

# Informe Minerva

Informes rápidos y fáciles.

v0.3.0

David Ibáñez

<https://github.com/Dav1com/minerva-report-fcfm>

## RESUMEN

**Minerva Report FCFM** es un template de Typst para informes de tareas, laboratorios o trabajos. Pensado para estudiantes y académicos de la Facultad de Ciencias Físicas y Matemáticas de la Universidad de Chile.

## ÍNDICE

|                          |    |
|--------------------------|----|
| 1. Introducción .....    | 1  |
| 2. Configuración .....   | 1  |
| 2.1. meta.typ .....      | 1  |
| 3. Documentación .....   | 1  |
| 3.1. minerva .....       | 1  |
| 3.2. departamentos ..... | 3  |
| 3.3. footer .....        | 5  |
| 3.4. front .....         | 6  |
| 3.5. header .....        | 8  |
| 3.6. rules .....         | 9  |
| 3.7. states .....        | 12 |
| 3.8. util .....          | 12 |

## 1. INTRODUCCIÓN

El objetivo del template es permitir a estudiantes de la Facultad de Ciencias Físicas y Matemáticas escribir informes de forma sencilla, y rápida. Reduciendo mucho la cantidad de archivos y código innecesario que solo se ven como ruido para el usuario final.

Además debe presentarse con un aspecto familiar para quienes llevan años utilizando templates en otros sistemas de documentos.

## 2. CONFIGURACIÓN

La configuración se realiza de 2 formas, primero, los valores generales sobre el informe, como curso, título y autores van en el archivo **meta.typ**, mientras que el cómo se quiere que se vea el documento se configura en la función `informe`.

### 2.1. meta.typ

La mayoría de valores permiten utilizar un `str`, un `array` de strings o `content`. La única excepción es autores, ya que se usa para configurar el autor del PDF!

También se puede configurar la institución del informe, esto se hace en `departamento`, los departamentos incluidos y cómo crear el tuyo propio está en `departamentos`.

## 3. DOCUMENTACIÓN

### 3.1. minerva

#### help

Esta función permite obtener ayuda sobre cualquier función del template. Para saber qué funciones y variables define el template simplemente deja que el autocompletado te guíe, luego puedes llamar esta función para obtener más ayuda.

#### Argumentos

```
help(nombre: str) -> content
```

**nombre**   `str`

Puede ser el nombre de una función o variable, entonces la función entrega ayuda general sobre esta. Si se entrega algo de la forma "help(nombre)" entonces entrega ayuda específica sobre el argumento nombre.

#### informe

Función que aplica los estilos del template para informes.

## Argumentos

```
informe(  
  meta: dictionary module ,  
  portada: function ,  
  header: function ,  
  footer: function ,  
  margenes-portada: dictionary ,  
  margenes: dictionary ,  
  showrules: bool ,  
  doc: content  
) -> content
```

**meta** dictionary or module

Archivo meta.typ.

**portada** function

Portada a usar.

Por defecto: front.portada1

**header** function

Header a usar.

Por defecto: header.header1

**footer** function

Footer a usar.

Por defecto: footer.footer1

**margenes-portada** dictionary

Márgenes de la portada.

Por defecto: (top: 3.5cm)

**margenes** dictionary

Márgenes del documento.

Por defecto: (top: 3.5cm)

**showrules** `bool`

Si es `true` se aplicarán showrules irreversibles. Si se requiere más personalización recomendando desactivar.

Por defecto: `true`

**doc** `content`

Documento a aplicar el template.

## resumen

Crea un resumen que no aparece en el índice (outline).

### Argumentos

`resumen`(body: `content`) -> `content`

**body** `content`

Cuerpo del resumen.

## 3.2. departamentos

### help

Esta función permite obtener ayuda sobre cualquier función del template. Para saber qué funciones y variables define el template simplemente deja que el autocompletado te guíe, luego puedes llamar esta función para obtener más ayuda.

### Argumentos

`help`(nombre: `string`) -> `content`

**nombre** `string`

Puede ser el nombre de una función o variable, entonces la función entrega ayuda general sobre esta. Si se entrega algo de la forma "`help(nombre)`" entonces entrega ayuda específica sobre el argumento nombre.

**adh** `Departamento`

Descripción del Área de Humanidades

**das** `Departamento`

Descripción del Departamento de Astronomía

**dcc** Departamento

Descripción del Departamento de Ciencias de la Computación

**dfi** Departamento

Descripción del Departamento de Física

**dgf** Departamento

Descripción del Departamento de Geofísica

**dic** Departamento

Descripción del Departamento de Ingeniería Civil

**die** Departamento

Descripción del Departamento de Ingeniería Eléctrica

**dii** Departamento

Descripción del Departamento de Ingeniería Industrial

**dim** Departamento

Descripción del Departamento de Ingeniería Matemática

**dimec** Departamento

Descripción del Departamento de Ingeniería Mecánica

**dimin** Departamento

Descripción del Departamento de Ingeniería de Minas

**diqbm** Departamento

Descripción del Departamento de Ingeniería Química, Biotecnología y Materiales

**geo** Departamento

Descripción del Departamento de Geología

### 3.3. footer

#### base

Agrega metadata y estado necesarios para otras funciones del template. Si quieres crear tu propio pie de página, es recomendable pasarlo por esta función.

#### Argumentos

```
base(it: content) -> content
```

**it** content

Contenido del pie de página.

#### footer-numero

Un footer con solamente el número de página.

#### Argumentos

```
footer-numero(  
  meta: dictionary module,  
  romano-hasta-primer-heading: bool  
) -> content
```

**meta** dictionary or module

Contenidos del archivo meta.typ

**romano-hasta-primer-heading** bool

Si es true, las páginas antes del primer heading con numbering utilizan números romanos en minúsculas. Por defecto es true.

Por defecto: true

## footer1

El footer por defecto.

### Argumentos

```
footer1(meta: dictionary module) -> content
```

**meta** dictionary or module

Contenidos del archivo meta.typ

## help

Esta función permite obtener ayuda sobre cualquier función del template. Para saber qué funciones y variables define el template simplemente deja que el autocompletado te guíe, luego puedes llamar esta función para obtener más ayuda.

### Argumentos

```
help(nombre: string) -> content
```

**nombre** string

Puede ser el nombre de una función o variable, entonces la función entrega ayuda general sobre esta. Si se entrega algo de la forma "help(nombre)" entonces entrega ayuda específica sobre el argumento nombre.

## sin-footer

Un header vacío.

### Argumentos

```
sin-footer(meta: dictionary module) -> content
```

**meta** dictionary or module

Contenidos del archivo meta.typ

## 3.4. front

### base

Crea un página y settea variables de estado requeridas por otras funciones del template. Si buscas extender el template con tu propia portada, es recomendado pasarlo por esta función.

### Argumentos

```
base(  
  it: content,  
  ..args: arguments  
) -> content
```

**it** `content`

Contenido de la portada.

**..args** `arguments`

Argumentos a pasar a la función `page`

## help

Esta función permite obtener ayuda sobre cualquier función del template. Para saber qué funciones y variables define el template simplemente deja que el autocompletado te guíe, luego puedes llamar esta función para obtener más ayuda.

### Argumentos

`help(nombre: string) -> content`

**nombre** `string`

Puede ser el nombre de una función o variable, entonces la función entrega ayuda general sobre esta. Si se entrega algo de la forma "`help(nombre)`" entonces entrega ayuda específica sobre el argumento `nombre`.

## portada-simple

Portada que contiene en una página el resumen y el outline.

### Argumentos

```
portada-simple(  
  meta: dictionary module ,  
  espaciado: length ,  
  indice: boolean  
) -> content
```

**meta** `dictionary` or `module`

Contenidos del archivo `meta.typ`.

**espaciado** `length`

Espaciado entre los elementos principales de la portada.

Por defecto: `4cm`



**indice**    `boolean`

Decide si incluir un índice.

Por defecto: `false`

## **portada1**

Diseño de portada básico, perfecto para informes y tareas.

### **Argumentos**

```
portada1(  
  meta: dictionary,  
  titulo-centrado: bool  
) -> content
```

**meta**    `dictionary`

Contenidos del archivo meta.typ

**titulo-centrado**    `bool`

Si es que el título debería ir centrado respecto a la página. Por defecto `false`.

Por defecto: `false`

## **3.5. header**

### **base**

Agrega metadata y estado necesarios para otras funciones del template. Si quieres crear tu propio encabezado, es recomendable pasarlo por esta función.

### **Argumentos**

```
base(it: content) -> content
```

**it**    `content`

Contenido del encabezado.

### **header1**

El header por defecto.

### **Argumentos**

```
header1(  
  meta: dictionary module,  
  romano-hasta-primer-heading: bool  
) -> content
```

**meta**    dictionary or module

Contenidos del archivo meta.typ

**romano-hasta-primer-heading**    bool

Si es true, las páginas antes del primer heading con numbering utilizan números romanos en minúsculas. Por defecto es true.

Por defecto: `true`

## help

Esta función permite obtener ayuda sobre cualquier función del template. Para saber qué funciones y variables define el template simplemente deja que el autocompletado te guíe, luego puedes llamar esta función para obtener más ayuda.

### Argumentos

`help(nombre: string) -> content`

**nombre**    string

Puede ser el nombre de una función o variable, entonces la función entrega ayuda general sobre esta. Si se entrega algo de la forma "help(nombre)" entonces entrega ayuda específica sobre el argumento nombre.

## sin-header

Un header vacío.

### Argumentos

`sin-header(meta: dictionary module) -> content`

**meta**    dictionary or module

Contenidos del archivo meta.typ

## 3.6. rules

### estilos-figure

Aplica los estilos por defecto a las figuras con alguno de los kind especificados.

### Argumentos

```
estilos-figure(  
  kind-target: array ,  
  doc: content  
) -> content
```

**kind-target** `array`

Lista de strings con los kind a afectar.

Por defecto: (`"image"`, `"table"`)

**doc** `content`

Documento a aplicar las reglas.

### formato-numeros-es

Show rule que cambia el formato de los números para usar coma decimal.

|  |                        |
|--|------------------------|
| <code>#show: minerva.rules.formato-numeros-es</code><br><code>\$1.2 + 4.8\$</code> | <code>1,2 + 4,8</code> |
|--|------------------------|

### Argumentos

`formato-numeros-es`(doc: `content`) -> `content`

**doc** `content`

documento a aplicar reglas

### help

Esta función permite obtener ayuda sobre cualquier función del template. Para saber qué funciones y variables define el template simplemente deja que el autocompletado te guíe, luego puedes llamar esta función para obtener más ayuda.

### Argumentos

`help`(nombre: `string`) -> `content`

**nombre** `string`

Puede ser el nombre de una función o variable, entonces la función entrega ayuda general sobre esta. Si se entrega algo de la forma `"help(nombre)"` entonces entrega ayuda específica sobre el argumento nombre.

### operadores-es

Esta show rule cambia los operadores definidos por Typst para que estén en español.

### Argumentos

```
operadores-es(doc: content) -> content
```

**doc** content

Contenido a aplicar las reglas.

### permite-compilar-secundarios

Esta show rule es para utiliza en archivos que no sean `main.typ`, con la idea es permitir que estos archivos sean compilables por separado. Esto es útil para mantener el proyecto ordenado, como también si el documento es demasiado grande como para que la webapp compile en tiempo real. Esta show rules no tienen ningún efecto si el archivo a compilar es `main.typ`.

### Argumentos

```
permite-compilar-secundarios(doc: content) -> content
```

**doc** content

Documento a aplicar la regla.

### permite-ref-rotas

Permite que el documento compile aún si hay referencias rotas, mostrando un mensaje en lugar de la referencia.

### Argumentos

```
permite-ref-rotas(  
  mensaje: content,  
  doc: content  
) -> content
```

**mensaje** content

Mensaje a mostrar.

Por defecto: `text(fill: red, "<ref>")`

**doc** content

Documento a aplicar la regla.

### primer-heading-en-nueva-pag

Hace que el primer heading con numbering esté en una página nueva. Esta show rule es aplicada por defecto en el template. Puede ser desactivada usando el parámetro `showrules: false` en la show rule del template. Puede ser reactivada agregando esta línea:

```
show: primer-heading-en-nueva-pag
```

#### Argumentos

```
primer-heading-en-nueva-pag(doc: content) -> content
```

**doc**    `content`

Documento a aplicar la regla.

## 3.7. states

### help

Esta función permite obtener ayuda sobre cualquier función del template. Para saber qué funciones y variables define el template simplemente deja que el autocompletado te guíe, luego puedes llamar esta función para obtener más ayuda.

#### Argumentos

```
help(nombre: string) -> content
```

**nombre**    `string`

Puede ser el nombre de una función o variable, entonces la función entrega ayuda general sobre esta. Si se entrega algo de la forma `"help(nombre)"` entonces entrega ayuda específica sobre el argumento nombre.

**is-main**    `state`

Este es un state que es usado por el template para saber si el documento siendo compilado es `main.typ` o un documento secundario.

## 3.8. util

### formato-fecha

Aplica el formato “[day] de [month repr: long] del [year]” en español

#### Argumentos

```
formato-fecha( fecha: datetime ) -> string
```

**fecha**    `datetime`

fecha a dar formato.

## help

Esta función permite obtener ayuda sobre cualquier función del template. Para saber qué funciones y variables define el template simplemente deja que el autocompletado te guíe, luego puedes llamar esta función para obtener más ayuda.

### Argumentos

```
help(nombre: string) -> content
```

**nombre**    `string`

Puede ser el nombre de una función o variable, entonces la función entrega ayuda general sobre esta. Si se entrega algo de la forma "help(nombre)" entonces entrega ayuda específica sobre el argumento nombre.

**meses**    `array`

Arreglo con los meses en español.

**días**    `array`

Arreglo con los días de la semana en español.