Reporte de la práctica 5 - I.C.C

Javier Alejandro Mancera Quiroz
Grupo: 7030

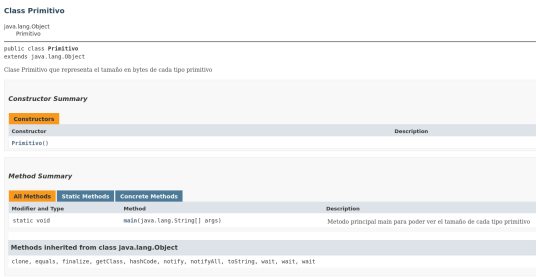
9 de noviembre de 2021

Javadoc es una utilidad de Oracle para la generación de documentación de APIs en formato HTML a partir de código fuente Java. Javadoc es el estándar de la industria para documentar clases de Java. La mayoría de los IDEs los generan automáticamente.
Para generar API con Javadoc han de usarse etiquetas de HTML o ciertas palabras reservadas precedidas por el carácter "@".
Estas etiquetas se escriben al principio de cada clase, miembro o método, dependiendo de qué objeto se desee describir, mediante un comentario iniciado con '/**' y acabado con '*/'.

a) @author - Nombre del desarrollador
b) @version - Versión de la clase o método
c) @param - Definición de un parámetro de un método
d) @throws - Excepción lanzada por el método.
e) @see - Asocia con otro método o clase.
f) @returns - Informa de lo que devuelve el método, no se puede usar en constructores o métodos "void".
g) @since - Especifica la versión del producto.
h) @serial - Describe el significado del campo y sus valores aceptables.
i) @deprecated - Indica que el método o clase es antigua y que no se recomienda su uso porque posiblemente desaparecerá en versiones posteriores.

Práctica 0

1. Primitivo.java



Class Primitivo

java.lang.Object
  Primitivo

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

Clase Primitivo que representa el tamaño en bytes de cada tipo primitivo

Constructor Summary

Constructors

Constructor	Description
Primitivo()	

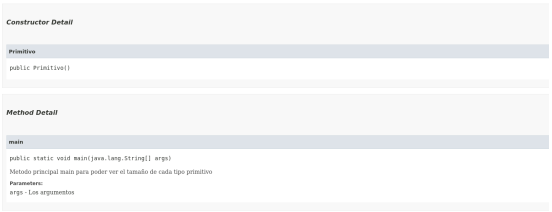
Method Summary

All Methods **Static Methods** **Concrete Methods**

Modifier and Type	Method	Description
static void	main(java.lang.String[] args)	Metodo principal main para poder ver el tamaño de cada tipo primitivo

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait



Constructor Detail

Primitivo

public Primitivo()

Method Detail

main

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

Metodo principal main para poder ver el tamaño de cada tipo primitivo

Parameters:
args - Los argumentos

2. Cilindro.java

Class Cilindro

java.lang.Object
  Cilindro

```
public class Cilindro
extends java.lang.Object
```

Clase Cilindro que representa el area y el volumen de un cilindro

Constructor Summary

Constructors

Constructor	Description
Cilindro()	

Method Summary

All Methods | Static Methods | Concrete Methods

Modifier and Type	Method	Description
static void	main(java.lang.String[] args)	Metodo principal main para sacar el area y el volumen de un cilindro

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Detail

Cilindro

```
public Cilindro()
```

Method Detail

main

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

Metodo principal main para sacar el area y el volumen de un cilindro

Parameters:
args - Los argumentos

Práctica 1

1. Condicionales.java

Class Condicionales

java.lang.Object
  Condicionales

```
public class Condicionales
extends java.lang.Object
```

Clase Condicionales que representa el número con mayor valor

Constructor Summary

Constructors

Constructor	Description
Condicionales()	

Method Summary

All Methods | Static Methods | Concrete Methods

Modifier and Type	Method	Description
static void	main(java.lang.String[] args)	Metodo principal main para poder ejecutar la condicional if para saber el número con mayor valor

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Detail

Condicionales

```
public Condicionales()
```

Method Detail

main

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

Metodo principal main para poder ejecutar la condicional if para saber el número con mayor valor

Parameters:
args - Los argumentos

2. Bisiesto.java

Class Bisiesto

java.lang.Object
  Bisiesto

```
public class Bisiesto
extends java.lang.Object
```

Clase Bisiesto que representa si el año es bisiesto o no

Constructor Summary

Constructors

Constructor	Description
Bisiesto()	

Method Summary

All Methods | Static Methods | Concrete Methods

Modifier and Type	Method	Description
static void	main(java.lang.String[] args)	Metodo principal main para poder ejecutar la condicional if - else para saber si un año es bisiesto o no

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Detail

Bisiesto

```
public Bisiesto()
```

Method Detail

main

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

Metodo principal main para poder ejecutar la condicional if - else para saber si un año es bisiesto o no

Parameters:
args - Los argumentos

3. CicloFor.java

Class CicloFor

java.lang.Object
  CicloFor

```
public class CicloFor
extends java.lang.Object
```

Clase CicloFor que representa los números primos con un cierto límite

Constructor Summary

Constructors

Constructor	Description
CicloFor()	

Method Summary

All Methods | Static Methods | Concrete Methods

Modifier and Type	Method	Description
static void	main(java.lang.String[] args)	Metodo principal main para poder ejecutar el ciclo for para calcular los numeros primos

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Detail

CicloFor

```
public CicloFor()
```

Method Detail

main

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

Metodo principal main para poder ejecutar el ciclo for para calcular los numeros primos

Parameters:
args - Los argumentos

Práctica 2

1. CicloWhile.java

Class CicloWhile

java.lang.Object
CicloWhile

```
public class CicloWhile
extends java.lang.Object
```

Clase CicloWhile que representa los números primos con un cierto límite

Constructor Summary

Constructors

Constructor	Description
CicloWhile()	

Method Summary

All Methods | Static Methods | Concrete Methods

Modifier and Type	Method	Description
static void	main(java.lang.String[] args)	Metodo principal main para poder ejecutar el ciclo while para calcular los numeros primos

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Detail

CicloWhile

```
public CicloWhile()
```

Method Detail

main

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

Metodo principal main para poder ejecutar el ciclo while para calcular los numeros primos

Parameters:
args - Los argumentos

2. CicloDoWhile.java

Class CicloDoWhile

java.lang.Object
CicloDoWhile

```
public class CicloDoWhile
extends java.lang.Object
```

Clase CicloDoWhile que representa los números primos con un cierto límite

Constructor Summary

Constructors

Constructor	Description
CicloDoWhile()	

Method Summary

All Methods | Static Methods | Concrete Methods

Modifier and Type	Method	Description
static void	main(java.lang.String[] args)	Metodo principal main para poder ejecutar el ciclo do - while para calcular los numeros primos

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Detail

CicloDoWhile

```
public CicloDoWhile()
```

Method Detail

main

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

Metodo principal main para poder ejecutar el ciclo do - while para calcular los numeros primos

Parameters:
args - Los argumentos

Práctica 4

1. Cuenta.java

Class Cuenta

java.lang.Object
Cuenta

```
public class Cuenta
extends java.lang.Object
```

Clase Cuenta que representa una cuenta bancaria del titular y su salario

Constructor Summary

Constructors

Constructor	Description
Cuenta()	Constructor sin parametros que representa la cuenta ya con valores iniciales
Cuenta(java.lang.String titular)	Constructor que recibe un titular de una cuenta con dinero disponible de 0

Method Summary

All Methods | Instance Methods | Concrete Methods

Modifier and Type	Method	Description
double	getDinero()	Metodo que devuelve el dinero disponible
java.lang.String	getTitular()	Metodo que devuelve el titular
void	setDinero(double dineroDisponible)	Metodo que cambia el valor del dinero disponible
void	setTitular(java.lang.String titular)	Metodo que cambia el nombre del titular
java.lang.String	toString()	Metodo toString para poder representar en forma de texto a 'titular' y 'dineroDisponible' y poder visualizarlo

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait

Constructor Detail

Cuenta

```
public Cuenta()
```

Constructor sin parametros que representa la cuenta ya con valores iniciales

Cuenta

```
public Cuenta(java.lang.String titular)
```

Constructor que recibe un titular de una cuenta con dinero disponible de 0

Parameters:
titular - Titular de la cuenta

Method Detail

getTitular

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

Metodo que devuelve el titular

Returns:
El titular de la cuenta

setTitular

```
public void setTitular(java.lang.String titular)
```

Metodo que cambia el nombre del titular

Parameters:
titular - El titular

getDinero

```
public double getDinero()
```

Metodo que devuelve el dinero disponible

Returns:
El dinero disponible

setDinero

```
public void setDinero(double dineroDisponible)
```

Metodo que cambia el valor del dinero disponible

Parameters:
dineroDisponible - El dinero disponible

toString

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

Metodo toString para poder representar en forma de texto a 'titular' y 'dineroDisponible' y poder visualizarlo

Overrides:
toString in class java.lang.Object

2. Main.java

Class Main

java.lang.Object
  Main

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

Clase Main que representa una cuenta bancaria del titular y su salario

Constructor Summary

Constructors

Constructor	Description
`Main()`	

Method Summary

All Methods | **Static Methods** | **Concrete Methods**

Modifier and Type	Method	Description
`static void`	`main(java.lang.String[] args)`	Metodo principal main para poder ejecutar la clase Main y ocupar lo metodos de la clase Cuenta

Methods inherited from class java.lang.Object

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructor Detail

Main

`public Main()`

Method Detail

main

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

Metodo principal main para poder ejecutar la clase Main y ocupar lo metodos de la clase Cuenta

Parameters:
`args` - Los argumentos

3. Rectangulo.java

Class Rectangulo

java.lang.Object
  Rectangulo

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

Clase Rectangulo que representa las caracteristicas de un rectangulo

Constructor Summary

Constructors

Constructor	Description
`Rectangulo()`	Constructor sin parametros que representa la base y la altura del rectangulo ya con valores iniciales
`Rectangulo(double base, double altura)`	Constructor que recibe los valores de la base y la altura de un rectangulo

Method Summary

All Methods | **Instance Methods** | **Concrete Methods**

Modifier and Type	Method	Description
`double`	`getAltura()`	Metodo que devuelve la altura
`double`	`getArea()`	Metodo que devuelve la base y la altura, sacando el area del rectangulo
`double`	`getBase()`	Metodo que devuelve la base
`double`	`getPerimetro()`	Metodo que devuelve la base y la altura, sacando el perimetro del rectangulo
`void`	`setAltura(double altura)`	Metodo que cambia el valor de la altura
`void`	`setBase(double base)`	Metodo que cambia el valor de la base
`java.lang.String`	`toString()`	Metodo toString para poder representar en forma de texto a 'base', 'altura' y los resultados del area y del perimetro de un rectangulo, y poder visualizarlo

Methods inherited from class java.lang.Object

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, wait, wait, wait`

Constructor Detail

Rectangulo

`public Rectangulo()`

Constructor sin parametros que representa la base y la altura del rectangulo ya con valores iniciales

Rectangulo

`public Rectangulo(double base,`
`  double altura)`

Constructor que recibe los valores de la base y la altura de un rectangulo

Parameters:
`base` - Base del rectangulo
`altura` - Altura del rectangulo

Method Detail

getBase

`public double getBase()`

Metodo que devuelve la base

Returns:
Base del rectangulo

setBase

`public void setBase(double base)`

Metodo que cambia el valor de la base

Parameters:
`base` - Base del rectangulo

getAltura

`public double getAltura()`

Metodo que devuelve la altura

Returns:
Altura del rectangulo

setAltura

`public void setAltura(double altura)`

Metodo que cambia el valor de la altura

Parameters:
`altura` - Altura del rectangulo

getArea

`public double getArea()`

Metodo que devuelve la base y la altura, sacando el area del rectangulo

Returns:
Area del rectangulo

getPerimetro

`public double getPerimetro()`

Metodo que devuelve la base y la altura, sacando el perimetro del rectangulo

Returns:
Perimetro del rectangulo

toString

`public java.lang.String toString()`

Metodo toString para poder representar en forma de texto a 'base', 'altura' y los resultados del area y del perimetro de un rectangulo, y poder visualizarlo

Overrides:
`toString` in class `java.lang.Object`

2. MainRec.java

Class MainRec

java.lang.Object
  MainRec

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

Clase MainRec que representa las caracteristicas de un rectangulo

Constructor Summary

Constructors

Constructor	Description
`MainRec()`	

Method Summary

All Methods | **Static Methods** | **Concrete Methods**

Modifier and Type	Method	Description
`static void`	`main(java.lang.String[] args)`	Metodo principal main para poder ejecutar la clase MainRec y ocupar lo metodos de la clase Rectangulo

Methods inherited from class java.lang.Object

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructor Detail

MainRec

`public MainRec()`

Method Detail

main

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

Metodo principal main para poder ejecutar la clase MainRec y ocupar lo metodos de la clase Rectangulo

Parameters:
`args` - Los argumentos