



# Algoritmos y Estructura de Datos

---

Unidad 1: Clases y objetos

Tema 4: Clase String





# Tema 4: Clase String

---





# Índice

---

## 1.4 Tema 4: Clase String

- 1.4.1 Descripción
- 1.4.2 Métodos básicos de la clase ***String***
- 1.4.3 Concatenación
- 1.4.4 Recorrido
- 1.4.5 Librería
- 1.4.6 Ejemplo





# Capacidades

---

- Identifica los cambios en la forma de hacer un programa.
- Diseña clases, objetos y maneja cadenas de texto.



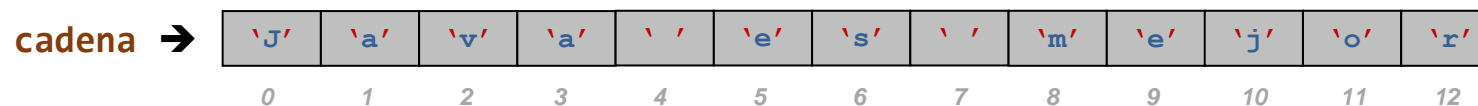


## 1.4.1 Descripción

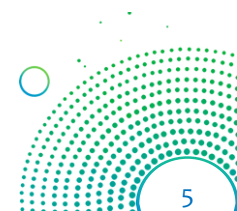
- La clase String cuenta con métodos para manipular *cadenas de texto*.
- Una *cadena de texto* es un conjunto de caracteres dispuestos uno a continuación de otro, donde cada carácter conserva su propio espacio (tamaño en bytes).
- Internamente, la clase String ubica a cada carácter en un espacio independiente y enumera de izquierda a derecha las posiciones, empezando de cero.

Ejemplo:

```
String cadena = "Java es mejor";
```

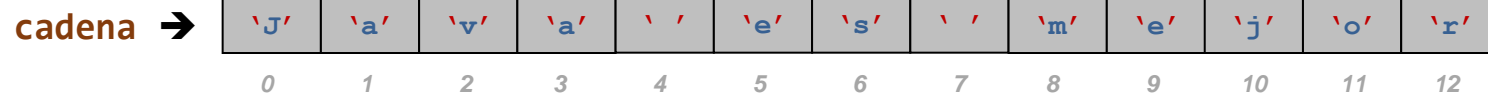


- El acceso a los métodos de la clase String se hace mediante la *variable referencia* (nombre del objeto) y el operador punto (.)





# 1.4.2 Métodos básicos de la clase String



1) `public int length() {`  
`}`

→ Retorna la cantidad de caracteres de la cadena

Ejemplo:

```
int longitud;  
longitud = cadena.length(); // longitud = 13
```

2) `public char charAt(int) {`  
`}`

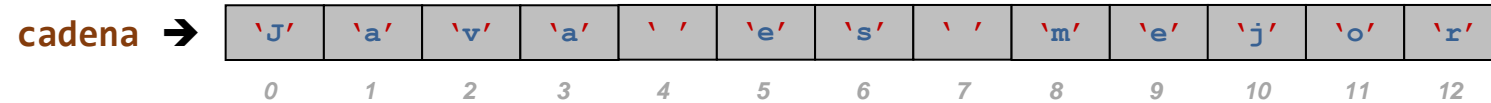
→ Retorna una copia del caracter ubicado en la posición indicada

Ejemplo:

```
char primerCaracter;  
primerCaracter = cadena.charAt(0); // primerCaracter = 'J'  
char ultimoCaracter;  
ultimoCaracter = cadena.charAt(longitud - 1); // ultimoCaracter = 'r'
```



# 1.4.2 Métodos básicos de la clase String



3) `public boolean equals(String)` {  
}

→ Retorna true o false si la cadena que invoca al método coincide en texto con la cadena enviada como parámetro

Ejemplo:

```
String cad1 = "Java es lo máximo";  
boolean sino1 = cadena.equals(cad1); // sino1 = false
```

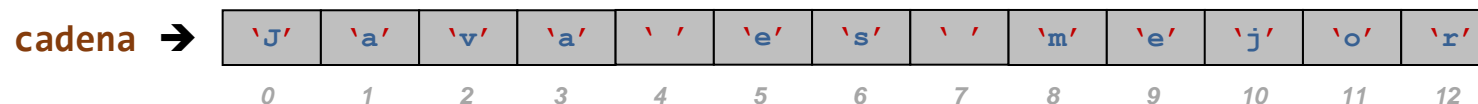
```
String cad2 = "Java es mejor";  
boolean sino2 = cadena.equals(cad2); // sino2 = true
```

```
String cad3 = "Java gusta a todos";  
boolean sino3 = cadena.equals(cad3); // sino3 = false
```

```
String cad4 = "Java es mejor hoy";  
boolean sino4 = cadena.equals(cad4); // sino4 = false
```



# 1.4.2 Métodos básicos de la clase String



4) `public int compareTo(String)` {  
}

→ Retorna un número entero (luego de comparar alfabéticamente la cadena que invoca al método con la cadena enviada como parámetro)

Ejemplo:

**cadena** → "Java es mejor";

**cad1** → "Java es lo máximo";

`int ok1 = cadena.compareTo(cad1); // ok1 = 1`

Nota: el valor de ok1 es la diferencia ASCII de los caracteres **m** y **l**

**cadena** → "Java es mejor";

**cad2** → "Java es mejor";

`int ok2 = cadena.compareTo(cad2); // ok2 = 0`

Nota: el valor de ok2 es la diferencia de longitudes de cadena y cad2

**cadena** → "Java es mejor";

**cad3** → "Java gusta a todos";

`int ok3 = cadena.compareTo(cad3); // ok3 = -2`

Nota: el valor de ok3 es la diferencia ASCII de los caracteres **e** y **g**

**cadena** → "Java es mejor";

**cad4** → "Java es mejor hoy";

`int ok4 = cadena.compareTo(cad4); // ok3 = -4`

Nota: el valor de ok4 es la diferencia de longitudes de cadena y cad4





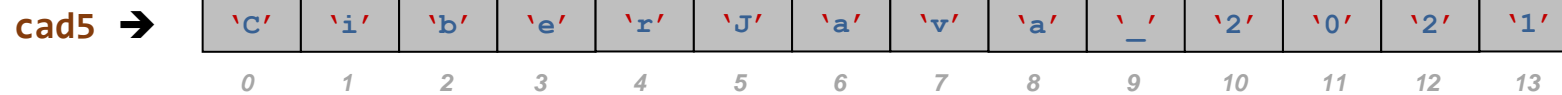
## 1.4.3 Concatenación

- Se utiliza el símbolo más para acoplar información. Una variable cadena puede concatenar textos, caracteres y números. Al final todo sigue siendo un String.

Ejemplo:

```
String cad5 = "Ciber" + "Java" + '_' + 2021;
```

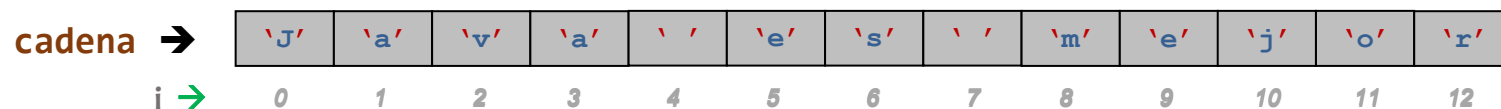
- El resultado es: "CiberJava\_2021"





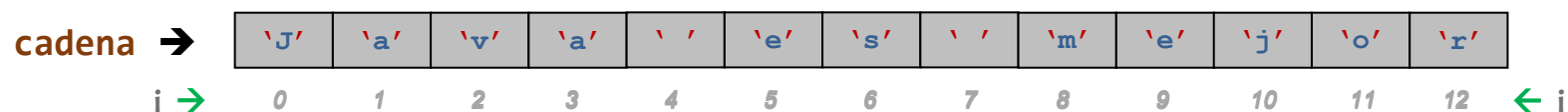
## 1.4.4 Recorrido

- Consiste en contar las posiciones de izquierda a derecha o viceversa.



**EN ASCENSO** (de izquierda a derecha)

```
for (int i=0; i<cadena.length(); i++) {  
    ...  
}
```



**EN DESCENSO** (de derecha a izquierda)

```
for (int i=cadena.length()-1; i>=0; i--) {  
    ...  
}
```



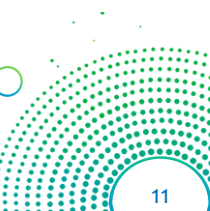
## 1.4.5 Librería

- Es una clase que implementa elementos **static** los cuales podrán ser invocados desde cualquier parte del programa a través de dicha clase, sin necesidad de tener que crear un objeto.

```
package lib;

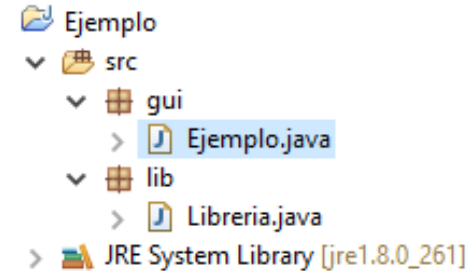
public class Libreria {
    // Constante pública de clase
    public static final double PI = 3.1416;
    // Constructor por defecto
    public Libreria() {
    }
    // Método público de clase
    public static int aleatorio(int min, int max) {
        return (int)((max - min + 1) * Math.random()) + min;
    }
}
```

```
imprimir("valor de PI : " + Libreria.PI);
imprimir("# aleatorio de 3 cifras : " + Libreria.aleatorio(100, 999));
```





## 1.4.6 Ejemplo



- a) A la pulsación del botón Procesar declara e inicializa una **cadena** con un texto.
- b) Visualiza:
  - cantidad de caracteres del texto
  - primer carácter del texto
  - último carácter del texto
  - resultado de comparar el texto con otras cadenas
  - resultado de concatenar información
  - recorrido de una cadena
  - valor de una constante a través de una librería
  - número natural aleatorio a través de una librería





# Conclusiones

---

- La clase String cuenta con métodos para manipular *cadena de texto*.
- A través del símbolo más podemos concatenar información variada. Al final todo es texto.
- Toda cadena se puede recorrer de izquierda a derecha y viceversa.
- Una librería permite crear herramientas de uso genérico para agilizar el diseño de proyectos.

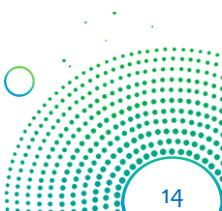




# Referencias bibliográficas

---

- **Joyanes Aguilar Luis.** *Fundamentos de programación: algoritmos, estructuras de datos y objetos.* Madrid, España: McGraw-Hill (005.1 JOYA/A 2021)
- **Lewis John.** *Estructuras de datos con Java: diseño de estructuras y algoritmos.* Madrid, Pearson Educación (005.73 LEWI/E 2021)
- **Deitel Harvey.** *Cómo programar en Java.* México, D.F.: Pearson Educación (005.133J DEIT 2021)



# GRACIAS



## **SEDE MIRAFLORES**

Calle Díez Canseco Cdra 2 / Pasaje Tello  
Miraflores – Lima  
Teléfono: 633-5555

## **SEDE INDEPENDENCIA**

Av. Carlos Izaguirre 233  
Independencia – Lima  
Teléfono: 633-5555

## **SEDE BREÑA**

Av. Brasil 714 – 792  
(CC La Rambla – Piso 3)  
Breña – Lima  
Teléfono: 633-5555

## **SEDE TRUJILLO**

Calle Borgoño 361  
Trujillo  
Teléfono: (044) 60-2000

## **SEDE SAN JUAN DE LURIGANCHO**

Av. Próceres de la Independencia 3023-3043  
San Juan de Lurigancho – Lima  
Teléfono: 633-5555

## **SEDE LIMA CENTRO**

Av. Uruguay 514  
Cercado – Lima  
Teléfono: 419-2900

## **SEDE BELLAVISTA**

Av. Mariscal Oscar R. Benavides 3866 – 4070  
(CC Mall Aventura Plaza)  
Bellavista – Callao  
Teléfono: 633-5555

## **SEDE AREQUIPA**

Av. Porongoche 500  
(CC Mall Aventura Plaza)  
Paucarpata - Arequipa  
Teléfono: (054) 60-3535