

# Table Of Content

---

<a href="#">instituto</a> .....	2
<a href="#">Alumno</a> .....	2
<a href="#">Instituto</a> .....	3
<a href="#">Pessoa</a> .....	4
<a href="#">Profesor</a> .....	6
<a href="#">Index</a> .....	9

# Package instituto

## Class Summary

[Alumno](#)

[Instituto](#)

[Pessoa](#)

[Profesor](#)

---

instituto

## Class Alumno

```
java.lang.Object
|
+-- Pessoa
    |
    +-- instituto.Alumno
```

---

< [Constructors](#) > < [Methods](#) >

---

```
public class Alumno
extends Pessoa
```

**Author:**

a18pablogn

**Version:**

Beta

1.0 Classe que pertence o pacote Instituto, e ademáis é unha clase filla de Pessoa.

## Constructors

# Alumno

```
public Alumno(java.lang.String nome,  
              int idade)
```

Método Constructor, necesario para instanciar obxetos desta clase.

## Parameters:

nome - atributo que procede da clase Pai e que fai falla pasarllo como argumento o constructor da clase Alumno.

idade - atributo que procede da clase Pai e que fai falla pasarllo como argumento o constructor da clase Alumno. numExpedienteActual e atributo estático (sempre ten o mesmo valor común en todos os fillos), utilízase para contabilizar os obxetos creados da clase Alumno numExpediente e un atributo individual de cada obxecto, no momento da creación desa instancia (obxecto) almacena o valor.

## Methods

### getNumExpediente

```
public int getNumExpediente()
```

Método Getter

## Returns:

devolve o número de expediente deste Obxecto, valor único de cada instancia

### toString

```
public java.lang.String toString()
```

Método toString()

## Returns:

devolve un String personalizado para visualizar cada instancia da clase Alumno. Acepta o uso do Polimorfismo.

## Overrides:

[toString](#) in class [Persoa](#)

instituto

## Class Instituto

```
java.lang.Object  
|  
+--instituto.Instituto
```

< [Constructors](#) > < [Methods](#) >

---

public class **Instituto**  
extends java.lang.Object

**Author:**  
a18pablogn

**Version:**  
Beta  
15/10/2019

## Constructors

### Instituto

```
public Instituto()
```

## Methods

### main

```
public static void main(java.lang.String[] args)
```

---

instituto

## Class Pessoa

```
java.lang.Object  
|  
+--instituto.Pessoa
```

**Direct Known Subclasses:**  
[Alumno](#), [Profesor](#)

---

< [Constructors](#) > < [Methods](#) >

---

public class **Pessoa**  
extends java.lang.Object

**Author:**  
a18pablogn

**Version:**  
Beta

15/10/2019 Clase principal onde se establecen os atributos básicos de nome e idade Declara os seus atributos como protected, para que soamente sexan accesibles dende esta clase, ou dende as clases fillas

## Constructors

### Persoa

```
public Persoa(java.lang.String nome,  
              int idade)
```

Método Constructor, necesario para instanciar obxetos desta clase.

**Parameters:**

nome - atributo tipo string para gardar o nome, necesario pasarlle este dato como parámetro do constructor.

idade - atributo tipo enteiro para gardar o nome, necesario pasarlle este dato como parámetro do constructor.

## Methods

### getIdade

```
public int getIdade()
```

Descripción: Método Getter do atributo idade.

**Returns:**

devolve (un enteiro) o valor do atributo idade.

---

### getNome

```
public java.lang.String getNome()
```

Descripción: Método Getter do atributo nome.

**Returns:**

devolve (un string) co valor do atributo nome

---

### setIdade

```
public void setIdade(int idade)
```

Descripción: Método Setter (modificador) do atributo idade.

**Parameters:**

idade - recibe como parámetro nome (un enteiro) para asignarlle este valor o atributo idade

---

## setName

```
public void setName(java.lang.String nome)
```

Descripción: Método Setter (modificador) do atributo nome.

**Parameters:**

nome - recibe como parámetro nome (un string) para asignarlle este valor o atributo nome.

---

## toString

```
public java.lang.String toString()
```

Descripción: Método toString Devuelve un string personalizado que se utiliza para mostrar la clase. Acepta o uso do Polimorfismo.

**Overrides:**

toString in class java.lang.Object

---

instituto

# Class Profesor

```
java.lang.Object
|
+-- Persoa
    |
    +-- instituto.Profesor
```

---

< [Constructors](#) > < [Methods](#) >

---

```
public class Profesor
extends Persoa
```

**Author:**

a18pablogn

**Version:**

Beta

15/10/2019 Clase que pertence o paquete Instituto, e ademáis é unha clase filla de Persoa.

## Constructors

# Profesor

```
public Profesor(java.lang.String nome,  
                int idade,  
                int salario)
```

Método Constructor da clase Profesor, necesario para instanciar obxetos desta clase.

## Parameters:

nome - atributo tipo string que se utiliza para chamar o constructor super (da clase Persoa), e que fai falla pasarllo como argumento o constructor da clase Profesor. Almacena o valor deste obxeto.

idade - atributo tipo enteiro que se utiliza para chamar o constructor super (da clase Persoa), e que fai falla pasarllo como argumento o constructor da clase Profesor. Almacena o valor deste obxeto.

salario - atributo tipo enteiro que almacena o valor deste atributo, ademais e necesario pasarllo como parámetro o constructo deste clase.

## Methods

### getSalario

```
public int getSalario()
```

Método Getter do atributo Salario

## Returns:

devolve un enteiro co valor do atributo Salario do obxeto que o executa

---

### setSalario

```
public void setSalario(int salario)
```

Descripción: Método Setter (modificador) do atributo salario.

## Parameters:

salario - recibe como parámetro salario (un enteiro) para asignarlle este valor o atributo Salario.

---

# toString

```
public java.lang.String toString()
```

Método toString()

**Returns:**

devolve un String personalizado para visualizar cada instancia da clase Profesor. Acepta o uso do Polimorfismo.

**Overrides:**

[toString](#) in class [Persoa](#)



# INDEX

## A

[Alumno](#) ... 2

[Alumno](#) ... 3

## G

[getIdade](#) ... 5

[getNome](#) ... 5

[getNumExpediente](#) ... 3

[getSalario](#) ... 7

## I

[Instituto](#) ... 3

[Instituto](#) ... 4

## M

[main](#) ... 4

## P

[Pessoa](#) ... 4

[Pessoa](#) ... 5

[Profesor](#) ... 6

[Profesor](#) ... 7

## S

[setIdade](#) ... 5

[setNome](#) ... 6

[setSalario](#) ... 7

## T

[toString](#) ... 3

[toString](#) ... 6

[toString](#) ... 8