

Desarrollo de Sistemas Web

Tec. Superior en Análisis Funcional de Sistemas

Práctica Prof. I

```
1 <!DOCTYPE HTML>
2 <html>
3   <head>
4     <title> HTML
5     <meta http-ec
6     <meta name="c
7     <meta name="
8     <link rel="s
9     <link rel="
10    <script type
11  </head>
12  <body>
13    <h1> HTML
14    <div classe
15    <div cla
```

1

Funciones def.

2

Tipado definido

3

Variables locales

4

Variables estáticas

5

Variables globales

6

Variables
superglobales



```
1 declare(strict_types=1);
2 function bienvenida(string $usuario = 'Invitado'):string{
3     return "Bienvenido $usuario";
4 }
5 echo bienvenida('Nicolás');
6
7
8 function sumar(int ... $numeros):int{
9     return array_sum($numeros);
10 }
11 echo sumar(1,2,3,4,5);
12
13
14 function sumar2($numero1, $numero2):int{
15     return $numero1+$numero2;
16 }
17 echo sumar2(1,2);
18
19 echo date('d-m-Y');
```

1

FUNCIONES

Conjunto de instrucciones que, a lo largo de la ejecución del programa, se van a poder invocar todas las veces que las necesitamos.

Palabra reservada

Nombre función

Parámetros

1

FUNCIONES

```
1 function bienvenida($usuario = 'Invitado')
2 {
3     echo "Bienvenido $usuario";
4 }
5 bienvenida('Juan');
```

Código de la
función

Invocación

1

FUNCIONES

Las que ejecutan acciones

Las que retornan valores



```
1 function bienvenida($usuario = 'Invitado')
2 {
3     echo "Bienvenido $usuario";
4 }
5 bienvenida('Juan');
```



```
1 function bienvenida($usuario){
2     return "Bienvenido $usuario";
3 }
4 echo bienvenida('Nicolás');
```

1

FUNCIONES



```
1 function bienvenida($usuario = 'Invitado')
2 {
3     echo "Bienvenido $usuario";
4 }
5 bienvenida('Juan');
```



```
1 function sumar( $numero1, $numero2){
2     return $numero1+$numero2;
3 }
4 echo sumar(1,2);
```


RESTRICCIONES DE INGRESO

RESTRICCIÓN
DE SALIDA

2

FUNCIONES
CON TIPADO
DEFINIDO



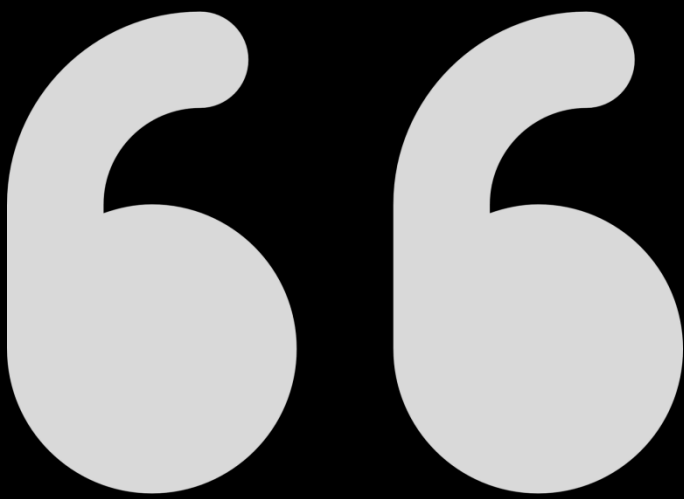
```
1 function sumar(int $numero1,int $numero2):int
2 {
3     return $numero1+$numero2;
4 }
5 echo sumar(1,2);
```

2

FUNCIONES CON TIPADO DEFINIDO



```
1 declare(strict_types=1);  
2 function sumar(int ... $numeros):int{  
3     return array_sum($numeros);  
4 }  
5 echo sumar(1,2,3,4,5);
```

El ámbito de una variable es el contexto dentro del que la variable está definida.



ÁMBITO DE VARIABLE



VARIABLES
LOCALES



VARIABLES
GLOBALES



VARIABLES
ESTÁTICAS



VARIABLES
SUPERGLOBALES

3

VARIABLES LOCALES



```
1 $varLocal = "Hola<br>";  
2 function miFuncion()  
3 {  
4     $varLocal = "Hola desde función <br>";  
5 }  
6 miFuncion();  
7 echo $varLocal;
```



```
1  function test()  
2  {  
3      static $a = 0;  
4      echo $a++;  
5  }  
6  test();  
7  test();  
8  test();
```

4

VARIABLES ESTÁTICAS

Valor: 0

Valor: 1

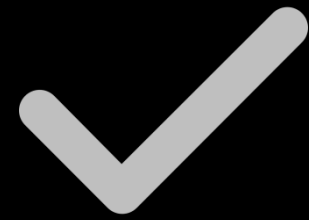
Valor: 2



```
1 $varGlobal = "Hola <br>";  
2 echo $varGlobal;  
3 function miFuncion2()  
4 {  
5     global $varGlobal;  
6     $varGlobal = "Hola desde función <br>";  
7 }  
8 miFuncion2();  
9 echo $varGlobal;
```



```
1 $varGlobal = "Hola <br>";  
2 echo $varGlobal;  
3 function miFuncion2()  
4 {  
5     $GLOBALS['varGlobal'] = "Hola desde función <br>";  
6 }  
7 miFuncion2();  
8 echo $varGlobal;
```



FORMULARIOS

⇒ \$_POST

⇒ \$_GET

⇒ \$_FILE

⇒ \$_REQUEST

5

VARIABLES
SUPERGLOBALES



SERVER

⇒ \$_SERVER



SESIÓN Y COOKIES

⇒ \$_SESSION

⇒ \$_COOKIE



ENTORNO

⇒ \$_ENV

POO

```
1 <?php
2 class Patitos {
3     function nueva ( $a ) {
4         print "Patitos: nada" <br>";
5     }
6 }
7 //Instanciamos objetos
8 $hugo= new Patitos();
```

1

PARADIGMA DE
PROGRAMACIÓN

2

UTILIZA OBJETOS

3

UTILIZA
INTERACCIONES

4

FORMA DE
PROGRAMAR,
CERCANA A LA
VIDA REAL

5

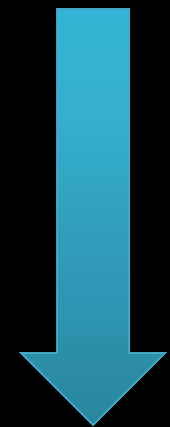
RESOLUCIÓN SE
REALIZA CON
OBJETOS

“

Metodología que basa la estructura de los programas en torno a los objetos.

1

POO



Cosa o entidad, que tiene atributos (propiedades) y formas de operar sobre ellos (métodos).

”

Palabra reservada

Nombre



```
1  class Usuario {  
2      #atributos  
3      #métodos  
4  }
```

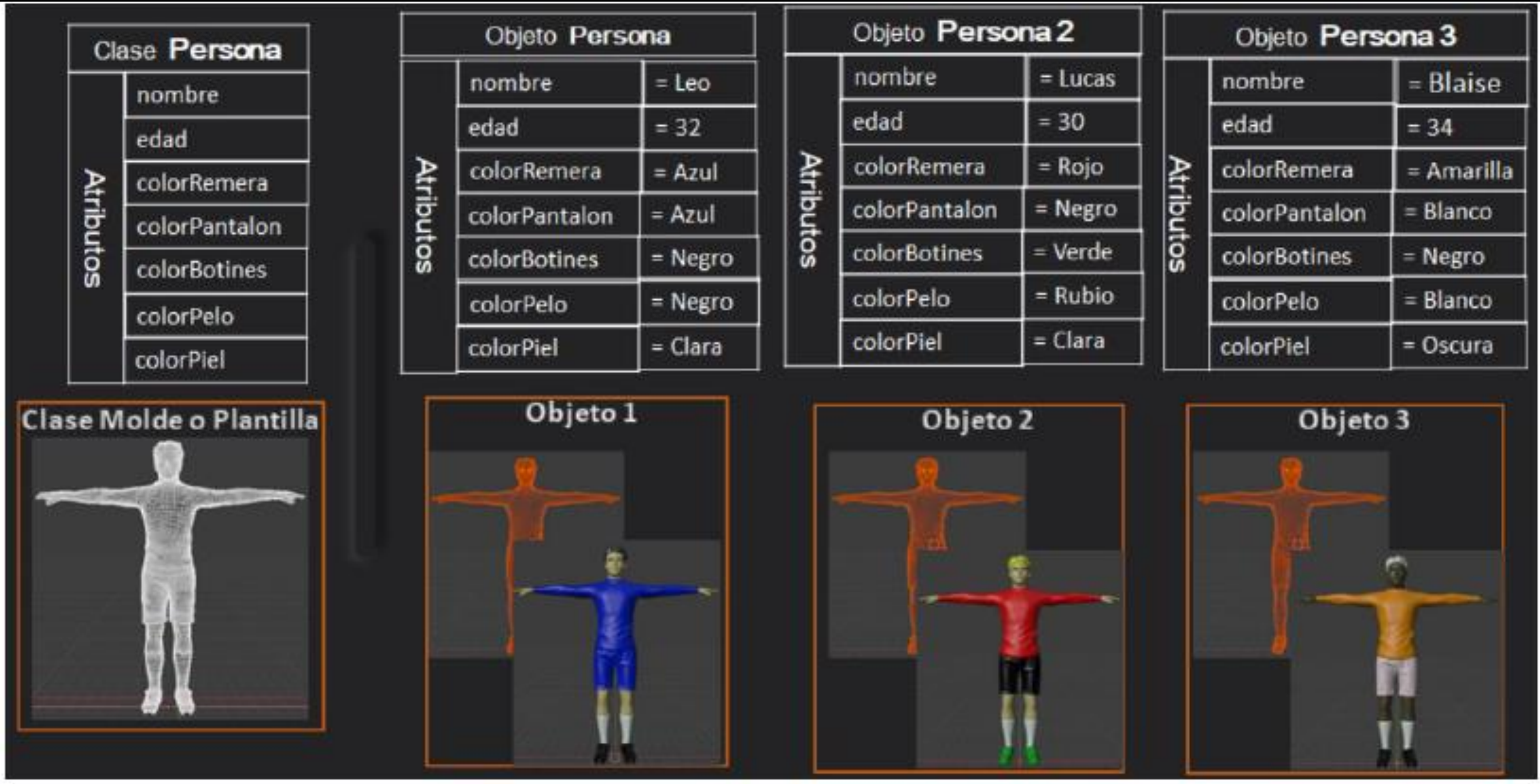
2

Sintaxis

Nombre Clase	Identidad
Atributo 1 Atributo 2 Atributo n	Estados
Método 1 Método 2 Método n	Comportamientos

3

UML



```
1 class Conexion
2 {
3     public $con;
4     public function conectar()
5     {
6         $this->con = mysqli_connect("localhost", "root", "", "poo");
7     }
8 }
```

```
1 include_once 'Conexion.php';
2 class Cliente extends Conexion
3 {
4     public $nombre, $apellido, $fecnac, $email;
5 }
```

PUBLIC	PROTECTED	PRIVATE
ACCEDO AL ATRIBUTO POR FUERA DE LA CLASE	EL ATRIBUTO ES PROPIO DE LA CLASE	EL ATRIBUTO ES PROPIO DE LA CLASE
LAS CLASES HIJAS LAS HEREDAN	LAS CLASES HIJAS LAS HEREDAN	LAS CLASES HIJAS NO LAS HEREDAN
NO SE ENCAPSULAN	ENCAPSULAMIENTO	ENCAPSULAMIENTO

4

Atributos



```
1  include_once 'Cliente.php';  
2  
3  $cliente = new Cliente;  
4  $cliente->nombre = "Nicolásss";  
5  $cliente->apellido = "Rotili";  
6  $cliente->fecnac = "1993-06-24";  
7  $cliente->email = "rot@rot.com";  
8  $cliente->create();
```

4

Instancia


```
1 class Cliente extends Conexion
2 {
3
4     public $nombre, $apellido, $fecnac, $email;
5
6     //Create
7     public function create()
8     {
9         $this->conectar();
10        $pre = mysqli_prepare($this->con, "INSERT INTO clientes (nombre, apellido, fecnac, email) VALUES (?, ?, ?, ?)");
11        $pre->bind_param("ssss", $this->nombre, $this->apellido, $this->fecnac, $this->email);
12        $pre->execute();
13    }
14 }
```

Método estático

```
1 class Cliente extends Conexion
2 {
3
4     public $nombre, $apellido, $fecnac, $email;
5
6     //Read
7     public static function all()
8     {
9         $conexion = new Conexion();
10        $conexion->conectar();
11        $pre = mysqli_prepare($conexion->con, "SELECT * FROM clientes");
12        $pre->execute();
13        $res = $pre->get_result();
14
15        $clientes = array();
16        while ($cliente = $res->fetch_object(Cliente::class)) {
17            array_push($clientes, $cliente);
18        }
19
20        return $clientes;
21    }
22
23 }
```

```
1 //READ
2 $clientes = Cliente::all();
```

Desarrollo de Sistemas Web

Tec. Superior en Análisis Funcional de Sistemas