



Clase 8 – Mini práctica integradora: for, color y giro

Módulo: Repeticiones y Turtle

Fundamentación

Después de trabajar con for, listas, colores y while, es importante que los estudiantes integren los conceptos en un proyecto breve pero completo. En esta clase se fomenta la elección personal del diseño para fortalecer la creatividad y la autonomía, manteniendo el uso de estructuras vistas hasta ahora.

Objetivo general

- Integrar y aplicar los conceptos de ciclos (for, while), listas y manipulación de colores para crear una figura visual con Turtle.

Objetivos particulares

1. Planificar un patrón visual antes de codificarlo.
2. Usar listas de colores y ciclos para generar secuencias visuales.
3. Ajustar giros y repeticiones para obtener el diseño deseado.
4. Trabajar de forma autónoma o en equipo en un mini-proyecto.

Contenidos

- **Conceptuales**
 - Revisión de for y while.
 - Uso de listas para datos repetitivos.
 - Parámetros de Turtle (color, pensize, speed).
- **Procedimentales**
 - Planificar y escribir un código que combine los conceptos.
 - Experimentar con figuras y parámetros hasta lograr un resultado estético.



PYTHON INICIAL

- **Actitudinales**

- Autonomía en la elección del diseño.
- Trabajo colaborativo y creatividad.

Desarrollo de la clase

Apertura

- Recordatorio rápido de:
 - Cómo se crean listas de colores.
 - Cómo recorrer listas con for.
 - Uso de while para animaciones.
- Tres ejemplos posibles:
 - **Flor** (círculos rotados).
 - **Estrella** (giro fijo con forward largo).
 - **Mosaico** (patrón repetitivo con cambios de color).

Ejemplos guiados

Flor:

```
import turtle as t
```

```
t.speed(0)
colores = ["red", "blue", "green", "orange", "purple", "pink"]
```

```
for color in colores:
    t.color(color)
    t.circle(50)
    t.right(60)
```

```
t.done()
```

Estrella:

```
import turtle as t
```

```
t.speed(0)
t.color("gold")
for i in range(36):
    for j in range(5):
        t.forward(100)
        t.right(144)
    t.right(10)
```



PYTHON INICIAL

```
t.done()
```

Mosaico simple:

```
import turtle as t

t.speed(0)
colores = ["red", "blue", "green", "yellow"]
for x in range(6):
    for color in colores:
        t.color(color)
        t.forward(50)
        t.right(90)
    t.forward(60)

t.done()
```

Diseño guiado de mosaico con Turtle

Explicación breve antes de programar

- Un mosaico es un **patrón repetitivo** formado por figuras simples (círculos, cuadrados, estrellas, etc.).
- Los ciclos for y while nos permiten **repetir y mover** figuras para llenar la pantalla sin dibujar cada una manualmente.
- Usaremos un **patrón base** y luego lo personalizaremos.

Plantilla editable (figura libre)

```
import turtle

# --- Configuración inicial ---

turtle.speed(0)          # Velocidad máxima
turtle.bgcolor("black")  # Fondo negro
turtle.pencolor("cyan")  # Color del lápiz

# --- Parámetros editables ---

filas = 5      # Cantidad de filas
columnas = 5   # Cantidad de columnas
tamano = 40    # Tamaño base de la figura
espaciado = 60 # Distancia entre figuras
```



PYTHON INICIAL

```
# --- Dibujo del mosaico ---  
for fila in range(filas):      # Recorre las filas  
    for columna in range(columnas): # Recorre las columnas  
  
        # Levantar lápiz y mover a la posición  
        turtle.penup()  
        turtle.goto(columna * espaciado, fila * espaciado)  
        turtle.pendown()  
  
    # --- Figura base editable ---  
    # Ejemplo: estrella  
    for i in range(5):  
        turtle.forward(tamano)  
        turtle.right(144)  
  
# --- Mantener la ventana abierta ---  
turtle.done()
```

Explicación del código

1. Ciclo anidado

- El for fila se encarga de bajar a la siguiente línea del mosaico.
- El for columna coloca las figuras una al lado de otra en la misma fila.

2. Posicionamiento con goto

- `turtle.goto(x, y)` mueve el lápiz a una coordenada de la pantalla.
- Multiplicamos columna y fila por el espaciado para que no se encimen las figuras.

3. Figura editable

- Dentro del segundo for (el de columnas) está el bloque que dibuja la figura.
- Cambiarlo por un cuadrado:



PYTHON INICIAL

```
for i in range(4):  
    turtle.forward(tamano)  
    turtle.right(90)
```

Cambiarlo por un círculo:

```
turtle.circle(tamano)
```

Creatividad

- Cambiar colores con `turtle.pencolor("red")` o dentro del ciclo para que cada figura sea distinta.
- Modificar filas, columnas, espaciado y tamaño para experimentar.