

Trabajo: El Zodiaco

Alumno: Puma Huanca Anthony Rusbel
Docente: Ing. Coyla Idme Leonel
Lenguajes de programación II – FINESI
Universidad Nacional del Altiplano
Facultad de Ingeniería Estadística e Informática

Codigo en Python del Zodiaco sin Tkinter

```
1 class Zodiaco:
2     def __init__(self, dia, mes):
3         self.dia = dia
4         self.mes = mes
5
6     def obtener_signo(self):
7         if self.mes < 1 or self.mes > 12:
8             return "Fecha inv lida"
9
10        if (self.mes == 3 and 21 <= self.dia <= 31) or (self.mes ==
11            4 and 1 <= self.dia <= 19):
12            return "Aries"
13        elif (self.mes == 4 and 20 <= self.dia <= 30) or (self.mes
14            == 5 and 1 <= self.dia <= 20):
15            return "Tauro"
16        elif (self.mes == 5 and 21 <= self.dia <= 31) or (self.mes
17            == 6 and 1 <= self.dia <= 20):
18            return "G minis"
19        elif (self.mes == 6 and 21 <= self.dia <= 30) or (self.mes
20            == 7 and 1 <= self.dia <= 22):
21            return "C ncer"
22        elif (self.mes == 7 and 23 <= self.dia <= 31) or (self.mes
23            == 8 and 1 <= self.dia <= 22):
24            return "Leo"
25        elif (self.mes == 8 and 23 <= self.dia <= 31) or (self.mes
26            == 9 and 1 <= self.dia <= 22):
27            return "Virgo"
28        elif (self.mes == 9 and 23 <= self.dia <= 30) or (self.mes
29            == 10 and 1 <= self.dia <= 22):
30            return "Libra"
```

```

24         elif (self.mes == 10 and 23 <= self.dia <= 31) or (self.mes
25             == 11 and 1 <= self.dia <= 21):
26             return "Escorpio"
27         elif (self.mes == 11 and 22 <= self.dia <= 30) or (self.mes
28             == 12 and 1 <= self.dia <= 21):
29             return "Sagitario"
30         elif (self.mes == 12 and 22 <= self.dia <= 31) or (self.mes
31             == 1 and 1 <= self.dia <= 19):
32             return "Capricornio"
33         elif (self.mes == 1 and 20 <= self.dia <= 31) or (self.mes
34             == 2 and 1 <= self.dia <= 18):
35             return "Acuario"
36         elif (self.mes == 2 and 19 <= self.dia <= 29) or (self.mes
37             == 3 and 1 <= self.dia <= 20):
38             return "Piscis"
39         else:
40             return "Fecha inv lida"
41
42     def main():
43         dia = int(input("Ingresa tu d a de nacimiento: "))
44         mes = int(input("Ingresa tu mes de nacimiento (en n mero): "))
45         persona = Zodiaco(dia, mes)
46         print("Tu signo zodiacal es:", persona.obtener_signo())
47
48     if __name__ == "__main__":
49         main()

```

Listing 1: Código Ejemplo 1

Codigo en Python del Zodiaco con Tkinter

```

1  import tkinter as tk
2  from tkinter import messagebox
3
4  class Zodiaco:
5      """Clase que determina el signo zodiacal bas ndose en el d a y
6          mes."""
7      def __init__(self, dia, mes):
8          self.dia = dia
9          self.mes = mes
10
11     def obtener_signo(self):
12         # Primero, validar la fecha de forma simple (sin considerar
13             d as espec ficos por mes,

```

```

12     # ya que la lógica de tu código original maneja la
13     # mayor a de los límites)
14     if self.mes < 1 or self.mes > 12 or self.dia < 1 or self.dia
15     > 31:
16         return "Fecha inv lida"
17
18     # Lógica de signos (copiada del código original)
19     if (self.mes == 3 and 21 <= self.dia <= 31) or (self.mes ==
20     4 and 1 <= self.dia <= 19):
21         return "Aries"
22     elif (self.mes == 4 and 20 <= self.dia <= 30) or (self.mes
23     == 5 and 1 <= self.dia <= 20):
24         return "Tauro"
25     elif (self.mes == 5 and 21 <= self.dia <= 31) or (self.mes
26     == 6 and 1 <= self.dia <= 20):
27         return "Geminis"
28     elif (self.mes == 6 and 21 <= self.dia <= 30) or (self.mes
29     == 7 and 1 <= self.dia <= 22):
30         return "Cncer"
31     elif (self.mes == 7 and 23 <= self.dia <= 31) or (self.mes
32     == 8 and 1 <= self.dia <= 22):
33         return "Leo"
34     elif (self.mes == 8 and 23 <= self.dia <= 31) or (self.mes
35     == 9 and 1 <= self.dia <= 22):
36         return "Virgo"
37     elif (self.mes == 9 and 23 <= self.dia <= 30) or (self.mes
38     == 10 and 1 <= self.dia <= 22):
39         return "Libra"
40     elif (self.mes == 10 and 23 <= self.dia <= 31) or (self.mes
41     == 11 and 1 <= self.dia <= 21):
42         return "Escorpio"
43     elif (self.mes == 11 and 22 <= self.dia <= 30) or (self.mes
44     == 12 and 1 <= self.dia <= 21):
45         return "Sagitario"
46     elif (self.mes == 12 and 22 <= self.dia <= 31) or (self.mes
47     == 1 and 1 <= self.dia <= 19):
48         return "Capricornio"
49     elif (self.mes == 1 and 20 <= self.dia <= 31) or (self.mes
50     == 2 and 1 <= self.dia <= 18):
51         return "Acuario"
52     elif (self.mes == 2 and 19 <= self.dia <= 29) or (self.mes
53     == 3 and 1 <= self.dia <= 20):
54         return "Piscis"
55     else:
56         return "Fecha inv lida (revisar día para el mes)"
57
58 class AppZodiaco(tk.Tk):
59     def __init__(self):

```

```

46     super().__init__()
47     self.title("Calculadora de Signo Zodiacal")
48     self.geometry("350x250")
49     self.config(padx=10, pady=10)
50
51     # Variables de control para almacenar los datos de entrada y
52     # salida
53     self.dia_var = tk.StringVar()
54     self.mes_var = tk.StringVar()
55     self.resultado_var = tk.StringVar(value="Ingresa tu fecha de
56     nacimiento")
57
58     self.crear_widgets()
59
60     def crear_widgets(self):
61         # T tulo
62         tk.Label(self, text=" Cul es tu Signo Zodiacal?", font=(
63             'Arial', 14, 'bold')).pack(pady=10)
64
65         # Contenedor para la entrada de datos
66         input_frame = tk.Frame(self)
67         input_frame.pack(pady=5)
68
69         # Entrada del D a
70         tk.Label(input_frame, text="D a :").grid(row=0, column=0,
71             padx=5, sticky='w')
72         tk.Entry(input_frame, textvariable=self.dia_var, width=10,
73             justify='center').grid(row=0, column=1, padx=5)
74
75         # Entrada del Mes
76         tk.Label(input_frame, text="Mes (1-12):").grid(row=1, column
77             =0, padx=5, sticky='w')
78         tk.Entry(input_frame, textvariable=self.mes_var, width=10,
79             justify='center').grid(row=1, column=1, padx=5)
80
81         # Bot n de C lculo
82         tk.Button(self, text="Calcular Signo", command=self.
83             calcular_signo_gui, bg='#AEC6E3', font=('Arial', 10, '
84             bold')).pack(pady=15)
85
86         # Resultado
87         tk.Label(self, textvariable=self.resultado_var, font=('Arial
88             ', 12), fg='darkblue').pack(pady=10)
89
90     def calcular_signo_gui(self):
91         """Funci n que maneja la l gica de la GUI al presionar el
92         bot n."""
93         try:

```

```

83     # 1. Obtener y validar las entradas
84     dia = int(self.dia_var.get())
85     mes = int(self.mes_var.get())
86
87     if not (1 <= mes <= 12 and 1 <= dia <= 31):
88         messagebox.showerror("Error de Entrada", "Por favor,
            ingresa un d a (1-31) y un mes v lido (1-12).")
89         )
90         self.resultado_var.set("Fecha inv lida")
91         return
92
93     # 2. Crear instancia y realizar el c lculo
94     persona = Zodiaco(dia, mes)
95     signo = persona.obtener_signo()
96
97     # 3. Mostrar el resultado en la GUI
98     self.resultado_var.set(f"Tu signo es: {signo}")
99
100     except ValueError:
101         # Manejar el caso donde el usuario no ingresa n meros
102         messagebox.showerror("Error de Entrada", "Solo se
            permiten n meros enteros para el d a y el mes.")
103         self.resultado_var.set("Error de entrada")
104     except Exception as e:
105         # Manejar cualquier otro error inesperado
106         messagebox.showerror("Error", f"Ocurri un error
            inesperado: {e}")
107
108 if __name__ == "__main__":
109     app = AppZodiaco()
110     app.mainloop()

```

Listing 2: Código Ejemplo 2