

Tema 8 MANEJO DE CADENAS

8.1. LA CLASE STRING

Dentro de un objeto de la clases *String* o *StringBuffer*, Java crea un array de caracteres de una forma similar a como lo hace el lenguaje C++. A este array se accede a través de la función miembro de la clase. Los strings u objetos de la clase *String* se pueden crear explícitamente o implícitamente. Para crear un **string implícitamente** basta poner una cadena de caracteres entre comillas dobles. Por ejemplo, cuando se escribe

```
System.out.println("El primer programa");
```

Java crea un objeto de la clase *String* automáticamente.

Para crear un **String explícitamente** escribimos

```
String str=new String("El primer programa");
```

También se puede escribir, alternativamente

```
String str="El primer programa";
```

Para crear un String nulo se puede hacer de estas dos formas

```
String str="";
```

```
String str=new String();
```

Un String nulo es aquél que no contiene caracteres, pero es un objeto de la clase *String*. Sin embargo, `String str;` está declarando un objeto *str* de la clase *String*, pero aún no se ha creado ningún objeto de esta clase.

8.2. METODOS DE LA CLASE String

La clase *String* proporciona métodos para el manejo de cadenas de caracteres:

MÉTODO	DESCRIPCIÓN
length()	Devuelve la longitud de la cadena
indexOf("caracter")	Devuelve la posición de la primera aparición de carácter
lastIndexOf("caracter")	Devuelve la posición de la última aparición de carácter
charAt(n)	Devuelve el carácter que está en la posición n
toUpperCase()	Devuelve la cadena convertida a mayúsculas
toLowerCase()	Devuelve la cadena convertida a minúsculas
equals("cad")	Compara dos cadenas y devuelve true si son iguales
equalsIgnoreCase("cad")	Igual que equals pero sin considerar mayúsculas y minúsculas
compareTo(OtroString)	Devuelve 0 si las dos cadenas son iguales. <0 si la primera es alfabéticamente menor que la segunda ó >0 si la primera es alfabéticamente mayor que la segunda.
compareToIgnoreCase(OtroString)	Igual que compareTo pero sin considerar mayúsculas y minúsculas.
valueOf(N)	Método estático. Convierte el valor N a String. N puede ser de cualquier tipo.

Ejemplo. Crear una clase donde se muestre el manejo de cadenas y que implemente los métodos:

- contar la cantidad de vocales en una cadena.
- Verificar si una cadena es palíndrome.
- Contar la cantidad de ocurrencias de una letra en una cadena

```
public class ManejoString {
    public static void main(String[] args) {
        ManejoString obj = new ManejoString();
        String cad = "primera cadena";
        String cad1 = "PRIMERA cadena";
        System.out.println("longitud " + cad.length());
        System.out.println("valor mayusculas " + cad.toUpperCase());
        System.out.println("el substring " + cad.substring(4, 7));
        System.out.println("caracter de la posicion 8: " + cad.charAt(8));
        if (cad.equals(cad1) == true) {
            System.out.println("las cadenas son iguales");
        } else {
            System.out.println("las cadenas no son iguales");
        }
    }
}
```

```
}  
if (cad.equalsIgnoreCase(cad1) == true) {  
    System.out.println("las cadenas son iguales");  
} else {  
    System.out.println("las cadenas no son iguales");  
}  
  
//llamando a los métodos creados dentro de la clase  
obj.vocales(cad);  
System.out.println("Introduzca una cadena");  
String c = sc.nextLine();  
System.out.println("Digite un caracter");  
String a = sc.nextLine();  
char aa = a.charAt(0);  
obj.palindrome(c);  
obj.ocurrencia(c, aa);  
obj.contar(c);  
obj.encriptar(c);  
sc.close();  
  
}  
}  
public void vocales(String cadena){  
    int i,cv=0;  
    int n=cadena.length();  
    for(i=0;i<n;i++){  
        char a=cadena.charAt(i);  
        if(a=='a' || a=='e' || a=='i' || a=='o' || a=='u'){  
            cv=cv+1;  
        }  
    }  
    System.out.println("La cadena tiene "+ cv+"vocales");  
}
```

```
public void palindrome(String cadena){
    int i; String cad="";
    int n=cadena.length();
    System.out.println("nn:"+n);
    for(i=n-1;i>=0;i--){
        cad=cad+cadena.charAt(i);
    }
    System.out.println("la cadena"+cad);
    if(cadena.equals(cad)==true)
        System.out.println("La palabra es palindrome");
    else
        System.out.println("La palabra no es palindrome");
}
```

```
public void ocurrencia(String cad,char car){
    int i,co=0;
    for(i=0;i<cad.length();i++){
        if(cad.charAt(i)==car)
            co=co+1;
    }
    System.out.println("La palabra tiene "+ co);
}
```

Ejemplo. Crear la clase Persona que contiene los atributos nombre, paterno, materno, cedula y edad; crear el metodo que genere el codigo de una persona con las iniciales del paterno, materno, nombre y cedula.

Fernandez Caceres Rocio cedula: 4234567

codigo generado: FCR4234567

```
public class Persona {
    //declaracion de atributos
    private String nombre;
```

```
private String ap_paterno;
private String ap_materno;
private int nro_cedula;
private int edad;

public Persona(String nombre, String ap_paterno, String ap_materno, int nro_cedula) {
    this.nombre = nombre;
    this.ap_paterno = ap_paterno;
    this.ap_materno = ap_materno;
    this.nro_cedula = nro_cedula;
}

public void codigo(){
    String cod="";
    cod=cod+ap_paterno.charAt(0);
    cod=cod+ap_materno.charAt(0);
    cod=cod+nombre.charAt(0);
    String cadena = String.valueOf(nro_cedula);
    cod=cod+cadena;
    System.out.println("codigo:"+cod);
    //Metodos get y set
}
}
```

Ejemplo. Crear una clase que implemente el manejo de caracteres con código Ascii con los métodos:

- a. Contar el número de mayúsculas y minúsculas de una cadena.
- b. Encriptar una cadena utilizando

```
public class ManejoAscii{
public void contar(String cad) {
    int i;    int cma = 0, cmi = 0;
    for (i = 0; i < cad.length(); i++) {
```

```
char ca = cad.charAt(i);
int a = (int) ca;
//System.out.println("ascii"+a+","); //System.out.println("caracter"+(char)a+",");
if (a >= 65 && a <= 90) {
    cma = cma + 1;
}
if (a >= 97 && a <= 122) {
    cmi = cmi + 1;
}
}
System.out.println("Numero mayusculas" + cma);
System.out.println("Numero minusculas" + cmi);
}
public void encriptar(String cad) {
    int i;
    String ncod = "";
    for (i = 0; i < cad.length(); i++) {
        char a = cad.charAt(i);
        int aa = (int) a;
        if (aa >= 65 && aa <= 90) {
            if (aa <= 77) {
                aa = aa + 13;
            } else {
                aa = aa - 13;
            }
        }
        if (aa >= 97 && aa <= 122) {
            if (aa <= 109) {
                aa = aa + 13;
            } else {
                aa = aa - 13;
            }
        }
    }
}
```

```
        char aux = (char) aa;
        ncod = ncod + aux;
    }
    System.out.println("Cadena encriptada: " + ncod);
}

public static void main(String[] args) {
    ManejoString obj = new ManejoAscii();
    Scanner sc = new Scanner(System.in);
    System.out.println("Introduzca una cadena");
    String c = sc.nextLine();
    obj.encriptar(c);
    sc.close();
}
}
```

PARTE PRACTICA

Ejercicios Propuestos

1. Se tiene una cadena, se pide eliminar de la misma todas las vocales, suponer que la cadena contiene solo letras minúsculas.
2. Si se tiene una palabra contar el numero de minúsculas, el numero de mayúsculas y la cantidad de números.
3. Si se tiene dos cadenas con el formato "dd/mm/aaaa" que una representa la fecha de nacimiento y la otra la fecha actual, calcular la edad exacta de la persona en años y meses.
4. Dadas dos palabras, verificar si son anagramas una de otra. Un anagrama está formado con las mismas letras de otra palabra.

cad1 = roma

cad2 = ramo

estas palabras son un anagrama porque están formadas con las mismas letras

5. Crear una función que añada a una cadena su cadena invertida, de tal forma que el resultado sea una cadena palindromo.

Ejemplo cadena: hola, mundo

salida:hola, mundoodnum ,aloh

6. Encriptar un mensaje utilizando el algoritmo inverso

a b c x y z

z y x c b a

7. Ordenar ascendentemente las letras de una cadena

Ejemplo cadena: HOLA

ordenado: AHLO