

202111718

# Manual técnico

Walter Javier Santizo  
Mazariegos

walterjav19@gmail.com

---

## programación Orientada a objetos

La programación orientada a objetos se basa en el concepto de crear un modelo del problema de destino en sus programas. La programación orientada a objetos disminuye los errores y promociona la reutilización del código. Python es un lenguaje orientado a objetos. Los objetos definidos en Python tienen las características siguientes

### Clases utilizadas

- Análisis Léxico
- Análisis Sintactico
- Archivo\_html
- Token
- Controles
- Propiedades
- Identificador
- Primitivos
- Instrucción

Se utilizaron estas clases con sus atributos valiéndose de las propiedades para crear un leer con este paradigma de programación

### Modelado de los Menús

```
from tkinter import *  
from tkinter import ttk  
import tkinter  
from tkinter import messagebox  
from tkinter.font import Font
```

Importamos la clase Tkinter que nos permite usar interfaces graficas y contenedores que facilitan la interacción del usuario

Se crearon 3 ventanas por medio de funciones las cuales contaron con la funcionalidad de regresar que guarda la ventana con la función withdraw y manda a llamar los componentes con deiconify

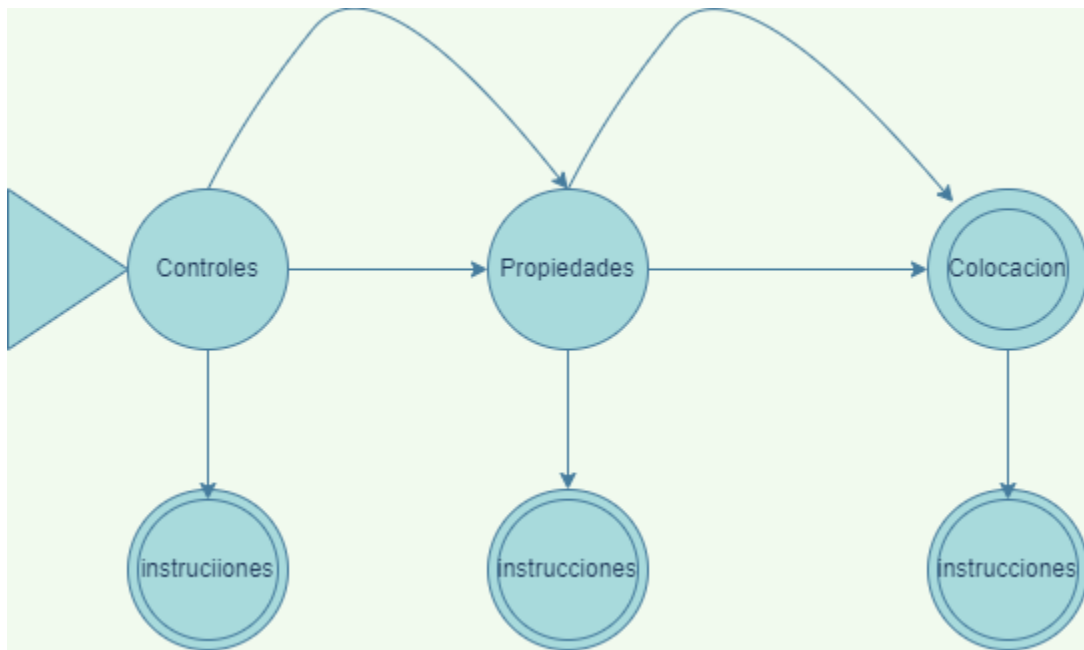
```
principal.deiconify()  
ruta.withdraw()
```

Se usaron componentes como label, treeview, Entry, button etc..

## Tokens en Python

Tabla de Tokens	
Token	expresión
RCONTROLES	Controles
RPROPIEDADES	Propiedades
RCOLOCACION	Colocación
RETIQUETA	Etiqueta
RBOTON	Boton
RCHECK	Check
RRADIOBOTON	RadioBoton
RTEXTO	Texto
RAREATEXTO	AreaTexto
RCLAVE	Clave
RCONTENEDOR	Contenedor
t_EXCLAMACION	!
t_LLAA	<
t_LLAC	>
t_GUION	-
t_PTCOMA	;
t_PUNTO	.
t_PARA	(
t_PARC	)
t_COMA	,
ID	[a-zA-Z_][a-zA-Z_0-9]*
Comentario simple	\W.*\n
Comentario Multilinea	\W*(.\n)*?\n
Salto de linea	\n+

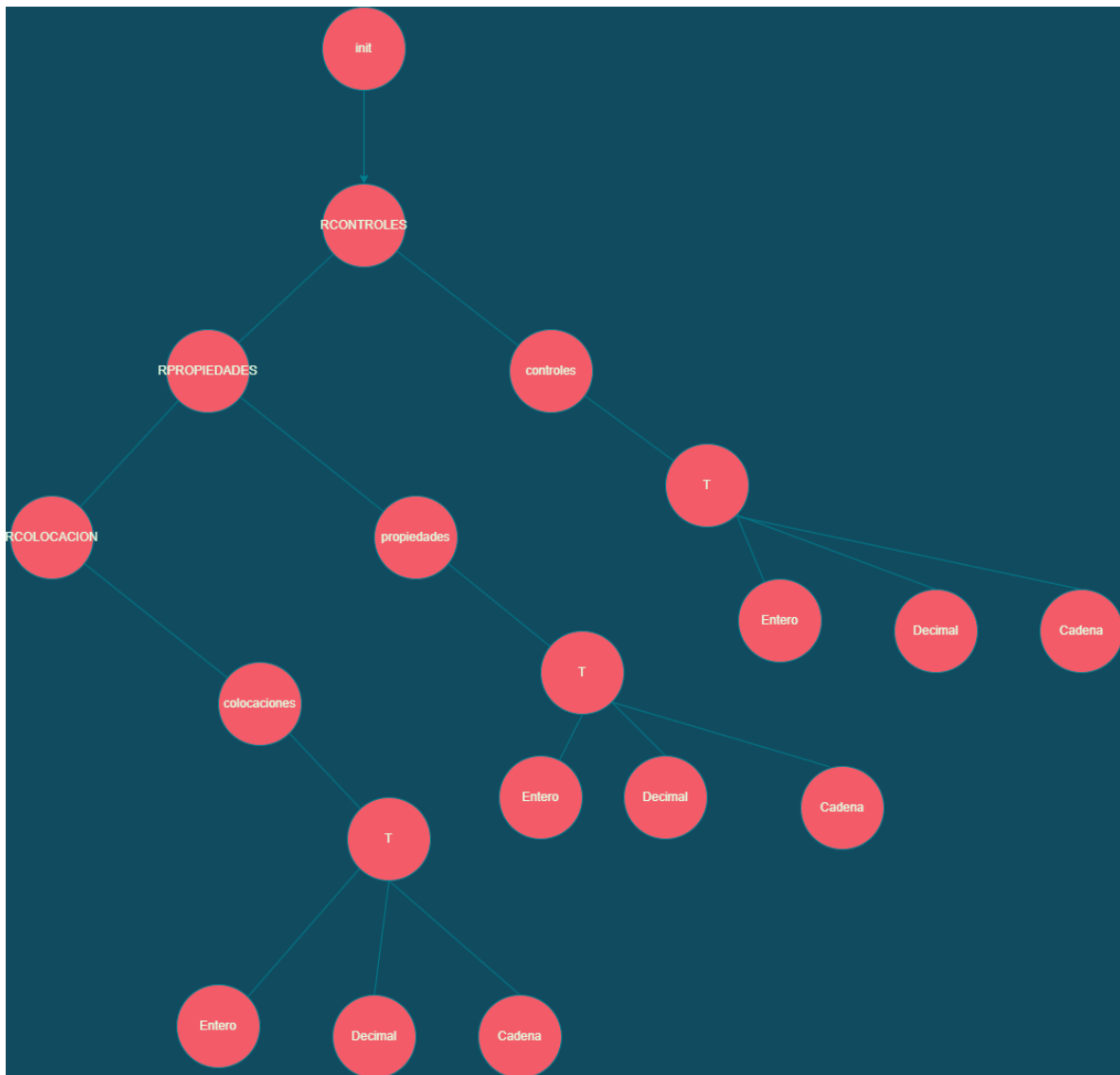
## AFD



$\Sigma = \{RCONTROLES, RPROPIEDADES, RCOLOCACION,$   
 $RETIQUETA, RBOTON, RCHECK, RRADIOBUTTON,$   
 $RTEXT, RAREATEXTO, RCLAVE, RCONTENEDOR$   
 $\}$

$\Sigma = \{t\_EXCLAMACION, t\_LLAA, t\_LLAC, t\_GUION, t\_PTCOMA,$   
 $t\_PUNTO, t\_PARA, t\_PARC, t\_COMA\}$

## método del árbol



## Gramática Libre de Contexto

=====

## GRAMATICA

SIMBOLO INICIAL => `init`

TERMINALES => ENTERO, DECIMAL, CADENA, ID

NO TERMINALES => LLAA EXCLAMACION GUION GUION RCONTROLES controles

RCONTROLES GUION GUION LLAC

| LLAA EXCLAMACION GUION GUION RPROPIEDADES propiedades RPROPIEDADES GUION

GUION LLAC | LLAA EXCLAMACION GUION GUION RCOLOCACION colocaciones

RCOLOCACION GUION GUION LLAC

PRODUCCIONES =>

`init` : instrucciones

instrucciones : instrucciones instruccion

instrucciones : instruccion

instruccion : LLAA EXCLAMACION GUION GUION RCONTROLES controles RCONTROLES  
GUION GUION LLAC

| LLAA EXCLAMACION GUION GUION RPROPIEDADES propiedades RPROPIEDADES GUION

GUION LLAC | LLAA EXCLAMACION GUION GUION RCOLOCACION colocaciones

RCOLOCACION GUION GUION LLAC

controles : controles control

controles : control

control : RETIQUETA ID PTCOMA

| RBOTON ID PTCOMA

| RCHECK ID PTCOMA

| RRADIOBUTTON ID PTCOMA

| RTEXTOT ID PTCOMA

| RAREATEXTO ID PTCOMA

| RCLAVE ID PTCOMA

| RCONTENEDOR ID PTCOMA

propiedades : propiedades propiedad

propiedades : propiedad

propiedad : ID PUNTO ID PARA parametros PARC PTCOMA

parametros : parametros COMA parametro

```
parametros : parametro
```

```
parametro : expresion
```

```
expresion : ENTERO
```

```
          | DECIMAL
```

```
          | CADENA
```

```
          | ID
```