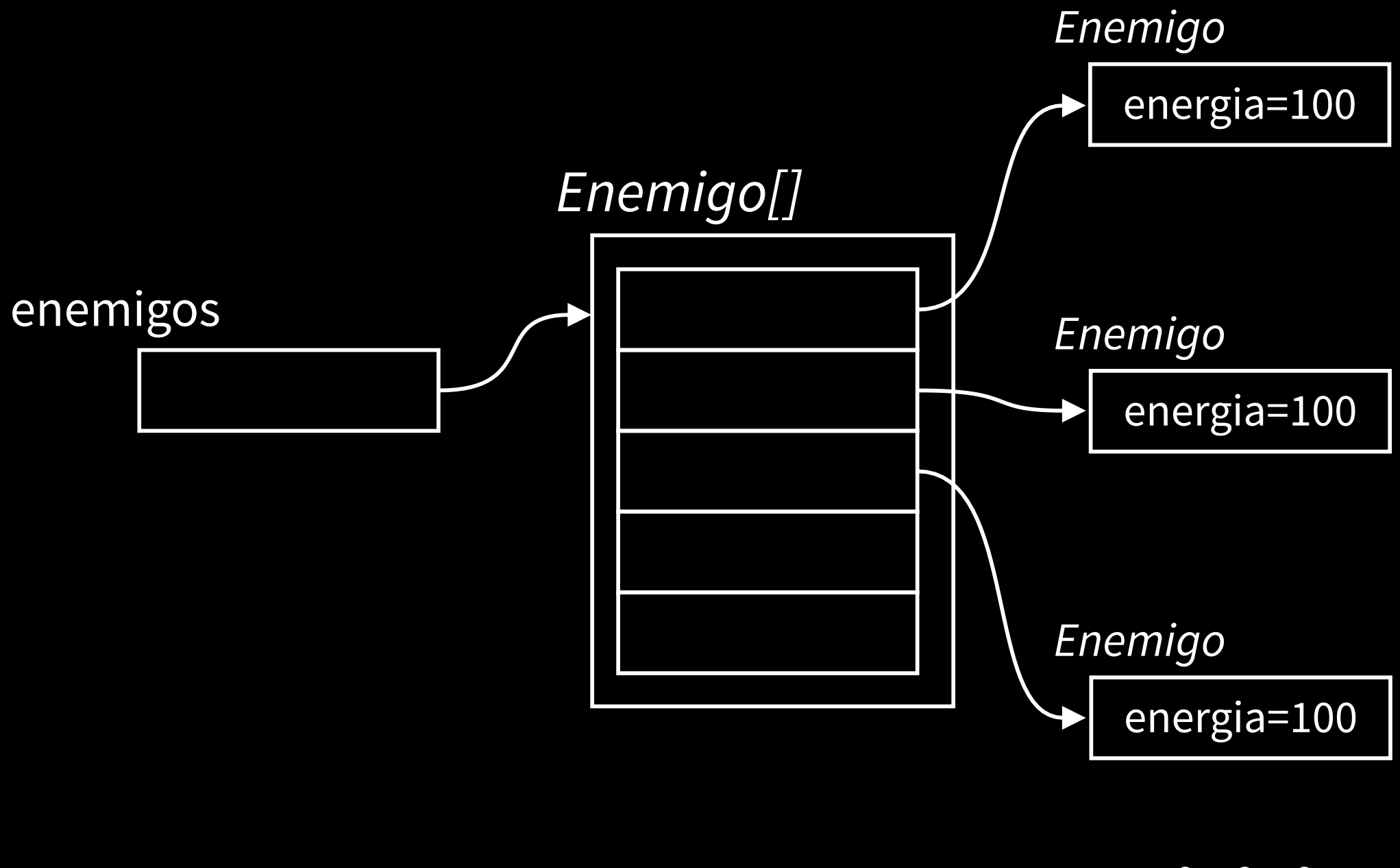
```
int[] numeros = new int[5];

numeros[10] = 343;  //out of range
```

```
Enemigo[] enemigos;
enemigos = new Enemigo[5];

Console.WriteLine(enemigos.GetType());



class Enemigo {
   public int energia = 100;
}
```

```
Enemigo[] enemigos = new Enemigo[5];

for(int i = 0; i < enemigos.Length; i++) {
    Console.WriteLine(enemigos[i].energia);
}   //qué se debería ver?


class Enemigo {
    public int energia = 100;
}
```


Enemigo[]
enemigos
null
null
null
null
null

```
Enemigo[] enemigos = new Enemigo[5];

for(int i = 0; i < enemigos.Length; i++) {
    enemigos[i] = new Enemigo();
    Console.WriteLine(enemigos[i].energia);
}


class Enemigo {
    public int energia = 100;
}
```


Enemigo
energia=100
Enemigo[]
enemigos
Enemigo
energia=100
Enemigo
energia=100
• • •

```
Enemigo[] enemigos = { new Enemigo(),
                       new Enemigo(), new Enemigo() };


foreach(Enemigo unEnemigo in enemigos) {
    unEnemigo.energia -= 50;
    Console.WriteLine(unEnemigo.energia);
}


class Enemigo {
    public int energia = 100;
}
```

Strings

- son objetos *(reference type)*
- se comportan como valores inmutables

```
string uno = "hola";
string dos = uno;

Console.WriteLine(uno.GetType());
Console.WriteLine(object.ReferenceEquals(uno, dos));

dos = "mundo";
string tres = "hola";
Console.WriteLine(object.ReferenceEquals(uno, dos));
Console.WriteLine(object.ReferenceEquals(tres, uno));
```

String

“hola”

String

“mundo”

Ojo!

```
string uno = null;
string dos = "";

Console.WriteLine(uno.Length); //error en ejecución
Console.WriteLine(dos.Length);

Console.WriteLine(String.IsNullOrEmpty(uno)); //ok
```

Ya que estamos... operador is

```
string uno = null;

if (uno is int)  Console.WriteLine("imposible");

if (uno is null)  Console.WriteLine("ojo");
```

`is` evalua “compatibilidad de tipos”
en tiempo de ejecución

```
Console.WriteLine(3 is int);   //warning

Enemigo e = new Enemigo();
Console.WriteLine(e is int);   //warning
Console.WriteLine(e is Enemigo);
Console.WriteLine(e is object);
```

```
3 is object   ?
```

strings son conjuntos de char

```
char[] letras = { 'ñ', 'a', 'n', 'd', 'ú' };
string animal = new String(letras);
Console.WriteLine(animal);

string hola = "你好";
Console.WriteLine(hola);
Console.WriteLine(hola.Length);
```

un char es un caracter Unicode

```
char copy = '\u00A9';
char omega = '\u03A9';

Console.WriteLine(sizeof(char));
Console.WriteLine(copy);
Console.WriteLine(omega);
```

@ = verbatim string

```
string a = "hola \t mundo";
string b = @"hola \t mundo";

string c = "Soy \"programador\" dijo";
string d = @"Soy ""programador"" dijo";

string e = "\\\\server\\datos\\sueldos.xlsx";
string f = @"\\server\datos\sueldos.xlsx";

string g = "uno\r\ndos\r\ntres";
string h = @"uno
             dos
             tres";
```

$ = interpolación

```
int dias = 7;
string texto = $"la semana tiene {dias} dias";
Console.WriteLine(texto);
```

```
float duda = .1f;
double prueba = duda;
Console.WriteLine($"flotante {duda * 0.1}");

double duda2 = .1;
Console.WriteLine($"doble {duda2 * 0.1}");
```

+ = Concatenación

```
string fecha = "Hoy es ";

fecha = fecha + DateTime.Now.ToString("D");
fecha += ".";

Console.WriteLine(fecha);
```

ojo

Ojo!!

```
string s1 = "hola ";
string s2 = s1;
s1 += "mundo";

Console.WriteLine(s2);
Console.WriteLine(s1);
```

Comparación de strings

```
string uno = "hola";
string dos = "que tal";

// if(uno < dos) ...   //no compila

Console.WriteLine(String.Compare(uno, dos));
Console.WriteLine(String.Compare(dos, uno));
```

Lectura de Consola

Ejercicios

```
int edad;
double sueldo;
string ingreso;

ingreso = Console.ReadLine();
edad = int.Parse(ingreso);

ingreso = Console.ReadLine();
sueldo = double.Parse(ingreso);

Console.WriteLine("Al jubilarse habrá
     cobrado {0}", (65 - edad) * 12 * sueldo);
```

Ejercicios

1) Se ingresan 5 double.
Calcular promedio, varianza y desviación estándar.

$$\frac{1}{n}\sum_{i=1}^{n}\left(X_i - \overline{X}\right)^2$$

{1, 2, 3, 4, 5} σ = 1.414

Ejercicios

2) El usuario ingresará frases hasta que ingrese “FIN”.

Mostrar en pantalla la frase que tiene más vocales.
Considerar mayúsculas, minúsculas y acentos.