# UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

Facultad de Ciencias



## Introducción a las Ciencias de la Computación

Práctica 2: Boolean y el objeto String.

Profesora:
Amparo López Gaona
Ayudante:
Víctor Emiliano Cruz Hernández
Ayudante de Laboratorio:
Kevin Jair Torres Valencia

### **Objetivos**

Al finalizar esta práctica, el alumno habrá reforzado sus conocimientos acerca de la creación y manipulación de objetos en Java de la clase String.

#### Introducción

Un tipo de dato muy utilizado es el de cadenas de caracteres, por tal motivo Java proporciona la clase String para trabajar con ellas. La declaración para datos de esta clase es similar a la declaración de datos de cualquier tipo primitivo, ya que el nombre de la variable para almacenar la referencia a una cadena va precedido de la palabra String. Por ejemplo, con la instrucción String nombre; se declara una variable llamada nombre para un objeto de la clase String.

Para crear un objeto de la clase String puede asignarse una cadena constante en la declaración de la referencia o se puede utilizar el operador new para creación de objetos. Por ejemplo, las dos instrucciones siguientes son equivalentes:

```
String nombre = new String("Hola Mundo");
String nombre = "Hola Mundo";
```

La clase String tiene varios métodos. Para ejecutar cualquier método se debe utilizar la referencia declarada, seguida de un punto y luego el nombre del método y los parámetros entre paréntesis; estos son obligatorios aunque no haya parámetros. La notación es la siguiente: referencialDelObjeto.nombreDelMetodo(parametrosOpcionales);

Por ejemplo, nombre.toUpperCase() llama al método para convertir una cadena en mayúsculas. Algunos métodos de la clase String son:

- char charAt(int)
- boolean equals(String)
- int indexOf(String)
- int length()
- String substring(int,int)
- String toLowerCase()
- String toUpperCase()
- String trim()
- String replace(String, String)

- Integer Integer.valueOf()
- Integer Integer.parseInt()

Además, en java puedes darle color a cadenas de tipo String. A continuación se muestra una tabla con algunos colores y sus respectivos códigos en Java:

Color	Código en Java
Negro	\u001B[30m
Rojo	\u001B[31m
Verde	\u001B[32m
Amarillo	\u001B[33m
Azul	\u001B[34m
Magenta	\u001B[35m
Cyan	\u001B[36m
Blanco	\u001B[37m
Reset	\Ŏ01B[Om

Con respecto a la tabla anterior, puedes declarar los siguientes colores:

```
public static final String RESET = "\u001B[0m";
public static final String BLACK = "\u001B[30m";
public static final String RED = "\u001B[31m";
public static final String GREEN = "\u001B[32m";
public static final String YELLOW = "\u001B[33m";
public static final String PURPLE = "\u001B[35m";
public static final String CYAN = "\u001B[36m";
public static final String WHITE = "\u001B[37m";
public static final String BLUE = "\u001B[34m";
```

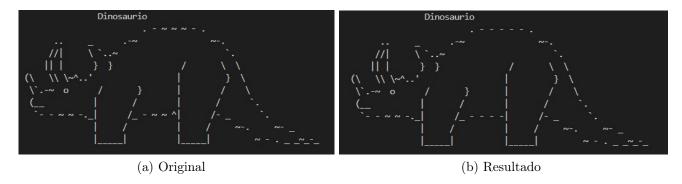
Estas definiciones se pueden utilizar de la siguiente manera:

```
System.out.println(RED + cadena + RESET);
```

Donde cadena es de tipo String.

#### Desarrollo

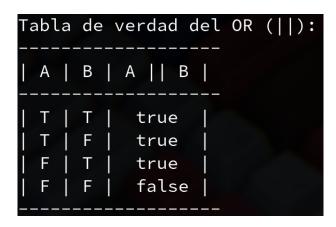
1. (Arte ASCII) Modifica el archivo Problema1. java y maneja el objeto de tipo String llamado dino para insertar unas modificaciones a la figura en ASCII. A continuación se muestra un ejemplo de salida:



Tansformación usando String

**NOTA:** No debes modificar directamente camello, lo debes hacer con el uso de los métodos que nos proporciona String.

2. (**Tabla de verdad**) Escribe un programa que imprima la tabla de verdad OR. La salida salida se debe ver semejante a la siguiente:



**NOTA:** Solo debes usar los tipos String y boolean. Los valores de verdad en la tabla deben estar dados por el or de java.

3. (Suma de fecha) Declara la fecha como un valor de tipo String en el siguiente formato dia-mes-anio. Ahora substrae los valores numéricos y conviertelos a tipo Int. Por último suma los tres números resultantes. E.g. Para 12-02-2024 el resultado final debe ser 2038.

## Formato de Entrega

- 1. Las prácticas se entregarán en parejas.
- 2. Cada práctica (sus archivos y directorios) deberá estar contenida en un directorio llamado EquipoX\_pY, donde:
  - (a) X es el número de equipo correspondiente.
  - (b) Y es el número de la práctica.

Por ejemplo: Equipo10\_p13

- 3. NO incluir los archivos .class dentro de la carpeta.
- 4. Los archivos de código fuente deben estar documentados.
- 5. Se pueden discutir y resolver dudas entre los integrantes del grupo. Pero cualquier práctica plagiada total o parcialmente será penalizada con cero para los involucrados.
- 6. La práctica se debe subir al Github Classroom correspondiente.
- 7. La entrega en classroom debe contener el link HTTPS y SSH de su repositorio y es lo único que se debe entregar.
- 8. El horario y día de entrega se acordará en la clase de laboratorio y no deberá sobrepasar 2 clases de laboratorio.