

# Documentación del paquete RetosMatematicos

10 de septiembre de 2025

## 1. Cómo usar

La clase `RetoMatematico.cls` puede ser llamada con tres parámetros distintos:

```
1 \documentclass[
2     autor={Yo}
3     fecha={dd de mm de AAAA},
4     palabrasclave={RetoSecundaria, mmmAAAA, álgebra, difX},
5     codigo=minted
6 ]{RetoMatematico}
```

Aquí:

- `autor` representa el autor del documento. Su valor por defecto es José Manuel Sánchez Muñoz.<sup>1</sup>
- `fecha` contiene la fecha de publicación del reto.
- `palabrasclave` denota las categorías en las que se inscribe el reto.<sup>1</sup>
- `codigo` ha de tener el tipo de resaltado de sintaxis que se utilice al incluir código, existen varias opciones para su valor:
  - `ninguno` (por defecto): No se va a incluir código en el documento y por tanto no se carga ningún paquete ni se crea ningún entorno.
  - `verbatim`: Utiliza el habitual `verbatim` con numeración por la parte izquierda, la sintaxis del entorno `codigo` será entonces:

```
1 \begin{codigo}
2 if __name__ == '__main__':
3     print(';Hola mundo!')
4 \end{codigo}
```

---

<sup>1</sup> Este dato puede verse únicamente en los metadatos del documento, los cuales son accesibles, por ejemplo, presionando `Ctrl+D` en Adobe Acrobat.

```

1 if __name__ == '__main__':
2     print(';Hola mundo!')
```

- `minted`: Emplea el más sofisticado paquete homónimo; para su instalación es necesario tener Python y PIP en el dispositivo y ejecutar `pip install latexminted` (`pip3` en Linux), así como llamar al compilador de  $\text{\LaTeX}$  con `-shell-escape`. La sintaxis del entorno código será entonces:

```

1 \begin{codigo}
2 if __name__ == '__main__':
3     print(';Hola mundo!')
```

```

1 if __name__ == '__main__':
2     print(';Hola mundo!')
```

## Macros básicas

Además, `RetoMatematico.cls` provee varios comandos que abstraen las distintas estructuras dentro de una solución:

- `\ejercicio{<enunciado>}{<proponente>}`: Crea una caja con el enunciado del ejercicio y seguidamente el proponente del mismo.<sup>2</sup>

**Ejercicio:** Esto es un ejemplo de un enunciado propuesto. Cabe la posibilidad de que el enunciado tenga varios apartados:

- a) Apartado 1º (puede ir referenciado).
- b) Apartado 2º (puede ir referenciado).

Propuesto por Menganito Flautas.

- `\forma`: Ejecutarlo por primera vez genera *1ª Forma*, por segunda vez *2ª Forma*...

### 1ª Forma

- `\[n]seccion{<título>}`: Genera una sección arbitraria, numerada si el comando posee la `n` opcional.

<sup>2</sup> Existe además el entorno `cajaejercicio` que genera una caja de ejercicio sin ningún contenido en su interior.

## Apartado arbitrario

- `\enfasis{<ecuación>}`: Produce una ecuación en una caja verde que le da énfasis.

$$\int_M K dA + \int_{\partial M} k_g ds = 2\pi\chi(M)$$

- `\resolutores{<resolutores>}`: Escribe *Resuelto por*, seguido de quienes hayan resuelto el problema.

Resuelto por Carl Friedrich Gauss y Pierre Ossian Bonnet

## 2. El paquete RetoExtra

Puede cargar algunas funciones y paquetes adicionales a través del módulo separado `RetoExtra`, tal que así:

```
1 \usepackage{RetoExtra}
```

Las definiciones del mismo son las siguientes:

## Tipografía

- Paquete `mathtools`: Herramientas de tipografía diversas.
- Paquete `multicol`: Varias columnas de texto.
- Paquete `setspace`: Altera el interlineado del documento.
- Comando `\inlinenequation`: Ecuación en línea que puede ser referenciada.
- Paquete `cleveref`: Mejores referencias.
- Paquete `polynom`: Operaciones con polinomios
- Paquete `systeme`: Mejores sistemas de ecuaciones.
- Paquete `cancel`: Cancelar términos en expresiones.
- Paquete `enumitem`: Enumeraciones personalizables.
- Paquete `xfrac`: Fracciones oblicuas.
- Paquete `natbib`: Gran soporte para bibliografías
- Paquete `lipsum`: El conocido "lorem ipsum dolor sit amen".
- Paquete `siunitx`: Permite una mejor tipografía de magnitudes y unidades.

## Símbolos y fuentes

- Paquete `upgreek`: Letras griegas no italizadas.
- Paquete `esvect`: Mejores vectores.
- Paquete `eurosym`: Símbolo de euro.
- Paquete `yhmath`: Fuentes matemáticas extendidas.
- Paquete `wrapfig`: Permite que el texto rodee a la figura.
- Comandos `\newint` y `\otrait`: Mejores integrales.
- Comando `\stretchint{⟨proporción⟩}`: Alargar (sin engordar) una integral.

## Gráficos

- Paquete `pgfplots`: Gráficos de funciones en 2D y 3D.
- Paquete `colortbl`: Colores en tablas.

## 3. Las clases `RetoTikZ` y `RetoPSTricks`

Por último, para la creación de figuras mediante `TikZ` y `PSTricks` se ponen a disposición del usuario las clases `RetoTikZ.cls` y `RetoPSTricks.cls` respectivamente, las cuales no toman ningún argumento y definen los estilos de los diagramas. Además, en el caso de esta última se definen también varias macros de utilidad:

- Comando `\pstBarycenter{⟨A⟩}{⟨B⟩}{⟨C⟩}{⟨nombre⟩}`: Macro para hallar el bari-centro de un triángulo  $\triangle ABC$ .
- Comando `\pstcenter{⟨A⟩}{⟨B⟩}{⟨C⟩}{⟨D⟩}{⟨nombre⟩}`: Macro para hallar el centro de un cuadrilátero  $ABCD$ .

**Nota:** Las figuras que se quieran incluir en el documento probablemente hayan de ser creadas en archivos separados para posteriormente incluir el PDF final en el principal.

*Licenciado bajo Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International.*