

A B C D E F G H I J  
K L M N O P Q R S T  
U V W X Y Z  
a b c d e f g h i j  
k l m n o p q r s t  
u v w x y z  
1 2 3 4 5 6 7 8 9 0  
. : ! ? , ; ( ) < >  
@ # \$ % ^ & \* ' " +  
- = [ ] { } \ /

## CARACTERES

# Type char

2

- char : Tipo primitivo que representa un caracter.
  - Cada caracter en una String es almacenado como valor char.
  - Una constante char se encierra en comillas simples tal como: 'a' or '4' or '\n' or '\'
- Los caracteres pueden ser concatenados con cadenas.

```
char initial = 'P';  
System.out.println(initial + " Diddy");  
// P Diddy
```

```
public class Caracteres {  
    public static void main (String [] args){  
  
        String palabra = "Super Cali Fragi Listi Cues Pialidoso";  
        for (int i = 0; i < palabra.length(); i++) {  
            char c = palabra.charAt(i);  
            System.out.println(c);  
        }  
    }  
}
```

# char vs. int

4

- Todos los caracteres tienen un valor entero asignado. Dependiendo de la codificación, puede ser ASCII o UNICODE

- Ejemplos:

'A' es 65,      'B' es 66,      ' ' es 32  
'a' es 97,      'b' es 98,      '\*' es 42

- Mezclar char e int provoca conversión automática a int.

'a' + 10 is 107,      'A' + 'A' is 130

- Para convertir en int un char, se debe hacer un cast.

(char) ('a' + 2) is 'c'

# Modificamos el código

5

```
public class Caracteres {  
    public static void main (String [] args){  
  
        String palabra = "Super Cali Fragi Listi Cues Pialidoso";  
        for (int i = 0; i < palabra.length(); i++) {  
            char c = palabra.charAt(i);  
            int valor = (int)c;  
            System.out.println("[ "+valor+" ] "+c);  
        }  
    }  
}
```

# char vs. String

6

- "h" es String  
'h' es char (Se comportan diferente)

- String es un objeto, tiene métodos.

```
String s = "h";  
s = s.toUpperCase();           // 'H'  
int len = s.length();         // 1  
char first = s.charAt(0);     // 'H'
```

- char es primitivo no tiene métodos.

```
char c = 'h';  
c = c.toUpperCase(); // ERROR:
```

# Comparing char values

7

- You can compare char values with relational operators:

'a' < 'b'    and    'X' == 'X'    and    'Q' != 'q'

```
public class EjemploAbcedario {  
  
    public static void main (String []args){  
        for (char character = 'a'; character<='z'; character++){  
            System.out.print (character);  
        }  
    }  
}
```

```
for (int i = 0; i<500; i++) {  
    char c = (char)i;  
    System.out.println ("[" + i + "] " + c);  
}
```



# Ejercicio



9

- El cifrado Cesar es un sistema de encriptación simple, donde un mensaje es codificado cambiando cada letra por una letra posterior cada N caracteres.
  - Ejemplo (cada 3):
    - $A \rightarrow D$ ,  $H \rightarrow K$ ,  $X \rightarrow A$ , and  $Z \rightarrow C$
- Escribe un programa que reciba un mensaje del usuario y codifique el mensaje.
- Ejemplo:

Your secret message: Brad thinks Angelina  
is cute

Your secret key: 3

The encoded message: eudg wklqnv dqjholqd  
lv fxwh



# Strings answer 1

10

```
// This program reads a message and a secret key from the user  
// and  
// encrypts the message using a Caesar cipher, shifting each  
// letter.
```

```
import java.util.*;
```

```
public class SecretMessage {  
    public static void main(String[] args) {  
        Scanner console = new Scanner(System.in);  
  
        System.out.print("Your secret message: ");  
        String message = console.nextLine();  
        message = message.toLowerCase();  
  
        System.out.print("Your secret key: ");  
        int key = console.nextInt();  
  
        encode(message, key);  
    }  
  
    ...  
}
```

# Strings answer 2

11

```
// This method encodes the given text string using a Caesar
// cipher, shifting each letter by the given number of places.
public static void encode(String text, int shift) {
    System.out.print("The encoded message: ");
    for (int i = 0; i < text.length(); i++) {
        char letter = text.charAt(i);

        // shift only letters (leave other characters alone)
        if (letter >= 'a' && letter <= 'z') {
            letter = (char) (letter + shift);

            // may need to wrap around
            if (letter > 'z') {
                letter = (char) (letter - 26);
            } else if (letter < 'a') {
                letter = (char) (letter + 26);
            }
        }
        System.out.print(letter);
    }
    System.out.println();
}
```