

Tipos de datos: 6 Qué ez un dato?

¿Qué es un dato?

Es una porción de información muy concreta y especifica que nos permite, a través de diferentes operaciones y estructuras, tomar decisiones.

Hacen parte de un algoritmo en una estructura más grande

Son la base sobre la cual partimos y sobre la cual llegamos Componen todas las estructuras para que puedan funcionar y sean operables.

Tipos de datos: Texto

Tipo de dato: Texto



Están compuestos por letras, palabras, frases, caracteres, símbolos especiales y espacios.



Char: Es un carácter



String: Es un conjunto de caracteres (cadena).



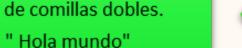
Para realizar la concatenación , según el lenguaje de programación se utilizan los operadores más (+) o amphersand (&): "Hola mundo" + "desde Oaxaca"



Podemos tener 2 cadenas de texto separadas y unirlas mediante la **concatenación**



Cada cadena esta dentro de un par de comillas dobles.



Un string o cadena de texto es la unión de diferentes caracteres que funcionan como eslabones y al unirse (concatenarse) nos da un sentido lógico de valor (cadena o frase)



Para concatenar necesitamos que ambas partes sean del mismo tipo de dato, en este caso, que sean de tipo texto.



Para que nuestra cadena de texto tenga sentido lógico es necesario concatenar un espacio, expresadas en su mayoria por un guión bajo (_)



Al momento de definir tu tipo de dato como texto, todo lo que incluya tu cadena será reconocido como texto, incluso si su representación gráfica es un número: "brenda92@platzi"

Tipos de datos: Námericos

Tipo de dato: númericos

Pueden ser Int y/o Float

Los datos **Int** (Enteros) contiene solo números enteros: 7, 10, 30, etc

Los datos **Float / Double** contiene número con decimales: 6.5 , 8.6, 3.141623 etc.

Los operadores artimeticos nos sirven para hacer operaciones matemáticas con los datos númericos

No es posible concatenar cadenas de diferente tipo.

Para hacer una concatenación de números deberían ser convertidos a texto

Para convertir un número a texto ingresamos el digito en comillas dobles: "22"

Sumar: 19 + 92 = 111 Concatenar: "19" + "92"= 1992 Tipos de datos: Booleanos

Tipo de dato: Booleanos



Este tipo de datos representa una cantidad lógica que es una respuesta a una pregunta de verdadero o falso.



Su representación

True = 1

False = 0

Tipos de datos: Arrays

¿Que es?

Conjunto de elementos del mismo tipo ordenados en fila.

Su primera posición (índex) será siempre 0.

También puedes almacenar números, caracteres y strings

Su tamaño puede variar

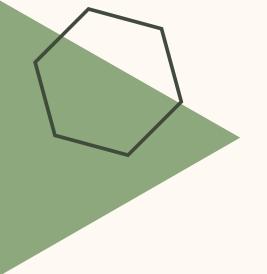
Puedes ordenarlo de la forma que tú lo requieras.

Puedes recorrerlo. Es decir, podemos pasar uno a uno sobre cada una de sus posiciones y operarlas.

Puedes acceder a una posición específica.

Arrays

Caracteristicas



60 né son las estructuras de control?

Estructuras

 Herramientas sobre las cuales se construye y controla el flujo de un algoritmo.
 Permiten tomar decisiones y realizar un

proceso

repetidas

veces.

- Condicionales
- •Permiten
 tomar
 decisiones a
 partir de una
 comparación
 u operación
 •If/else,
 Switch,
 Try/Catch.

Ciclos

- •Realizan
 diferentes
 iteraciones
 de acuerdo a
 condiciones o
 situaciones
 que pide el
 algoritmo
- •For, While, Do-While.

IF/Else

- •Si se cumple determinada condición se ejecuta acción/código anidada en el if, si no es así se ejecutará la anidada en el else.
- Esta es muy usada para validar información.

Switch

- •Nos permite
 evaluar
 múltiples
 casos que
 puede llegar
 a cumplir una
 variable y
 realizar una
 acción en esa
 situación.
- Su estructura consta de un Switch, Case, Break y un Default.

While

•While (ciclo indefinido):
Cuando no sabes las veces
(iteraciones)
que se repetirá un ciclo.

Do While

Do While

(ciclo
indefinido):
Cuando no
sabes las
veces que se
repetirá un
ciclo pero
necesitas que
se realice por
lo menos una
vez.

For

•For (ciclo definido):
Cuando sabes
(o puedes saber) las veces
(iteraciones)
que repetirás el ciclo

Try/Catch

Intenta hacer
 X, pero, si
 falla haz Y.





Condicionales

IF

Switch

Uso

Se usa para tomar decisiones, este evalúa una operación lógica (expresión) que de como resultado True o False.

Selecciona un caso entre multiples alternativas

Estructura

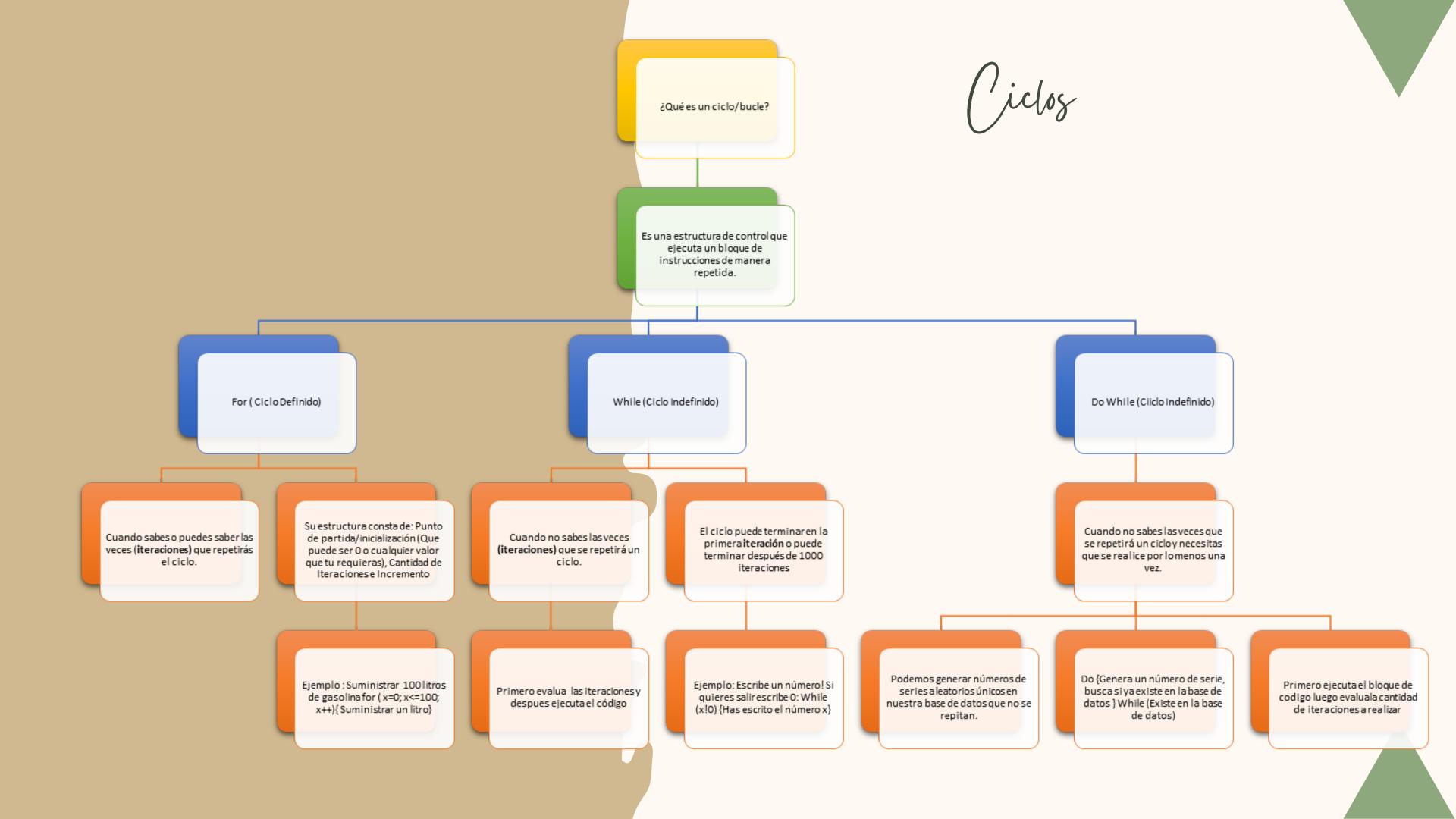
Si se devuelve un True se ejecutara el bloque de instruciones anidadas en el IF, por el contrario, si se devuelve un valor False, se ejecutara el bloque de código anidado en el Else. A veces las instrucciones no requieren que se explique un else en una condición dada.

Su estructura consta de un **Switch**, **Case**, **Break** y un **Default** (
Sirve para que el flujo de nuestro algoritmo no se rompa en
caso de que no se cumpla ningún *case*, mostrando un mensaje
por defecto.)

¿Cuando usarlo?

Si nuestro planteamiento tiene 1 o 2 resultados posibles. Es decir, comprobar True or False

Si nuestro planteamiento tiene más de 2 resultados posibles. Es decir, comparar un valor contra una serie de valores



Try Catch: Funcionamiento

Primero, se ejecuta el código en try {...}.



Si no hubo errores, se ignora catch (err): la ejecución llega al final de try y continúa, omitiendo catch.



Si se produce un error, la ejecución de try se detiene y el control fluye al comienzo de catch (e).



La variable e (podemos usar cualquier nombre para ella) contendrá un objeto de error con detalles sobre lo que sucedió.

Dué ez una función?

Funcion

•Una función es un bloque de instrucciones que hace una actividad especifica de forma independiente al resto del algoritmo.

Objetivo

•Con una función puedes organizar y segmentar partes de tu algoritmo, las cuales puedes reutilizar.

- Parametros
- •Instrucciones de la función

Componente

Resultado

Parametros

•Son los valores que recibe la función como entrada (puede llevarlos o no, pero en la mayoría de casos si lleva).

Instrucciones

•Son las operaciones que hace la función.

•O valor de retorno, es el valor final que entrega la función.

Resultado

Scope p alcance en funciones

Función Pública

El alcance de la función es global

Puedes mandar a llamar las funciones de un algoritmo a otro. Función Privada

El alcance de la función es local.

No pueden ser llamadas en otros algoritmos Recomendaciones

Es importante determinar el alcance de la función.

