Unidad 01.1

Objetos y clases

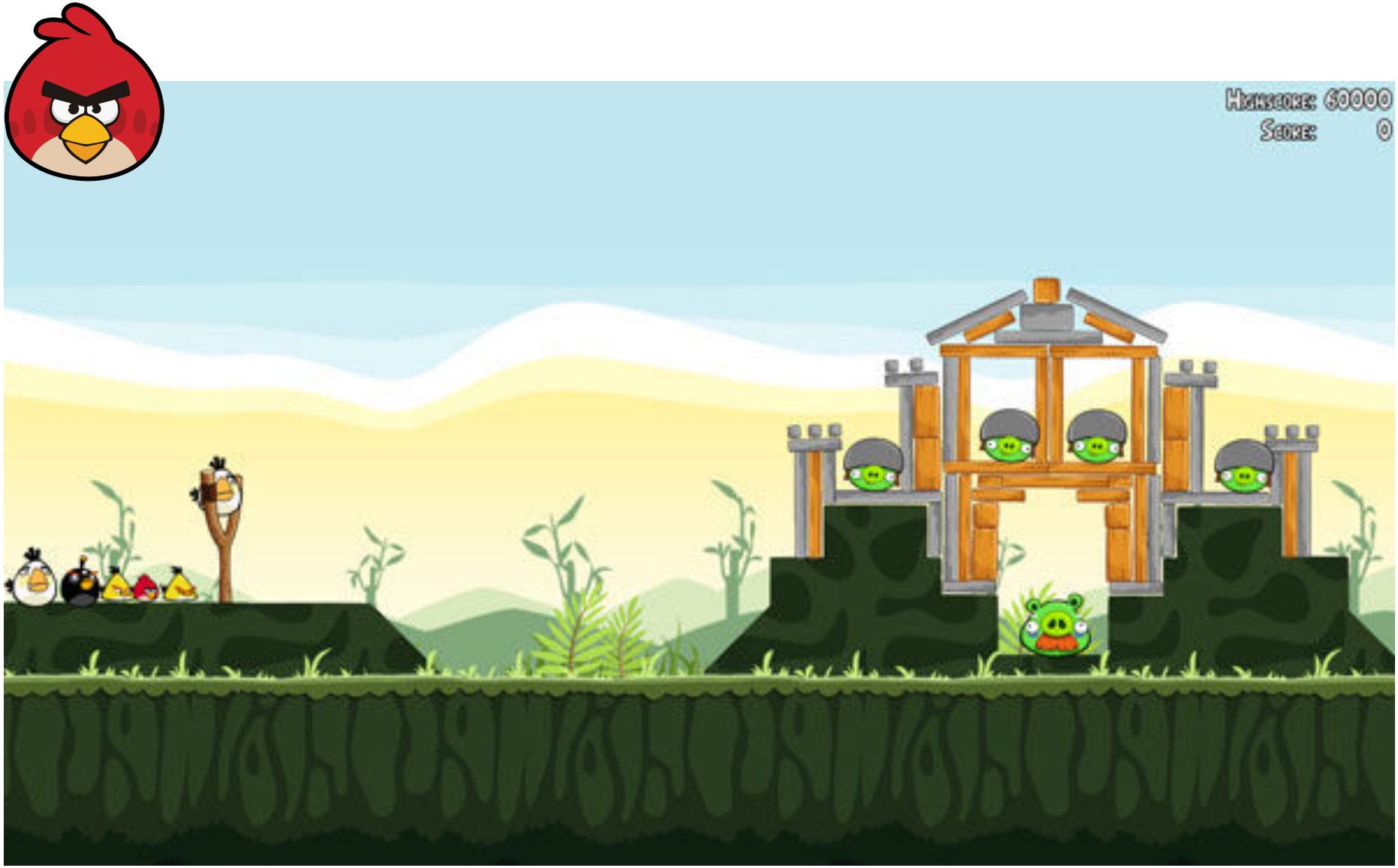
Control de versiones

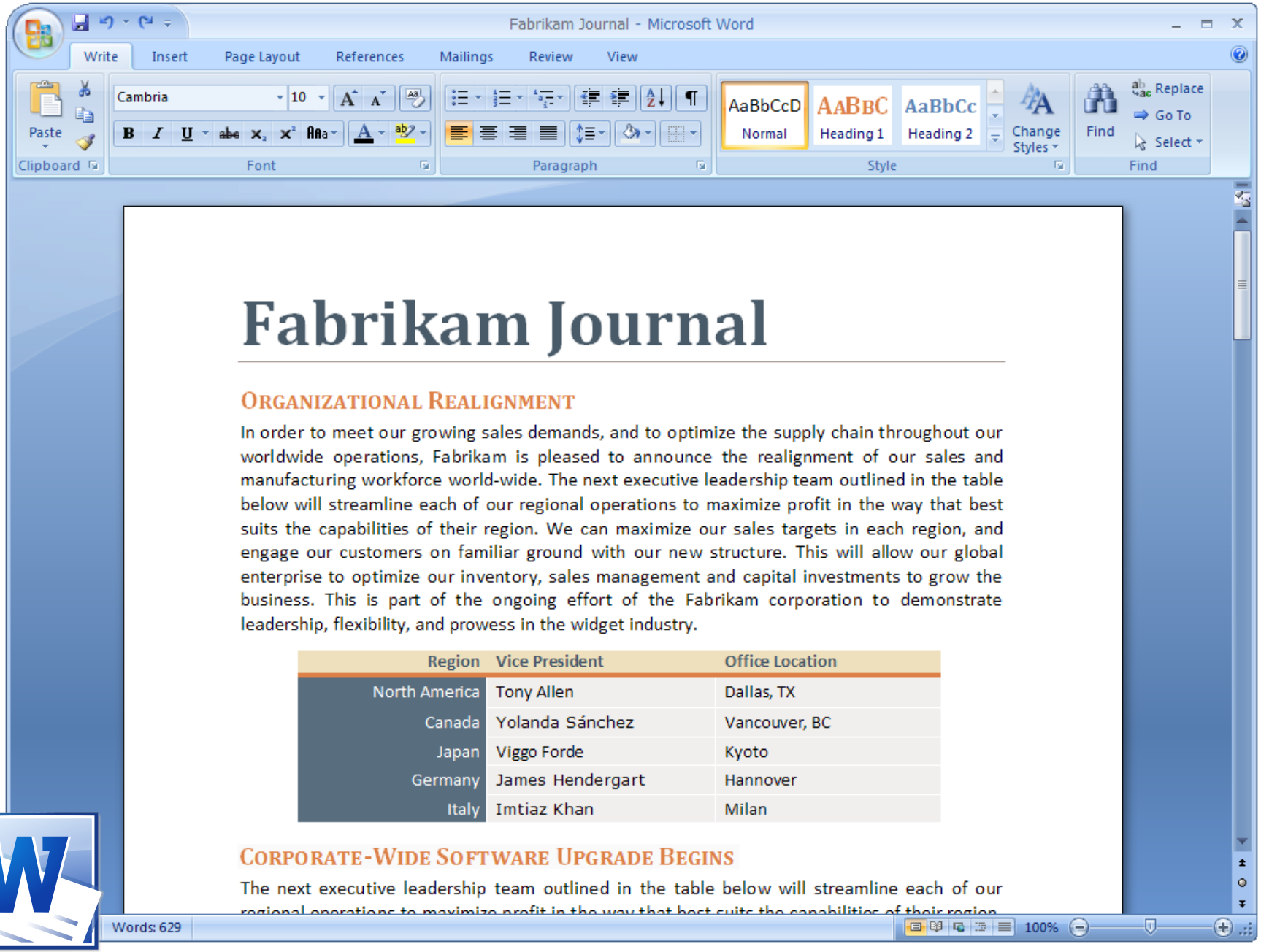
github.com/miguelbayon

Objetos (objects)

Objetos concretos del dominio del problema


4G 6:04 PM
Edit
Chats
Broadcast Message
New Group
Whitmans Chat
5:11 PM
Jack:
I'm having a small party tonight...
Peter Whitman
1:14 PM
New place spotted
Blue Rock Coffee
Vladimir Wolodarsky
10:45 AM
Belafonte Crew
Yesterday
Cap:
Guys I need you to come...
Suzy Bishop
10/5/12
Can't wait!
Edit your message notifications, set profile photo and many other things
Chats
Settings
Chats (1)
Whitmans Chat
Alice, Jack, Francis, Pet...
Info
Francis Whitman joined
How about a small adventure tomorrow?
4:36 PM
Angel Island
Golden Gate National Recreati...
4:40 PM
Alice Whitman:
Wonderful idea!
4:41 PM
Francis Whitman:
I've heard they have these segway tours
4:42 PM
Play
Forward
0:37
4:51 PM
Jack Whit...
Download image 9 KB
Send
Download image 9 KB
5:02 PM
Alice Whitman:
Just check this out guys
5:03 PM
Alice W...
Play
Forward
5:05 PM
Jack Whitman:
She was singing along while Lady Gaga played!
5:06 PM
5:06 PM
Lol
Btw the veterinary contact I promised to share
5:26 PM
Molly Clark DVM
5:28 PM
Send


Highscore: 60000
Score: 0


Fabrikam Journal - Microsoft Word
Write
Insert
Page Layout
References
Mailings
Review
View
Cambria
10
Paste
Clipboard
Font
Paragraph
AaBbCcD
Normal
AaBbC
Heading 1
AaBbCc
Heading 2
Change Styles
Style
Find
Replace
Go To
Select
Find
Fabrikam Journal
ORGANIZATIONAL REALIGNMENT
In order to meet our growing sales demands, and to optimize the supply chain throughout our worldwide operations, Fabrikam is pleased to announce the realignment of our sales and manufacturing workforce world-wide. The next executive leadership team outlined in the table below will streamline each of our regional operations to maximize profit in the way that best suits the capabilities of their region. We can maximize our sales targets in each region, and engage our customers on familiar ground with our new structure. This will allow our global enterprise to optimize our inventory, sales management and capital investments to grow the business. This is part of the ongoing effort of the Fabrikam corporation to demonstrate leadership, flexibility, and prowess in the widget industry.
Region
Vice President
Office Location
North America
Tony Allen
Dallas, TX
Canada
Yolanda Sánchez
Vancouver, BC
Japan
Viggo Forde
Kyoto
Germany
James Hendergart
Hannover
Italy
Imtiaz Khan
Milan
CORPORATE-WIDE SOFTWARE UPGRADE BEGINS
The next executive leadership team outlined in the table below will streamline each of our
Words: 629
100%

Clases (classes)

Tipos abstractos de objetos

Clases vs Objetos

¿Que color?
¿Cuantas puertas?
¿Que velocidad?
¿Donde esta ubicado?

Coche

Objetos

Coche

Convenciones de nombrado

`miCocheNuevo`

Coche

`Coche`

ACTIVIDAD

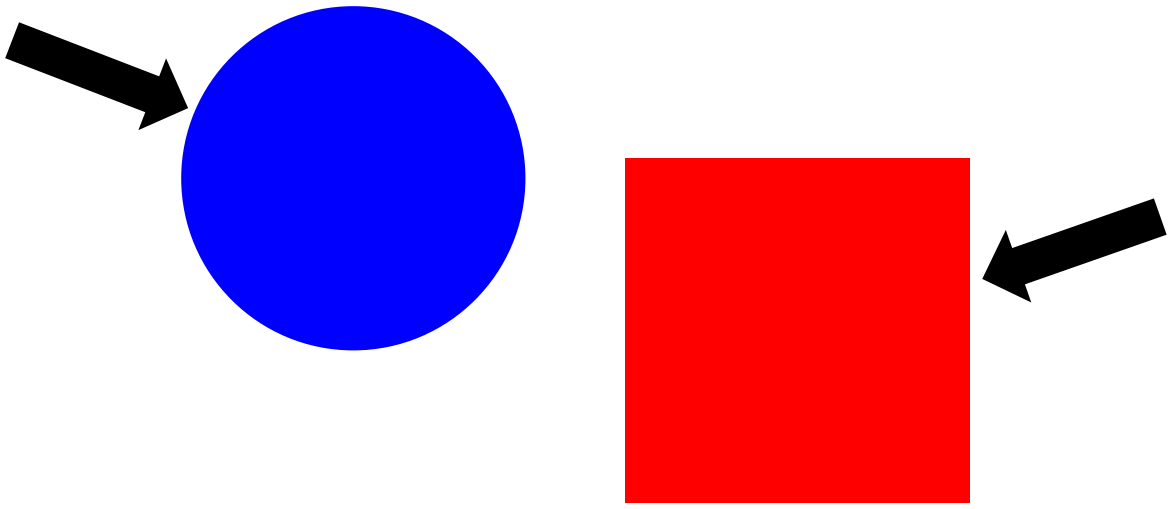
Proyecto “Figures”

1) Clona el proyecto `figures` del repositorio de mi GitHub
2) Abrélo con BlueJ
3) Crea un círculo llamado `circle1`
4) Crear un cuadrado llamado `square1`
5) Invocar `makeVisible()` en `circle1`
6) Invocar `makeVisible()` en `square1`

Métodos (methods)

Nos podemos comunicar con los objetos **invocando** sus métodos

Así conseguimos que hagan “algo”



ACTIVIDAD

Proyecto “Figures”

1. ¿Qué sucede si invocamos dos veces al método `moveDown()` de `circle1`?

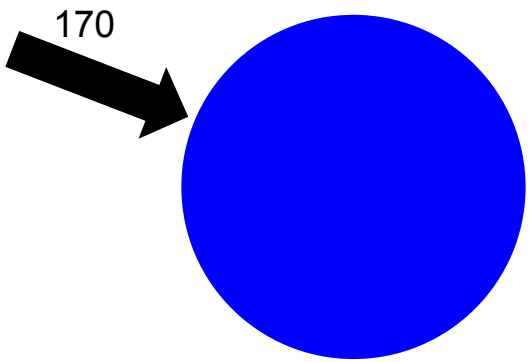
2. ¿Qué sucede si invocamos dos veces al método `makeInvisible()` de `circle1`?

Parámetros (parameters)

Método `moveHorizontal(int distance)`

Permiten pasar información adicional a un método

No los tienen todos los métodos



ACTIVIDAD

Proyecto “Figures”

1. Mueve el cuadrado 60 píxeles a la derecha
2. Mueve el círculo lentamente 100 píxeles a la derecha
3. Haz el círculo el doble de grande aproximadamente
4. Mueve el círculo 200 píxeles a la izquierda

Cabecera de un método (header/signature)

Indica la información que le tenemos que pasar a un método

Por cada parámetro indica tipo y nombre del parámetro

Uso de los paréntesis

Número de parámetros de 0 a n, separados por coma

```
void moveHorizontal(int distance)
```

Tipos de datos (data types)

Definen la clase de valor que toma un parámetro

```
void moveHorizontal(int distance)
```

`int` = número entero

```
void changeColor(String newColor)
```

`String` = cadena de caracteres

“`red`”, “`blue`”, “`green`”, “`magenta`”...

Hay más tipos: decimales, letras, si/no...

Comentarios (comments)

Proporcionan ayuda a los humanos

```
//change the color; valid colors are “red”, “yellow”, “blue”,
//”green”, “magenta” and “black”

void changeColor(String newColor)
```

ACTIVIDAD

Proyecto “Figures”

1. Invoca al método `changeColor` con el texto `black` sin comillas como parámetro. ¿Qué sucede?
2. Pon el cuadrado y el circulo del mismo color
3. Pon el cuadrado de color “`orange`”. ¿Qué pasa?

Múltiples objetos

A partir de una clase se pueden crear tantos objetos como queramos

Los métodos cambian **únicamente** a los objetos sobre los que se invocan

ACTIVIDAD

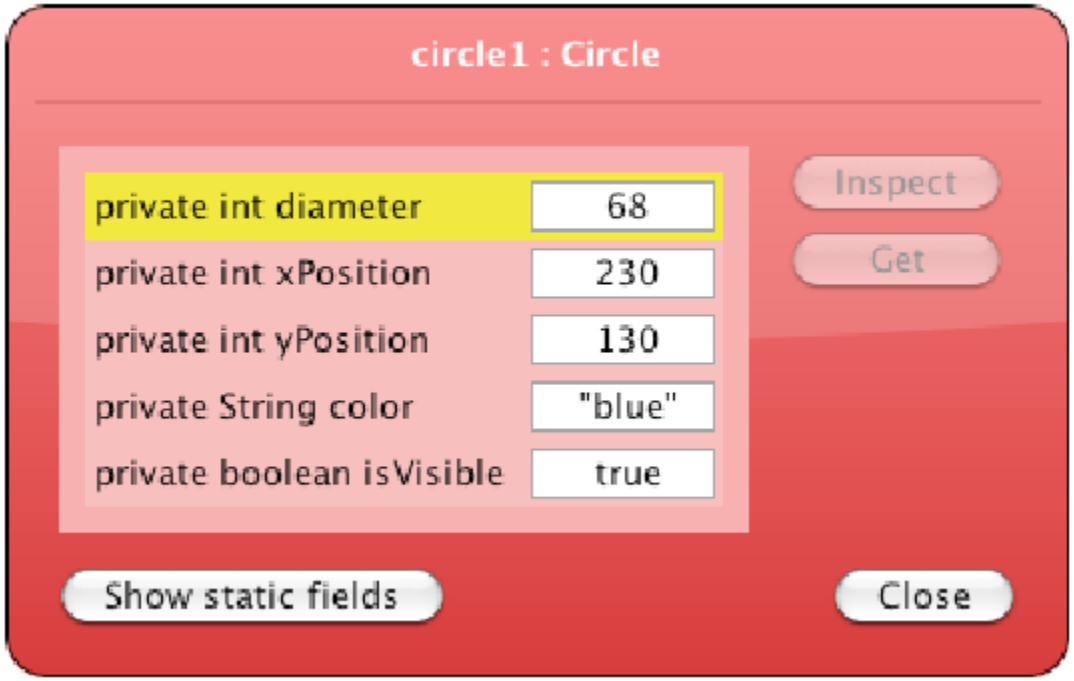
Proyecto “Figures”

1. Consigue que se muestre la siguiente imagen y apunta de forma detallada los pasos que das:

Cuando termines, compacta los pasos que has dado para que queden los menos posibles. ¿Cuántos has necesitado?

Estado de un objeto (state)

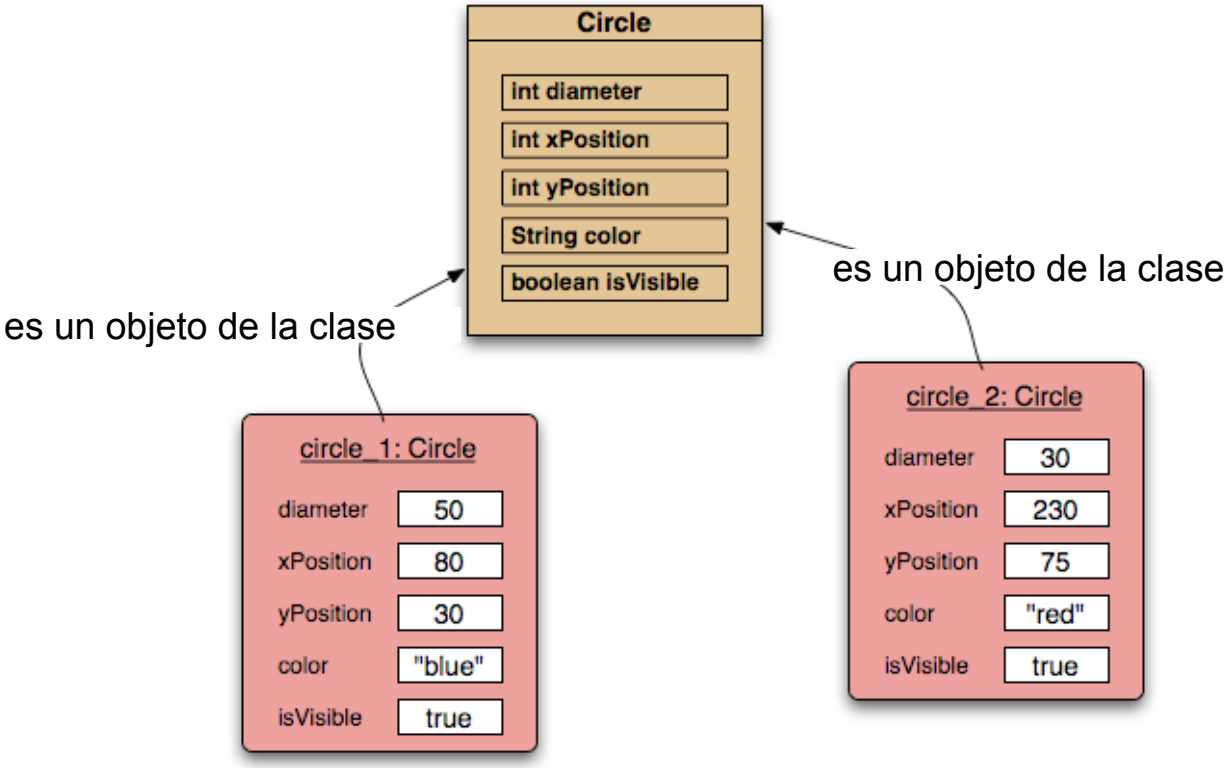
Conjunto de valores que tienen sus atributos (attributes o fields)



Estado de un objeto (state)

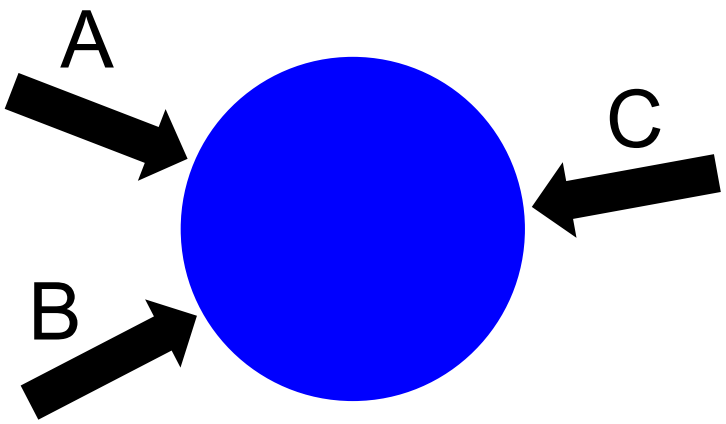
Los atributos están definidos en la clase del objeto

Cada objeto tiene unos valores para el conjunto de atributos



Estado de un objeto (state)

Los métodos son definidos en la clase
e invocados en los objetos



Código Java (Java code)

Cuando programamos escribimos instrucciones.

```
try {
    String host = "jdbc:derby://localhost:1527/Employees";
    String uName = "admin";
    String uPass = "admin";
    Connection con = DriverManager.getConnection(host, uName, uPass);

    Statement stmt = con.createStatement();
    String sql = "SELECT * FROM Workers";
    ResultSet rs = stmt.executeQuery(sql);

    rs.next();
    int id_col = rs.getInt("ID");
    String first_name = rs.getString("First_Name");
    String last_name = rs.getString("Last_Name");
    String job = rs.getString("Job_Title");

    String p = id_col + " " + first_name + " " + last_name + " " + job;
    System.out.println(p);
}
catch ( SQLException err ) {
    System.out.println( err.getMessage( ) );
}
```

Código Java (Java code)

```
Person person1 = new Person();
person1.makeVisible();
person1.moveRight();
```

¿Qué estamos haciendo en este ejemplo?

Uso del punto

Uso del punto y coma

¿Qué es person1?: una **variable**

ACTIVIDAD

Proyecto “Figures”

1. Abre la Terminal de BlueJ
2. Activa `Record methods call`
3. Consigue que se muestre la siguiente imagen poniendo nombres descriptivos a los objetos que vas creando
4. Observa lo que va apareciendo en la Terminal

5. Desactiva `Record methods call`

ACTIVIDAD

Proyecto “Figures”

1. Cierra y abre BlueJ y carga el proyecto “Figures”
2. Abre la ventana **Code Pad** y crea una persona y un cuadrado escribiendo las instrucciones necesarias.
3. ¿Notas algo que te llame la atención en el banco de objetos?

ACTIVIDAD

Proyecto “Figures”

1. Cierra y abre BlueJ y carga el proyecto “Figures”
2. Crea a golpe de ratón dos personas y dos círculos
3. Abre la ventana **Code Pad**
4. Escribiendo código Java consigue que se muestre el siguiente dibujo (utiliza nombres descriptivos para los objetos que crees):

Por cada instrucción escrita debes pulsar ***Enter*** para que se ejecute. Si aparece un error es que o has escrito algo mal o te falta algo por escribir.