

mat|r

Bienvenidos a la Clase N°02

En unos minutos comenzamos...

Clase 02

Introducción a mat|r script

mat|r project

Ecosistema 360 para la creación de experiencias digitales

2020

Expositor



Agustina Dinamarca

Capacitadora mat|r

Se dedica a representar el producto en actividades que tienen como objetivo educar a usuarios sobre qué es mat|r, para qué sirve y cómo usarlo a través del dictado de webinars y capacitaciones, y el desarrollo de contenido educativo.

OBJETIVOS DE LA CLASE

- Aprender qué es un algoritmo y cómo crear uno utilizando mat|r script.
- Dar a conocer los distintos tipos de dato disponibles en mat|r script, cómo definirlos en un algoritmo y cómo operar con y entre ellos.
- Aprender que experiencias y aplicaciones tienen asociados eventos de ciclos de vida que determinan sus funcionamientos.
- Estudiar e interiorizarse sobre qué es un servicio broker, y cuáles ofrece mat|r script.

HOY APRENDERÁS

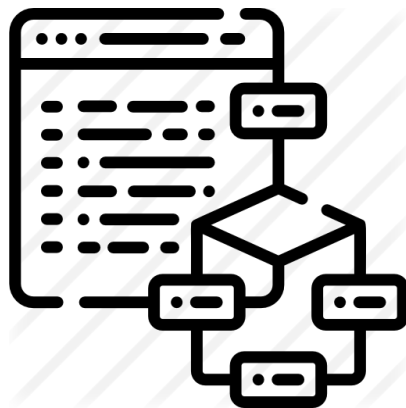
- Definición y construcción de algoritmos.
- Tipos de datos y declaración de variables.
- Operadores.
- Los eventos del ciclo de vida de una experiencia y de una aplicación.
- Qué son los servicios broker, cuáles hay y qué permiten hacer.



ALGORITMOS

¿QUÉ ES UN ALGORITMO?

ES UN CONJUNTO DE INSTRUCCIONES O REGLAS DEFINIDAS Y NO-AMBIGUAS, ORDENADAS Y FINITAS QUE PERMITE, TÍPICAMENTE, SOLUCIONAR UN PROBLEMA, REALIZAR UN CÁMPUTO, PROCESAR DATOS Y LLEVAR A CABO OTRAS TAREAS O ACTIVIDADES.



EJEMPLOS (:

- Manuales de usuario, que muestran algoritmos para usar un aparato.
- Instrucciones que recibe un trabajador de su patrón.
- Algoritmo de multiplicación, para calcular el producto.
- Algoritmo de Euclides, para obtener el máximo común divisor de dos enteros positivos.
- Método de Gauss, para resolver un sistema de ecuaciones lineales.

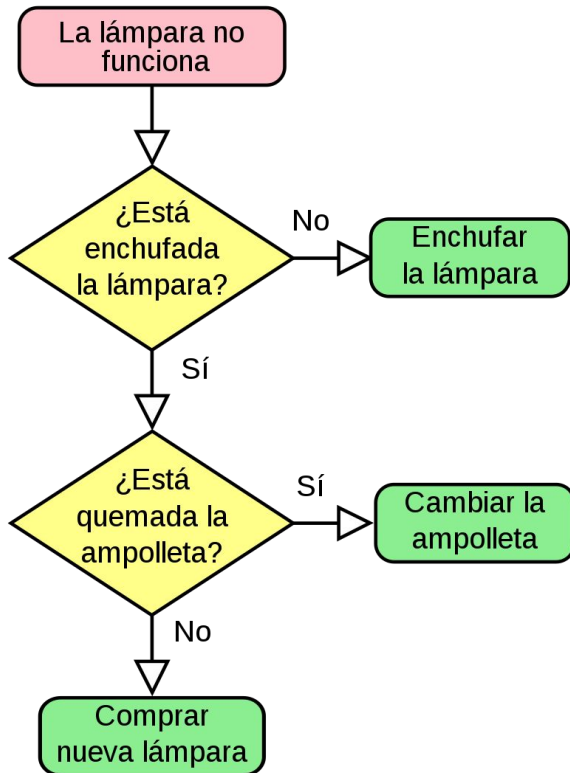
EXPRESIÓN DE UN ALGORITMO

LOS ALGORITMOS PUEDEN SER EXPRESADOS DE MUCHAS MANERAS.

- El pseudocódigo y diagramas de flujo son las formas más estructuradas para representar algoritmos.
- Se mantienen independientes de un lenguaje de programación específico.

EXPRESIÓN DE UN ALGORITMO

Ej. 1



Ej. 2

Si condición Entonces

instrucciones₁;

Si no Entonces

instrucciones₂;

Fin Si

CONSTRUCCIÓN DE ALGORITMOS

EXISTEN BLOQUES DE CÓDIGO QUE SE PUEDEN USAR PARA DEFINIR UN ALGORITMO PARA EJECUTAR EN UNA APLICACIÓN MAT|R

Bloques de código:

- *Application* → *OnInit*, *OnResume* y *OnEnterBackground*.
- *Experience* → *OnCreate*, *OnDestroy* y *OnResume*.
- Contextos de reglas y reglas (*RuleContext* y *Rule*).
- Módulos (*Module*) y funciones.



TIPOS DE DATO

¿QUÉ ES UN TIPO DE DATO?

ES LA PROPIEDAD DE UN VALOR QUE DETERMINA SU DOMINIO, LOS VALORES QUE PUEDE TOMAR, QUÉ OPERACIONES SE LE PUEDEN APLICAR Y CÓMO ES REPRESENTADO INTERNAMENTE POR UNA COMPUTADORA

Integer MediaFile MediaCameraConfigure
String Bool
Date Double

TIPOS DE DATO EN MAT|R SCRIPT

SIMPLES, ESPECIALES, MODELOS Y EXPERIENCIAS

1. **Simples:** no necesitan de un constructor para ser inicializados.
Ej.: String, Integer, Double y Bool.
2. **Especiales:** solo sirven para operar los servicios de los brokers mat|r. No se pueden almacenar en la memoria del dispositivo, en la nube ni ser enviados mediante servicios HTTP.
Ej.: Filter, PageFetcher, DateFormatter, MediaFile, MediaCameraConfigure, MediaGalleryConfigure, Timer.
3. **Modelos y Experiencias:** estructuras básicas de representación y modelado de datos para una aplicación.



DECLARACIÓN DE VARIABLES

¿QUÉ ES DECLARAR UNA VARIABLE Y CÓMO SE HACE?

ES DEFINIR LAS VARIABLES QUE SE USARÁN ESPECIFICANDO SU TIPO Y UN IDENTIFICADOR COMO NOMBRE

Ejemplos:

```
String varName
```

```
Integer age
```

El nombre no puede ser una palabra reservada del lenguaje y no puede ser un nombre ya utilizado en el mismo alcance de ejecución de la variable.



OPERADORES

¿QUÉ SON LOS OPERADORES?

SON SÍMBOLOS QUE INDICAN CÓMO SE DEBEN MANIPULAR LOS OPERANDOS. LOS OPERADORES JUNTO CON LOS OPERANDOS FORMAN UNA EXPRESIÓN, QUE ES UNA FÓRMULA QUE DEFINE EL CÁLCULO DE UN VALOR.

Podemos agruparlos en:

- Aritméticos
- Relacionales
- Lógicos
- Asignación
- Incremento y Decremento
- Concatenación
- Negativo

OPERADORES ARITMÉTICOS

OPERADORES ARITMÉTICOS (Integer o Double)	SÍMBOLO
Suma o adición	+
Resta o sustracción	-
Multiplicación	*
División	/
Resto (entre enteros)	%

OPERADORES RELACIONALES

OPERADORES RELACIONALES	SÍMBOLO	EJEMPLO
Mayor que	>	2 > 4 FALSO 9 > 3 VERDADERO
Mayor o igual que	>=	7 >= 7 VERDADERO 5 >= 5 FALSO
Menor que	<	1 < 8 VERDADERO 7 < 4 FALSO
Menor o igual que	<=	2 <= 2 VERDADERO 3 <= 2 FALSO
Igual	==	6 == 6 VERDADERO 3 == 2 FALSO
Distinto	!=	6 != 7 VERDADERO 3 != 3 FALSO

OPERADORES LÓGICOS

OPERADORES LÓGICOS	SÍMBOLO
AND lógico	&&
OR lógico	
NOT lógico	!

Ejemplos:

- $3 > 2 \ \&\& \ 4 < 5$ VERDADERO
- $3 > 2 \ \&\& \ 4 < 3$ FALSO
- $3 > 2 \ || \ 4 < 3$ VERDADERO
- $6 < 4 \ || \ 7 > 8$ FALSO

OPERADORES ASIGNACIÓN, INCREMENTO Y DECREMENTO

OPERADORES DE ASIGNACIÓN	SÍMBOLO	EJEMPLO
Asignación	=	var1 = 3; var2 = 4; var3 = true; var4 = "hola"

OPERADORES INCREMENTO Y DECREMENTO	SÍMBOLO	EJEMPLO
Incremento (de a 1 unidad)	++	++var1 (si var1 = 3, ++var1 es igual a 4)
Decremento (de a 1 unidad)	--	--var2 (si var2 = 2, --var2 es igual a 1)

OPERADORES CONCATENACIÓN Y NEGATIVO

OPERADORES	SÍMBOLO	EJEMPLO
Concatenación	+	var1 = "mat r "; var2 = "project" var3 = var1 + var2 El valor de var3 es "mat r project"
Negativo	-	-b (si b =3 entonces -b es igual a -3)



EVENTOS DE CICLOS DE VIDA

EVENTOS DE CICLOS DE VIDA

AL DEFINIR UNA EXPERIENCIA Y UNA APLICACIÓN, PODEMOS AGREGAR BLOQUES DE CÓDIGO QUE SE EJECUTARÁN AL DISPARARSE LOS EVENTOS DEL CICLO DE VIDA DE LA MISMA.

Experiencia → onCreate, onResume y onDestroy.

Aplicación → onInit, onResume y onStart.

EVENTOS CICLO DE VIDA DE UNA EXPERIENCIA

```
Experience Perfil {
    onCreate {
        //Al ser creada mediante una sentencia de navegación push o present por
        primera vez.
    }

    onResume {
        // Cada vez que la experiencia se visualiza, es decir, queda en el tope de
        la pila de navegación.
    }

    onDestroy {
        //Previo al aplicar un pop o dismiss sobre la experiencia.
    }
}
```

EVENTOS DEL CICLO DE VIDA DE UNA APLICACIÓN

```
Application {
    OnInit {
        //Se ejecuta al iniciar la aplicación.
    }
    OnResume {
        //Se ejecuta cuando la aplicación sale del estado background.
    }
    OnEnterBackground {
        //Se ejecuta previo a que la aplicación entre al estado de background.
    }
}
```

MODIFICACIÓN RECIENTE

USAR PRINCIPALMENTE CUANDO SE QUIERA IMPLEMENTAR UN CROSSMENU

LOS EVENTOS DE CICLO DE VIDA DE UNA EXPERIENCIA RECIBEN EL DATASOURCE DE LA EXPERIENCIA. ESTO EVITA LLAMAR AL MÉTODO `BROKER.UI.GETDATASOURCE()`.

```
Experience Perfil{  
    String nombre as TextField  
    String apellido as TextField
```

```
    OnResume{  
        Perfil exp =  
broker.ui.getDataSource()  
        exp.nombre = null  
        exp.apellido = null  
    }  
}
```

AHORA



```
Experience Perfil{  
    String nombre as TextField  
    String apellido as TextField
```

```
    OnResume(Perfil exp) {  
        exp.nombre = null  
        exp.apellido = null  
    }  
}
```



SERVICIOS BROKER

¿QUÉ ES UN SERVICIO BROKER?

ES UN SERVICIO A TRAVÉS DEL CUAL SE TIENE ACCESO Y CONTROL DE FUNCIONES NATIVAS DISPONIBLES EN EL DISPOSITIVO.

Se acceden de la siguiente forma:

```
broker.nombreNameSpace
```

Ej. `broker.ui`
`broker.cloudPersistence`
`broker.share`
`etc...`

Ej. de funcionalidades nativas:
acceso y control al GPS, cámara,
galería, etc, del dispositivo.

¿QUÉ SERVICIOS HAY DISPONIBLES?

1. `broker.cloudPersistence.*` funciones sobre almacenamiento de datos en la nube.
2. `broker.cloudPrivatePersistence.*` funciones sobre almacenamiento de datos en la nube, de forma privada.
3. `broker.core.*` funciones del core de mat|r.
4. `broker.localKeychain.*` funciones sobre acceso al almacenamiento local encriptado.
5. `broker.localPersistence.*` funciones sobre almacenamiento local de tipos de datos persistentes en mat|r.
6. `broker.location.*` acceso y control del GPS utilizado en el dispositivo.
7. `broker.media.*` funciones relacionadas con la cámara y la galería de imágenes del dispositivo.
8. `broker.oauth.*` ofrece métodos para autorización utilizando el protocolo OAuth2.
9. `broker.share.*` funciones para compartir texto e imagen.
10. `broker.system.*` acceso a funciones del sistema.
11. `broker.ui.*` funciones de UI y control de navegación.
12. `broker.push.*` funcionalidades mat|r para recibir notificaciones push desde los dispositivos Android e iOS. **NEW! (: (2019)**
13. `broker.async.*` funciones sobre intercambio de mensajes asíncronos entre dispositivos.

EJEMPLO

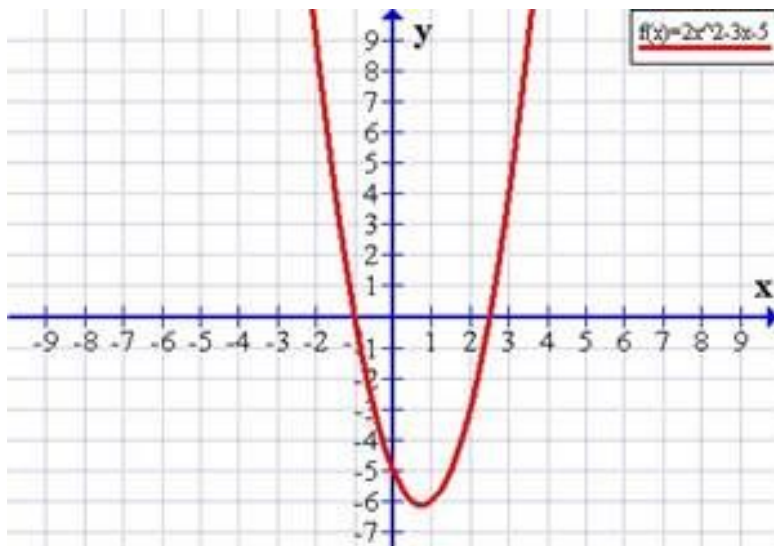
1. Crearemos una aplicación sencilla, una calculadora de raíces de una función cuadrática, con el objetivo de integrar los conceptos aprendidos.

CONTEXTO DEL EJEMPLO

¿Qué es una función cuadrática?

$f(x) = ax^2 + bx + c$, con $a \neq 0$ y a, b, c y x números reales.

Ej.:



Ecuación de la curva (parábola):

$$y = 2x^2 - 3x - 5$$

CONTEXTO DEL EJEMPLO

¿Qué es una raíz de una función?

Se conoce como **raíz** (o **cero**) de una función $f(x)$ a todo elemento x perteneciente al dominio de dicha función tal que se cumpla:

$$f(x) = 0$$

¿Cómo encontrar fácilmente las raíces de una función cuadrática?

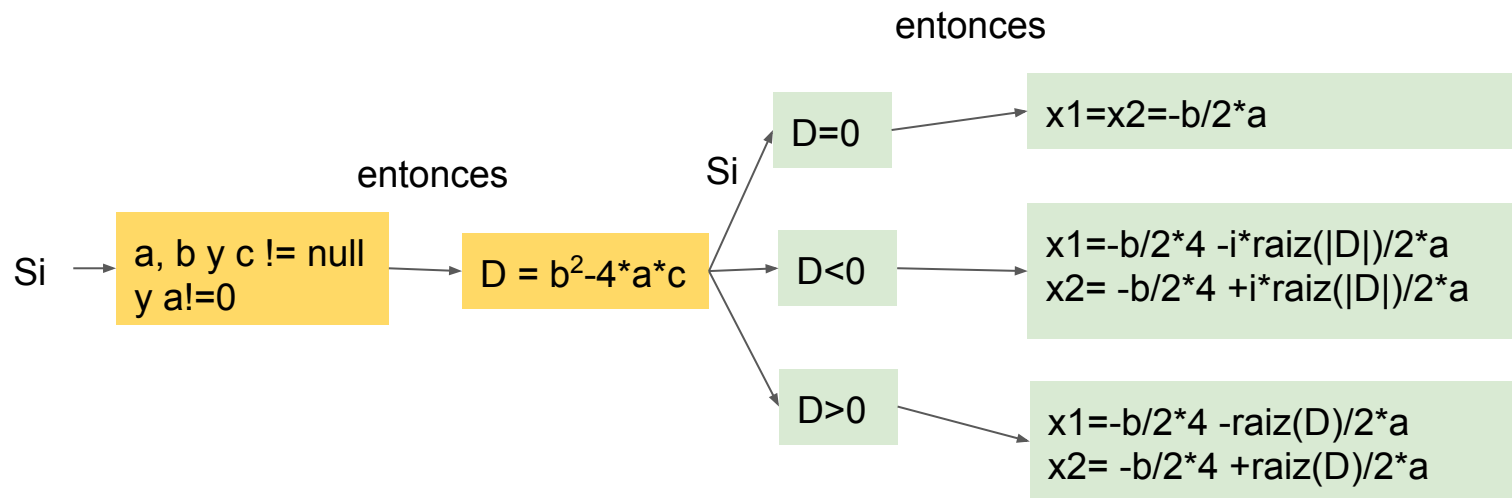
Resolver la ecuación:

$$ax^2 + bx + c = 0, \quad a \neq 0$$

Soluciones:

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad \text{y} \quad x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

EJEMPLO DE DIAGRAMA DE FLUJO





RESUMEN

RESUMEN

- Existen bloques de código que se utilizan para definir un algoritmo para ejecutar en una aplicación mat|r.
- Hay distintos tipos de datos que permiten definir variables en un algoritmo, especificando sus tipos y asignándoles nombres.
- Hay operadores que permiten operar entre tipos de datos y definir el cálculo de un valor.
- Una aplicación y una experiencia presentan eventos que definen sus ciclos de vida. En estos eventos se ejecutan bloques de código que definen ciertas lógicas de funcionamiento para aquellas estructuras.
- mat|r cuenta con una variedad de servicios broker para acceder y controlar funcionalidades del dispositivo.



¿CONSULTAS O
DUDAS?

COMUNÍCATE NOSOTROS



support@matrproject.com



<http://forum.matrproject.com>

mat|r

matrproject.com

—

Contacto

Soledad Peñaloza

Community Manager

soledad.p@matrproject.com

Sergio Farias

Product Development

sergio.f@matrproject.com

Agustina Dinamarca

Capacitadora mat|r

agustina.d@matrproject.com

mat|r

iMuchas Gracias!

mat|r project

Ecosistema 360 para la creación de experiencias digitales