

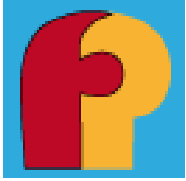
Fundamentos de Programación

Presentación de la asignatura

2º Cuatrimestre

Curso 2024/2025

Grupo TI3



2024/25

Fundamentos de Programación

Horario y aulas

- **Clases de teoría (TI3):**

Día: Viernes; Hora: 15:30-17:40

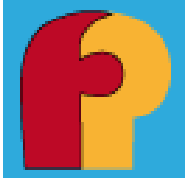
Prof.: Alfonso Bengoa Díaz Aula: H0.11

- **Clases de laboratorio:**

Día: Lunes; Hora: 17:40 -19:30

Profs.:

- Alfonso Bengoa Díaz (Subgr. 1: **TI31**) (F1.30)
- Manuel Carranza García (Subgr. 2: **TI32**) (F1.31)



2024/25

Fundamentos de Programación

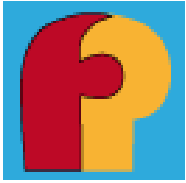
Profesor teoría: Tutorías

Horario Tutorías

- lunes de 19:30 a 21:00
- viernes de 17:30 a 20:30

Observaciones:

1. Previa petición por email al profesor (indicado el día preferido), con objeto de citar al alumnado a una hora concreta dentro de dichos días y horas establecidos. No obstante, también podría ser en otro horario y día de la semana que se pudiese acordar con el profesor.
2. Otra opción, siempre que la consulta o cuestiones planteadas lo permitan, se pueden atender por email (bengoa@us.es) o a través de videoconferencia.



2024/25

Fundamentos de Programación

Calendario de trabajo segundo cuatrimestre

LABORATORIOS	TEORÍA
LUNES	VIERNES

FEBRERO

TEMAS DE JAVA

1. Introducción al lenguaje Java
2. Diseño de tipos
3. Colecciones y Map
4. Tratamientos secuenciales

27	28	29	30	31	1	2		T1
3	4	5	6	7	8	9	T2	T3
10	11	12	13	14	15	16	T4	T5
17	18	19	20	21	22	23	L1	T6

MARZO

24	25	26	27	28	1	2	L2	
3	4	5	6	7	8	9	L3	T7
10	11	12	13	14	15	16	L4	EXAMEN 3
17	18	19	20	21	22	23	L5	T8
24	25	26	27	28	29	30	L6	T9

Temas: 1, 2 y 3

ABRIL

31	1	2	3	4	5	6	L7	T10
7	8	9	10	11	12	13	L8	T11
14	15	16	17	18	19	20	SEMANA	SANTA
21	22	23	24	25	26	27	L9	T12

MAYO

28	29	30	1	2	3	4	L10	T13
5	6	7	8	9	10	11	SEMANA	FERIA
12	13	14	15	16	17	18	L11	T14
19	20	21	22	23	24	25	EXAMEN 4	T15
26	27	28	29	30	31			

Temas: 4

SEGUNDO EXAMEN PRÁCTICO

JUNIO

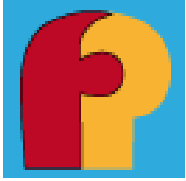
						1	
2	3	4	5	6	7	8	
9	10	11	12	13	14	15	
16	17	18	19	20	21	22	
23	24	25	26	27	28	29	

PRIMERA CONVOCATORIA: 11 DE JUNIO DE 2025

JULIO

30	1	2	3	4	5	6	
7	8	9	10	11	12	13	
14	15	16	17	18	19	20	
21	22	23	24	25	26	27	
28	29	30					

SEGUNDA CONVOCATORIA: 9 DE JULIO DE 2025

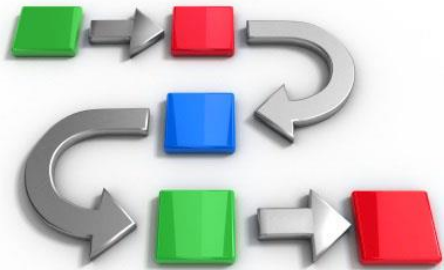


2024/25

Fundamentos de Programación

Objetivos de la Asignatura

➤ *Pensamiento computacional*



Metodología



Abstracción

ARRAY LIST
JAVA - CREATE, ADD
DATA STRUCTURE

Estructuras de datos

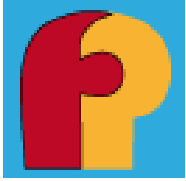
➤ *Implementación de programas*



Lenguaje PYTHON



Lenguaje JAVA



2024/25

Fundamentos de Programación

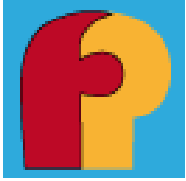
Implementación de programas

Primer cuatrimestre

- Programación imperativa en Python

Segundo cuatrimestre

- Programación orientada a objetos en Java



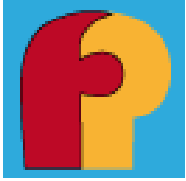
2024/25

Fundamentos de Programación

Metodología de trabajo

- Clases de teoría y, sobre todo, ejercicios
- Sesiones de laboratorio
- Evaluaciones teóricas y prácticas (presenciales)
- **Programar, programar, programar**

Se aprende a conducir:
¡conduciendo!, no viendo
cómo conducen otros

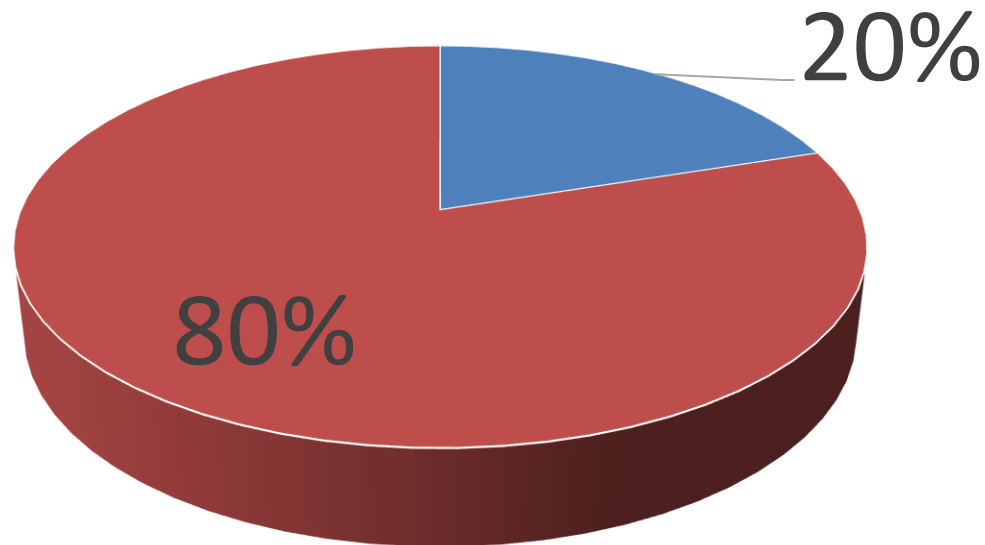


2024/25

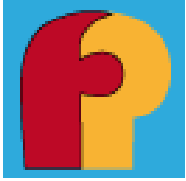
Fundamentos de Programación

Metodología de trabajo

Propuesta de distribución del tiempo estudio



■ Apuntes, documentación ■ **Escribir programas y probarlos**



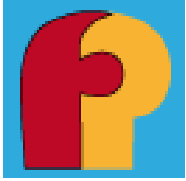
2024/25

Fundamentos de Programación

Metodología de trabajo



- Será la base de nuestro aprendizaje
- Completaremos proyectos que:
 - *Sean interesantes para lo explicado*
 - *Su alcance sea fácil comprender*
 - *Usen diferentes elementos del lenguaje Java*
 - *Tengan un tamaño razonable*



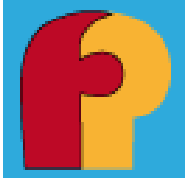
2024/25

Fundamentos de Programación

Lo más importante ...



La constancia...



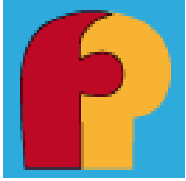
2024/25

Fundamentos de Programación

Sistema de Evaluación

Dos opciones para superar la asignatura:

- Evaluación continua (*recomendada*)
- Evaluación ordinaria



2024/25

Fundamentos de Programación

Evaluación continua

Se realiza a lo largo del curso y se divide en dos cuatrimestres TOTALMENTE diferenciados y con la siguiente estructura cada uno:

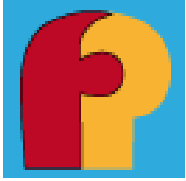


2 exámenes teóricos por cuatrimestre

1 examen práctico por cuatrimestre



iiiPara presentarse al examen práctico de un cuatrimestre hay que obtener una media superior o igual a 4 entre los exámenes teóricos de dicho cuatrimestre!!!



2024/25

Fundamentos de Programación

Evaluación continua – Fechas 2º Cuatr.

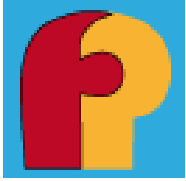
Exámenes teóricos (*cuestionarios en papel*):

T3 (NB 1,2,3): **14 de marzo de 2025**

T4 (NB 4): **19 de mayo de 2025**

Examen práctico (*proyecto sobre ordenador*):

P2: **28 de mayo de 2025**



2024/25

Fundamentos de Programación

Evaluación continua – Calificación

SI $(T1+T2)/2 \geq 4$, ENTONCES

$$C1 = 0,1 \times (T1 + T2) + 0,8 \times P1 \text{ (sobre 10 puntos)}$$

SI NO, $C1 = 0$

SI $(T3+T4)/2 \geq 4$, ENTONCES

$$C2 = 0,1 \times (T3 + T4) + 0,8 \times P2 \text{ (sobre 10 puntos)}$$

SI NO, $C2 = 0$

SI $C1 \geq 4$ Y $C2 \geq 4$, ENTONCES

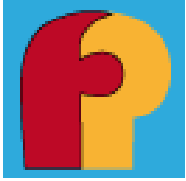
$$\text{Nota_EC} = (C1 + C2) / 2 \quad \text{(sobre 10 puntos)}$$

SI NO,

$$\text{Nota_EC} = \text{mínimo} (4, (C1 + C2) / 2)$$

Ti = Examen teórico, Pi = Examen práctico, Ci = Nota cuatrimestre

Se supera la asignatura por evaluación continua si se cumple que la calificación de la evaluación continua es igual o superior a 5 puntos ($\text{Nota_EC} \geq 5$)



2024/25

Fundamentos de Programación

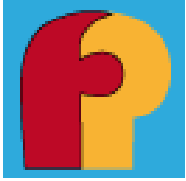
Evaluación ordinaria: fechas

Fechas de las convocatorias:

Primera: **11 junio 2025**

Segunda: **9 julio 2025**

Tercera: **en octubre 2025 (*) Solo repetidores**

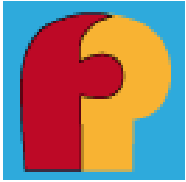


2024/25

Fundamentos de Programación

Evaluación ordinaria: Primera convocatoria

- Un solo examen.
- El examen tendrá dos partes diferenciadas en horarios distintos (**F1** y **F2**), una por cada cuatrimestre. **F1** será *Python* y **F2** será *Java*
- *Quien haya obtenido 4 o más puntos en un cuatrimestre por evaluación continua, podrá optar por presentarse solo a la parte correspondiente al otro cuatrimestre, o a las dos partes del examen.*
- Cada parte se puntuará con 10 puntos.



2024/25

Fundamentos de Programación

Evaluación ordinaria: Primera convocatoria

CALIFICACIÓN:

SI **F1** (o **C1**) ≥ 4 Y **F2** (o **C2**) ≥ 4 , ENTONCES

Nota_1ªConv = (**F1**(o **C1**) + **F2**(o **C2**)) / 2 (sobre 10 puntos)

SI NO

Nota_1ªConv = mínimo (4, (**F1**(o **C1**) + **F2**(o **C2**)) / 2)

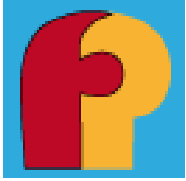
Fi = parte del examen Final de la convocatoria

Ci = Nota del cuatrimestre por evaluación continua

Quien haya realizado las pruebas de la evaluación continua y obtenido una nota mayor o igual a 4 en **C1** o **C2**, puede presentarse únicamente a una de las partes del examen final. Para el cálculo de la nota final, se tendrá en cuenta la nota **C1** o **C2** correspondiente a la parte del examen final a la que no se presente.

Se supera la asignatura en la evaluación ordinaria si:

- la nota de la primera convocatoria es superior o igual a 5 (**Nota_1ªConv** ≥ 5).



2024/25

Evaluación - Ejemplos

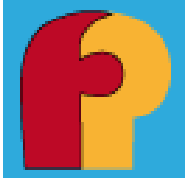


ESTUDIANTE 1

Evaluación continua:

- Primer cuatrimestre (**C1**) 9 puntos.
- Segundo cuatrimestre (**C2**) 8 puntos.
- **Nota** = $(9+8) / 2 = 8,5$

Aprueba en la evaluación continua



2024/25

Fundamentos de Programación

Evaluación - Ejemplos

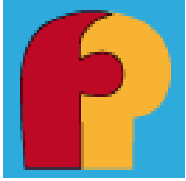


ESTUDIANTE 2

Evaluación continua:

- Primer cuatrimestre (C1) 6 puntos
- Segundo cuatrimestre (C2) 4 puntos.
- **Nota**=(6+4)/2= 5 puntos

Aprueba en la evaluación continua



2024/25

Fundamentos de Programación

Evaluación - Ejemplos



Evaluación continua:

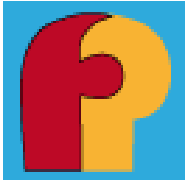
- Primer cuatrimestre (**C1**) 9 puntos
- Segundo cuatrimestre (**C2**) 3 puntos.
- **Nota**= no se hace media

Suspende en la evaluación continua

ESTUDIANTE 3

Opciones para aprobar en la **evaluación ordinaria**:

- Se examina de **todo** y saca un promedio de 5 o más puntos, con un 4 o más en cada parte.
- O bien, se examina del **segundo cuatrimestre** y saca 4 o más.



2024/25

Fundamentos de Programación

Evaluación - Ejemplos



ESTUDIANTE 4

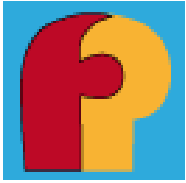
Evaluación continua:

- Primer cuatrimestre (**C1**) 6 puntos.
- Segundo cuatrimestre 3 puntos como nota media de **T3** Y **T4**. No se puede presentar a P2.
- **Nota**: No se hace media

Suspende en la evaluación continua

Opciones para aprobar en la **evaluación ordinaria**:

- Se examina de **todo** y saca un promedio de 5 o más, con un 4 o más en cada parte.
- O bien, se examina del **segundo cuatrimestre** y saca 4 o más.



2024/25

Fundamentos de Programación

Evaluación - Ejemplos

Evaluación continua:

- Primer cuatrimestre (**C1**) 4 puntos.
- Segundo cuatrimestre (**C2**) 3 puntos.
- **Nota**=No se hace media

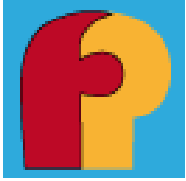


Suspende en la evaluación continua

ESTUDIANTE 5

Opciones para aprobar en la evaluación ordinaria:

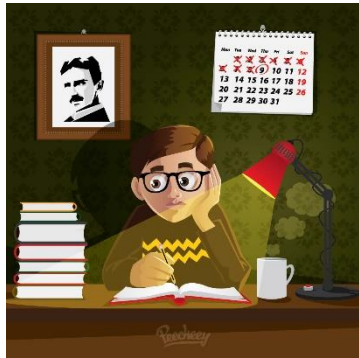
- Se examina de **todo** y saca un promedio de 5 o más, con un 4 o más en cada parte.
- O bien, se examina del **segundo cuatrimestre** y saca 6 o más.



2024/25

Fundamentos de Programación

Evaluación - Ejemplos



ESTUDIANTE 6

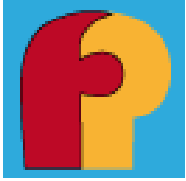
Evaluación continua:

- Primer cuatrimestre 4 puntos.
- Segundo cuatrimestre 4 puntos.
- **Nota**=(4+4)/2= 4 puntos

Suspende en la evaluación continua

Opciones para aprobar en la evaluación ordinaria:

- Se examina de todo y saca un promedio de 5 o más, con 4 o más en cada parte.
- O bien, se examina de una parte y saca 6 o más.



2024/25

Fundamentos de Programación

Evaluación - Ejemplos



ESTUDIANTE 7

Evaluación continua:

- Primer cuatrimestre (**C1**) 3 puntos.
- Segundo cuatrimestre (**C2**) 2 puntos.
- **Nota**=No se hace media

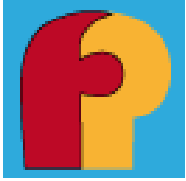
Suspende en la evaluación continua

Opciones para aprobar en la evaluación ordinaria:

- Se examina de todo y saca un promedio de 5 o más, con 4 o más en cada parte.



**¡¡ Se debe examinar
de toda la materia!!**

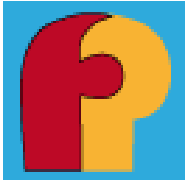


2024/25

Fundamentos de Programación

Evaluación ordinaria: Segunda convocatoria

- Un solo examen.
- El examen tendrá dos partes diferenciadas en horarios distintos (**F1** y **F2**), una por cada cuatrimestre. **F1** será *Python* y **F2** será *Java*
- *Quien haya obtenido 5 o más puntos (lo arrastre de evaluación continua o lo obtuviese en la 1ª convocatoria de la evaluación ordinaria) en un cuatrimestre por evaluación continua, podrá optar por presentarse solo a la parte correspondiente al otro cuatrimestre, o a las dos partes del examen.*
- Cada parte se puntuará con 10 puntos.



2024/25

Fundamentos de Programación

Evaluación ordinaria: Segunda convocatoria

CALIFICACIÓN:

SI **F1** (o C1) ≥ 5 Y **F2** (o C2) ≥ 5 , ENTONCES

Nota_2ªConv = (F1(o C1) + F2(o C2)) / 2 (sobre 10 puntos)

SI NO

Nota_2ªConv = mínimo (4, (F1(o C1) + F2(o C2)) / 2)

Fi = parte del examen de la convocatoria

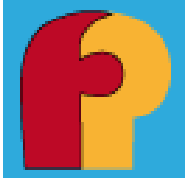
Ci = Nota del cuatrimestre por evaluación continua

Un alumno puede presentarse únicamente a una de las partes del examen de la segunda convocatoria, si en la correspondiente a la otra parte “arrastra” 5 o más puntos.

Para el cálculo de la nota final, se tendrá en cuenta la nota **F1** o **F2** correspondiente a la parte del examen que se presente, y la nota obtenida en la primera convocatoria en la parte a la que no se presente.

El estudiante aprueba la asignatura en la segunda convocatoria si:

- la nota de la segunda convocatoria es superior o igual a 5 (**Nota_2ªConv** ≥ 5).

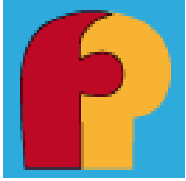


2024/25

Fundamentos de Programación

Evaluación ordinaria: Tercera convocatoria

Análogo a la segunda convocatoria, pero el alumno se debe presentar a **TODA la materia de la asignatura**, independientemente de los resultados de las convocatorias anteriores. ¡No se guarda calificación alguna!



2024/25

Fundamentos de Programación

Medios de contacto

Además de asistir de forma **proactiva** a las clases de teoría y laboratorio



Tutorías



E-mail

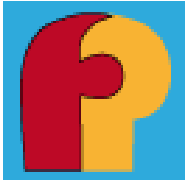


Foros



@fundprog

¡Usar cuenta de @alum.us.es!



2024/25

Fundamentos de Programación

Material de la asignatura en EV



Presentaciones y documentación



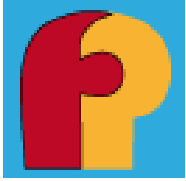
Foros



Anuncios



Inscripción exámenes

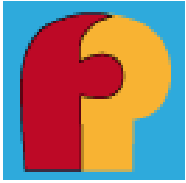


2024/25

Fundamentos de Programación

¿Qué debo hacer para comenzar?

- Leerme el documento sobre el entorno de trabajo
- Instalar en mi ordenador las herramientas y...
- **Comenzar a programar cuanto antes.**



2024/25

1

Fundamentos de Programación

Instalación de herramientas

The screenshot shows the Oracle Java Downloads page. The header includes the Oracle logo and navigation links: Productos, Sectores, Recursos, Clientes, Fagonadura, Desarrolladores, and Compañía. There are also links for 'Ver cuentas' and 'Comunicarse con Ventas'. The main heading is 'Descargas de Java' with the Java logo. Below this, there are tabs for 'Herramientas y recursos', 'Descargas de Java', and 'Archivo Java'. The main content area features a promotional banner for 'JavaOne 2025 - 18 al 20 de marzo de 2025', stating that JavaOne 2025 marks the return of the main conference for Java developers. It mentions the 30th anniversary of Java and the launch of Java 24. A 'Regístrate ahora y ahorra \$100' button is present. Below the banner, there are links for '¿Buscas otras descargas de Java?', 'Compilaciones de acceso anticipado de OpenJDK', and 'JRE para consumidores'.

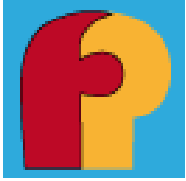
Java 23, Java 21 y versiones anteriores ya están disponibles

JDK 23 es la última versión de la plataforma Java SE.

[Obtén más información sobre la suscripción a Java SE](#)

The screenshot shows the Eclipse Foundation website. The header includes the Eclipse Foundation logo and navigation links: Proyectos, Partidarios, Colaboraciones, Recursos, and La Fundación. There is a 'Descargar' button. The main heading is 'La comunidad para la colaboración abierta y la innovación'. Below this, there is a paragraph: 'La Fundación Eclipse ofrece a nuestra comunidad global de individuos y organizaciones un entorno maduro, escalable y favorable a las empresas para la colaboración y la innovación en software de código abierto.' A 'Más información' button is present. Below this, there is a section titled 'Nuevo informe: El valor comercial del OSS en la industria automotriz'. It includes a paragraph: 'Descubre cómo OSS puede ayudar a la industria automotriz a satisfacer la creciente demanda de vehículos innovadores, rentables y sustentables descargando nuestro nuevo informe de encuesta.' A 'Descarga hoy' button is present.

2



2024/25

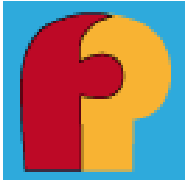
Fundamentos de Programación

Presentación de la asignatura

Temas de teoría

Bloque 1: Introducción a Java	
Autor: Miguel Toro, Mariano González. Revisor: Fermin Cruz. Última modificación: 14/2/2018.	
Tabla de contenidos	
1.	Conceptos básicos de la P.O.O. 3
1.1	Programación Orientada a Objetos 3
1.2	Objeto 3
1.3	Interfaz 4
1.4	Clases 6
1.5	Otros conceptos y ventajas de la POO 6
1.6	Estructura de un programa en Java 7
2.	Elementos básicos del lenguaje 9
2.1	Identificadores 9
2.2	Palabras reservadas de Java 9
2.3	Literales 10
2.4	Comentarios 10
3.	Tipos de datos 10
3.1	Tipos envoltura (wrappers) 11
3.2	Envolturas y concepto de inmutabilidad 11
4.	Variables y Constantes 11
4.1	Variables 11
4.2	Constantes 12
5.	Expresiones y operadores 12
5.1	Expresiones 12
5.2	Operadores 13
5.3	Precedencia y asociatividad de los operadores 13
6.	El tipo String, tipos para el manejo de fechas 14
6.1	Tipo String 14
6.2	Tipos para el manejo de fechas y horas 14
7.	Escritura de datos en pantalla 16
8.	Sentencias de control selectivas 17
8.1	Sentencia if-else 17
8.2	Sentencia switch 17

Temas teoría (ev)



2024/25

Fundamentos de Programación

Presentación de la asignatura

Laboratorio – Proyectos de laboratorio

WSJava - T01.1-Figuras/src/fp/figuras/CirculoImpl.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

- T01.1-Figuras
 - src
 - fp.figuras
 - Canvas.java
 - Circulo.java
 - CirculoImpl.java
 - Cuadrado.java
 - CuadradoImpl.java
 - Figura.java
 - Monigote.java
 - MonigoteImpl.java
 - Triangulo.java
 - TrianguloImpl.java
 - TrianguloImpl2.java
 - fp.figuras.test
 - JRE System Library [jre]
 - doc
 - T01.1-Figuras (sol)
 - T01.2-Casa
 - T01.2-Casa (sol)
 - T01.3-ClaseLaboratorio
 - T01.3-ClaseLaboratorio (sol)

CirculoImpl.java

```
1 package fp.figuras;
2
3 import java.awt.geom.*;
4
5 /**
6  * Un círculo que puede ser manipulado y dibujarse en un canvas.
7  * @author Adaptado por Toñi Reina para FP de 1 código de
8  * Michael Welling and David J. Barnes
9  * @version 1.0.0
10 */
11
12 public class CirculoImpl implements Circulo {
13     private Integer diametro;
14     private Integer xPosition;
15     private Integer yPosition;
16     private String color;
17     private Boolean esVisible;
18
19     /**
20      * Create a new circle at default position with default color.
21      */
22     public CirculoImpl() {
23         diametro = 68;
24         xPosition = 230;
25         yPosition = 90;
26         color = "blue";
27         esVisible=false;
28     }
29
30 }
```

Outline

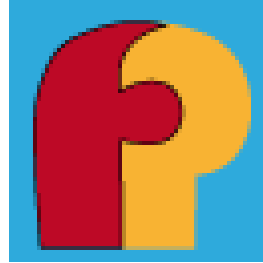
- fp.figuras
 - CirculoImpl
 - diametro : Integer
 - xPosition : Integer
 - yPosition : Integer
 - color : String
 - esVisible : Boolean
 - CirculoImpl()
 - hacerVisible() : void
 - hacerInvisible() : void
 - moverDerecha() : void
 - moverIzquierda() : void
 - moverArriba() : void
 - moverAbajo() : void
 - moverHorizontalmente(Integer) : void
 - moverVerticalmente(Integer) : void
 - moverHorizontalmenteDespacio(Integer) : void
 - moverVerticalmenteDespacio(Integer) : void
 - cambiarTamanyo(Integer) : void
 - cambiarColor(String) : void
 - dibujar() : void
 - borrar() : void

Problems

0 items

Description	Resource	Path	Location	Type
-------------	----------	------	----------	------

fp.figuras.CirculoImpl.java/fp.figuras - T01.1-Figuras/src



Fundamentos de Programación

¿Otras dudas? ¿Preguntas?