



**Benjamin Buchailot**

### **Trabajo Práctico – Tipo de Datos y Funciones de Cadena**

1- CASTEO: Codifique un programa que solicite el ingreso de un numero decimal y asigne el mismo a una variable **valorDecimal** de tipo double, aplique las operaciones de CASTING para convertir la variable a los siguientes tipos de datos, short, int, long, String, float investigue las diferentes formas de lograr la conversión. Muestre por pantalla el resultado de las conversiones.

2- Si se asigna un valor a una variable fuera de rango (mayor de lo establecido) ¿Qué ocurre? ¿Existe alguna forma de resolverlo? Ejemplifique.

**Si se asigna un valor fuera de rango a una variable al querer ejecutar el programa salta un error impidiendo ejecutar el programa, esto se soluciona cambiando el tipo de variable que vamos a utilizar a una que sea capaz de almacenar el valor que le queremos asignar.**

**Ej: quiero asignar un valor de 1000 a una variable tipo byte, al ejecutar el programa salta error porque las variables tipo byte no llegan a esos valores, por lo que cambio la variable a tipo short, y esta si puede almacenarlo.**

3- Codifique un algoritmo que solicite el ingreso de un numero de 3 dígitos (100 - 999) y por medio del uso de las operaciones matemáticas módulo 10 y división por 10 efectué la suma de los 3 dígitos del número. Ejemplo ingreso 563, salida del algoritmo 14.

4- Desarrolle un programa que ayude a una cajera a determinar el número de billetes y monedas que se necesitan de cada una de las siguientes denominaciones 200, 100, 50, 20, 10, 5, 2 y 1, y monedas de 0.50, 0.25, 0.10 y 0.05 centavos para una cantidad de dinero dada. Ejemplo si la cantidad es 1390,55 se necesitan 6 billetes de 200, 1



billete de 100, 1 billete de 50, 2 billetes de 20, 1 moneda de 0.50 y una moneda de 0.05 centavos.

**Ejercicios con cadenas:**

- 5- Solicite el ingreso de un número y conviértalo a un String mediante `String.valueOf`
- 6- Del siguiente String “La lluvia en Mendoza es escasa” indique cual es el tamaño de la cadena es decir su número de caracteres.
- 7- Solicite el ingreso de una cadena y determine el tamaño de la misma y cuantas vocales tiene en total (recorre el String con `charAt`)
- 8- Reemplaza todas las **a** del String anterior por una **e**. (aplique `replace`)
- 9- Recorre el String del ejercicio 6 y transforma cada carácter a su código ASCII. Muéstralos en línea recta, separados por un espacio entre cada carácter.
- 10- Convertir una frase a mayúsculas o minúsculas, que daremos opción a que el usuario lo pida y mostraremos el resultado por pantalla.
- 11- Pedir dos palabras por teclado, indicar si son iguales. (`equals`, `compareTo`, `compareToIgnoreCase`)
- 12- Dada una cadena, extraer la cuarta y quinta letra usando el método `substring`.
- 13- Pedir el ingreso de dos cadenas por por teclado, indicar si la segunda cadena se encuentra dentro de la primera (usar `indexOf` o `contains`).



### Ejercicios con Tipos Referencia (Objetos)

**14-** Indique cual es la diferencia entre los datos primitivos (por valor) ejemplo el int y su contraparte por referencia ejemplo Integer.

**La diferencia que existe es que las variables de tipo primitivo acceden al valor asignado directamente, mientras que las referenciadas acceden a través de un puntero, almacenan la dirección de memoria no un valor.**

**15-** Indique que sucede si realizo la siguiente declaración de variable: `int numero = null;`  
¿Qué debo modificar para poder asignar null a la variable?

**Al realizar esa declaración saltaría un error ya que no es posible asignarle a una variable de tipo primitivo null, para poder asignar null debo cambiar la variable int a su contraparte Integer, de esta manera si se podría asignarlo.**

**16-** Como transformo una variable tipo Double a una variable tipo double.

```
//creamos la variable tipo Double  
  
Double num = 1.99;  
  
/*luego creamos la variable tipo double  
y le asignamos el valor de la variable  
tipo Double*/  
  
double num2 = (Double) num;
```

**17-** Cree una clase FuncionesPrograma y codifique una función estática que reciba como parámetro una fecha de tipo Date y retorne la fecha como una cadena.

```
public static String getFechaString(Date fecha){ .....  
}
```



Cree una clase Principal que contenga un método main y haga uso de la función `getFechaString`.

- 18-** En la clase `FuncionesPrograma` codifique una función estática que reciba como parámetro 3 valores enteros, día, mes, año y retorne la fecha de tipo `Date` correspondiente.

```
public static Date getFechaDate(int día, int mes, int año){ .....  
}
```

En la clase Principal creada en el punto anterior haga uso de la función `getFechaDate`.

- 19-** Cree una clase `OperacionMatematica` con dos atributos numéricos decimales, `valor1` y `valor2` y un atributo `String` de nombre operación. Cree los correspondientes métodos `get/set`.

Agregue a la clase los siguientes 5 métodos:

```
private double sumarNumeros(){ return  
    valor1 + valor2;  
}  
private double restarNumeros(){ return  
    valor1 - valor2;  
}  
private double multiplicarNumeros(){ return  
    valor1 * valor2;  
}  
private double dividirNumeros(){ return  
    valor1 / valor2;  
}
```

El quinto método será el siguiente:



```
public double aplicarOperacion(String operacion){  
    .....  
}
```

Cree una clase Calculo que contenga un método main, donde cree una instancia de la clase OperacionMatematica, asigne 2 valores para las variables de la instancia y ejecute la función aplicarOperacion, pasando como parámetro primero "+", después "-", a continuación "\*" y finalmente "/". Muestre por pantalla el resultado de las operaciones.

- 20-** Cree una clase Fracción con dos atributos numéricos enteros, numerador y denominador. Agregue un constructor sobrecargado (debe contener como parámetros el numerador y el denominador) que cree el objeto Fracción correspondiente.

Agregue a la clase los siguientes 4 métodos e implemente los mismos:

```
public Fraccion sumarFracciones(Fraccion f1, Fraccion f2){ .....  
}  
public Fraccion restarFracciones(Fraccion f1, Fraccion f2){ .....  
}  
public Fraccion multiplicarFracciones(Fraccion f1, Fraccion f2){ .....  
}  
public Fraccion dividirFracciones(Fraccion f1, Fraccion f2){ .....  
}
```

Cree una clase **OperacionesFraccion** que contenga un método main donde se solicite al usuario el ingreso de 4 valores numéricos enteros con los cuales se crearan 2 objetos Fracción y finalizada la creación de los mismos se ejecutaran los 4 métodos implementados anteriormente asignando el resultado a una nueva variable de tipo Fracción y mostrando por pantalla el resultado de las operaciones realizadas. **Ejercicios con Recursión**



**21-** Codifique un programa que solicite un número entero mayor a cero y que mediante recursión sume todos los números naturales desde el número ingresado hasta 1.

Ejemplo: Ingreso 10

El programa debe sumar  $10 + 9 + 8 + 7 + 6 + 5 + 4 + 3 + 2 + 1$

**22-** Suma los dígitos de un número ingresado por el usuario de forma recursiva.

Ejemplo: el usuario ingresa 1596

El programa debe sumar  $1 + 5 + 9 + 6$

**23-** Crea un programa donde se pida el ingreso de una cadena y por medio de recursión mostrar la cadena de forma inversa. Ejemplo: Ingreso “computadora de escritorio”

Invertir cadena “oirtircse ed arodatupmoc”