**Versión 0.2:**

**1. Características de esta versión**

**- Comandos TikZ disponibles:** \draw, \fill, \filldraw, \tikzset, \definecolor, \newcommand, \animarPytikz.

**- Figuras disponibles:** rectangle, arc, circle, controls.

**- Colores disponibles:** red, blue, green, cyan, black, yellow, white, color\_1! porcentaje! color\_2 (Ej: cyan!75!green, tendra un 75% de cyan y un 25% de green)).

**- Tipo de línea disponibles:** Dotted.

**- Estilos disponibles:**

**- Por valor:** Color, tipo de línea.

**- Por {clave=valor}:**line width, fill, draw, top color, bottom color.

**- Unidades métricas disponibles:** pt, cm.

**2. Función por comandos:**

**- \draw, \filldraw, \fill:** La característica de cada comando es la siguiente:

\draw: Dibuja la figura con un borde de color negro por predeterminado.

\filldraw: Dibuja la figura con un borde de color negro y un relleno negro por predeterminado.

\fill: Dibuja la figura con un relleno de color negro por predeterminado.

**Estructura del comando \draw:** (La estructura es igual para los comandos anteriormente mencionados)

\draw[parámetros] (posicion\_x,posicion\_y) figura(opcional) (ancho,alto);

**Parámetros:** Son los estilos que se aplicara a la figura, si no hay figura (--) son los estilos que se aplicara a las lineas.

- Por valor, ej: [color, tipo\_de\_linea]

- Por clave=valor, ej: [fill=color\_relleno, line width=numero, draw=color\_borde]

**Posicion\_x, Posicion\_y:** Son las coordenadas desde donde, hasta donde se dibujará la figura, la cantidad de coordenadas varia según la figura. Deben tener un valor INTEGER o FLOAT, también soporta un valor con unidad métrica.

**Figuras:**

- **Rectangle:** **(Dibuja un rectángulo)**

**Ejemplo de uso**

\draw[red, line width= 3pt] (100,100) rectangle (200,200);

- **Arc: (Dibuja un arco)**

**Ejemplo de uso**

\draw[draw=red, line width= 5pt] (10,10) arc (0:90:3cm);

\draw[draw=red, line width= 5pt] (10,10) arc[start angle=0, end angle=90, radius=3cm];

- **Circle: (Dibuja un círculo)**

**Ejemplo de uso**

\draw[draw=red,fill=yellow] (10,10) circle (3cm);

- **Controls: (Dibuja un Bézier curvas libres)**

**Ejemplo de uso**

\draw[draw=red, line width= 5pt] (100,100) .. controls (100,200) and (200,300) .. (300,400) .. (200,100) .. (300,200);

**Combinación de figuras:**

* **Arc con -- (líneas):**

**Ejemplo de uso**

\draw (100,100) -- (90:3cm) arc (90:360:3cm) arc (0:0:3cm) -- cycle;

* **Controls con más Controls**:

**Ejemplo de uso**

\draw (100,100) .. controls (100,125) and (150,200) .. (200,200) .. controls (200,175) and (250,100) .. (300,100);

**Si no hay figura:** **(El tipo de dibujo es con líneas)**

**Ejemplo de uso**

\draw[line width= 1.5pt, fill = yellow, draw = black] (100,100) -- (200,50) -- (50,300) -- (40,200) -- cycle;

**- \definecolor:** La función de este comando consiste en guardar códigos de color definidos por el usuario, la variable estará disponible de forma global.

**Construcción de la variable:**

\definecolor{nombre\_variable }{modelo\_de\_color}{código\_de\_color}

**Ejemplo de uso:**

\definecolor{ColorA}{RGB}{255,0,0}

Donde “ColorA” es el nombre de la variable, “RGB” es el modelo de color, y “255,0,0” es el Código RGB.

**Uso de la variable:**

\comando\_de\_dibujo[parametro\_de\_estilo=nombre\_variable]

**Ejemplo de uso:**

\draw[fill=ColorA,line width=5pt] (500,100) arc (0:90:3cm);

Donde \draw es el comando de dibujo, donde "fill" es el parametro de estilo a utilizar y el valor de la variable “ColorA” sería el valor de ese estilo. (En este caso utilizamos la misma variable que creamos anteriormente en modo de ejemplo llamado " ColorA ").

**- \tikzset:** La función de este comando consiste en guardar estilos definidos por el usuario, disponible tanto de forma global como de forma local en una variable (Actualmente solo disponible de forma global), también además de guardar variable con estilos ya definidos, está la opción de cambiar el valor de esos estilos de las variables cuando se está invocando en una figura, de forma paramétrica.

**Construcción de la variable sin parámetros a definir:**

\tikzset{nombre\_variable global/.style={estilo=valor}}

**Ejemplo de uso:**

\tikzset{estilo global/.style = {draw=red,fill=yellow,line width=5pt}}.

Donde "estilo" es el nombre de la variable, "global" es el entorno en que estará disponible esta variable, "draw" es el estilo que se aplicara, "red" es el valor de ese estilo.

**Uso de la variable sin parámetros a definir:**

\comando\_de\_dibujo[nombre\_variable entorno\_de\_variable]

**Ejemplo de uso:**

\draw[estilo global] (100,100) rectangle (250,250);

Donde "\draw" es el comando de dibujo, donde "estilo" es el nombre de la variable, "global" es el entorno de variables a utilizar en este caso utilizaremos la variable "estilo" que se encuentra en el entorno "global".

**Construcción de la variable con parámetros a definir:**

\tikzset{nombre\_variable global/.style n args = {cantidad\_de\_parametros}{estilo=valor, estilo=#n\_arg} nombre\_variable global/.default = {valor\_arg}

**Ejemplo de uso:**

\tikzset{estilo global/.style n args = {2}{line width=#2, draw = #1}, estilo global/.default = {cyan}{5pt}}

Donde "estilo " es el nombre de la variable, "global" es el entorno en que estará disponible esta variable, "2" es la cantidad de parámetros que el usuario puede cambiar en la variable, "#1" y "#2" se hace referencia a los valores que el usuario puede cambiar en la variable, "cyan" y "5pt" son los valores que ocupan por defecto en los parámetro "#1" y "#2".

**Uso de la variable con parámetros a definir:**

\comando\_de\_dibujo[nombre\_variable entorno\_de\_variable = {valor\_parametro}]

**Ejemplo de uso:**

\draw[estilo global = {red}{1cm}] (100,100) rectangle (250, 250);

Donde \draw es el comando de dibujo, donde "estilo" es el nombre de la variable, "global" es el entorno de variables a utilizar en este caso utilizaremos la variable "estilo" que se encuentra en el entorno "global", donde "red" y "1cm" es el valor del parámetro que se encuentra en la variable (En este caso utilizamos la misma variable que creamos anteriormente en modo de ejemplo llamado "estilo ").

**- \newcommand:** La función de este comando consiste en guardar un conjunto de comandos definidos por el usuario, la variable estará disponible de forma global.

**Construcción de la variable:**

\newcommand{nombre\_variable }[cantidad\_de\_parametros\_a\_definir][1]

**Ejemplo de uso:**

\newcommand{circulo}[2][1]{

\draw[fill=#1] (215,115) circle (#2);

}

Donde “circulo” es el nombre de la variable, “2” es la cantidad de parametros a definir, y “\draw[fill=#1] (215,115) circle (#2);”, es el comando que se ejecutara al ser invocado variable “circulo”.

**Uso de la variable:**

\nombre\_variable[parametro\_a\_definir]{…parametro\_a\_definir}

**Nota:** Este último “…parámetro\_a\_definir” es opcional, se invoca de esa manera las veces que desees, siempre y cuando la “cantidad\_de\_parametros\_a\_definir“ sea mayor a 1, si es solo 1 parámetro solo se utiliza [parametro\_a\_definir].

**Ejemplo de uso cuando es solo 1 parametro:**

**Construcción de la variable:**

\newcommand{oabajocentro}[1][1]{

\draw[fill=#1] (215,115) circle (30pt);

}

**Uso de la variable:**

\oabajocentro[yellow]

**Ejemplo de uso cuando es más de 1 parametro:**

**Construcción de la variable:**

\newcommand{circulo}[2][1]{

\draw[fill=#1] (215,115) circle (#2);

}

**Uso de la variable:**

\oabajocentro[yellow]{30pt}

Donde \oabajocentro es el nombre de la variable del comando, “[yellow]” es el primer valor a definir identificado en la construcción de la variable como “#1”, y “30pt” en el “#2”. De esa forma es como se invoca el comando y ejecuta la instrucción “\draw”, que se encuentra almacenada en su interior.

**- \animarPytikz:** La función de este comando consiste en guardar un conjunto de comandos invocables creados por el usuario, y generar un GIF a partir de ese conjunto de comandos en orden de invocación.

**Construcción de la variable:**

\animarPytikz{

\nombre\_variable

}

**Ejemplo de uso:**

\animarPytikz{

\oizqarriba[yellow]

\ocentro[yellow]

\oabajocentro[yellow]

}

Donde el conjunto de comandos a ejecutar en orden de arriba hacia abajo será, “\oizqarriba” luego “\ocentro” y por último “\oabajocentro” donde cuyo GIF generado empezara con el primero de abajo hacia arriba, hasta el último “\oabajocentro”.

**3. Función por estilo:**

**Estilos por valor:**

**-** **color, tipo de línea:** La característica de cada estilo es la siguiente:

**Color:** El estilo que se aplica al añadir un color, es el de cambiar el color de los bordes de la figura.

**Ejemplo de uso:**

\draw[red] (100,100) rectangle (200, 200);

**Tipo de línea:** El estilo que se aplica al añadir un tipo de línea, es el de cambiar el tipo de línea de la figura.

**Ejemplo de uso:**

\draw[dotted] (100,100) rectangle (200, 200);

**Estilos {Clave=valor}:**

**-** **line width, fill, draw, top color, bottom color:** La característica de cada estilo es la siguiente:

**Line width:** El estilo que aplica a las figuras es la de aumentar el grosor de los bordes de la figura (Si tiene un tipo\_de\_linea diferente a solid, no se aplicara).

**Ejemplo de uso:**

\draw[line width = 1cm] (100,100) rectangle (200, 200);

**Fill:** El estilo que aplica a la figura es la de rellenar su interior, con un color.

**Ejemplo de uso:**

\draw[fill = red] (100,100) rectangle (200, 200);

**Draw:** El estilo que aplica a la figura es la de aplicar un color a los bordes de la figura.

**Ejemplo de uso:**

\draw[draw = red] (100,100) rectangle (200, 200);

**Estilos de degradado:** Se pueden aplicar varios estilos de degradado en una figura para combinar sus efectos.

**Ejemplo de uso:**

\draw[top color = red, bottom color = blue] (100,100) rectangle (200, 200);

**Top color:** El estilo que aplica a la figura es una degradado que va desde arriba hacia abajo.

**Bottom color:** El estilo que aplica a la figura es un degradado que va desde abajo hacia arriba.