

# DESCRIPCIÓN DIAGRAMA DE CLASES

## Nombre de la clase: Document

- **Breve descripción de la clase:** Documento que contiene diferentes hojas de cálculo visualizadas y modificadas por el usuario
- **Cardinalidad:** Tantos documentos como el usuario quiera
- **Descripción de los atributos:**
  - *name*: Nombre del documento
  - *cjtSheets*: ArrayList <Sheet> que representa las hojas que tiene un documento
- **Descripción de las relaciones:**
  - Relación de composición con la clase *Sheet*: indica qué hojas de cálculo componen el documento
  - Relación de asociación con la clase *CtrlDocuments*: relación con la clase que permite gestionar los documentos

## Nombre de la clase: Sheet

- **Breve descripción de la clase:** Hoja de cálculo formada por celdas, a las cuales se les puede aplicar diversas operaciones
- **Cardinalidad:** Tantas hojas de cálculo como quiera el usuario, pero como mínimo 1 hoja
- **Descripción de los atributos:**
  - *name*: Nombre de la hoja, se puede modificar
  - *maxRow*: Número máximo de filas que puede tener una hoja, default 255
  - *maxColumn*: Número máximo de columnas que puede tener una hoja, default 255
  - *cjt\_cells*: Estructura del Sheet: una hoja es una matriz de celdas.
  - *references*: Referencias en esa hoja.
- **Descripción de las relaciones:**
  - Relación de composición con *Document*: Es el documento al cual pertenecen
  - Relación de composición con *Cell*: Indica las celdas que contiene la hoja
  - Relación de asociación con clase *References*: Indica el conjunto de referencias para todas las celdas de esa hoja

## Nombre de la clase: References

- **Breve descripción de la clase:** Conjunto de referencias de las celdas de la hoja.
- **Cardinalidad:** Tantas referencias como el usuario quiera, pero solo referencias dentro de la misma hoja, no referencias entre hojas distintas
- **Descripción de los atributos:**
  - *keys*: ArrayList de los identificadores de las celdas referenciadas
  - *values*: ArrayList<ArrayList> con las celdas que usan referencias de las celdas que están en keys. Cada índice i de values[i][j] corresponde a la celda con key i, y dentro del array con índice j, están las celdas que usan la referencia de la

celda con key i. Es decir, en `values[0][j]`, en todas las posiciones de j se encuentran las celdas que contienen la celda referenciada con key 1.

- **Descripción de las relaciones:**

- Relación de asociación con *Sheet*: Indica las referencias para las celdas de esa hoja

**Nombre de la clase:** Cell

- **Breve descripción de la clase:** Cada una de las celdas que forman una hoja

- **Cardinalidad:**  $255^2$  celdas como máximo en un Sheet por default, y mínimo una.

- **Descripción de los atributos:**

- *row*: Fila a la que pertenece la celda
- *column*: Columna a la que pertenece la celda
- *cellValue*: Atributo de tipo `CellValueInterface` que implementa el tipo de valor en una celda

- **Descripción de las relaciones:**

- Relación de composición con la clase *Sheet*: indica qué celdas componen la hoja de cálculo
- Relación de asociación con la interfaz `CellValueInterface`: indica el tipo del valor contenido en la celda

**Nombre de la clase:** `CellValueInterface`

- **Breve descripción de la clase:** Interfaz del tipo de valor que puede haber en una celda

- **Cardinalidad:** Un tipo de valor por cada celda

- **Descripción de los atributos:**

- No tiene atributos

- **Descripción de las relaciones:**

- Relación de asociación con la clase `Cell`: indica el tipo del valor contenido en una celda
- Implementada por `Función` y `CellValue`

**Nombre de la clase:** `CellValue`

- **Breve descripción de la clase:** Clase que implementa la interfaz `CellValueInterface`, y que contiene el tipo de valor que puede haber en una celda que no sea función

- **Cardinalidad:** Implementa `CellValueInterface`, por lo tanto la cardinalidad es un tipo de valor por cada celda

- **Descripción de los atributos:**

- *value*: Valor que tiene la celda (no función)

- **Descripción de las relaciones:**

- Implementa `CellValueInterface`

**Nombre de la clase:** Function

- **Breve descripción de la clase:** Clase abstracta que implementa la interface CellValueInterface, y que indica el tipo de función que hay en una celda
- **Cardinalidad:** Una función por celda (por el momento)
- **Descripción de los atributos:**
  - *formula*: Guarda la función aplicada a la celda
  - *parameters*: Indica los parámetros que se les aplicará la función
- **Descripción de las relaciones:**
  - Implementa CellValueInterface

**Nombre de la clase:** GetFunctionFactory

- **Breve descripción de la clase:** Classe que genera un objeto de las clases concretas de función, basándose en el tipo de función que el usuario decida
- **Cardinalidad:** Crea Función, por lo tanto una Factoría para todas las funciones.
- **Descripción de los atributos:**
  - *instances*: Map cuya key es el nombre de la función y el value es la instancia de dicha función
- **Descripción de las relaciones:**
  - Relación de creación con Function

**Nombre de la clase:** Truncate

- **Breve descripción de la clase:** Contiene la función *getValue*, que aplica el truncado a un número decimal pasado como parámetro, de tantos decimales como se indique en la función
- **Cardinalidad:** Una función de truncado por celda
- **Descripción de los atributos:** No hay atributos.
- **Descripción de las relaciones:**
  - Subclase de la clase *Function*

**Nombre de la clase:** Absolute

- **Breve descripción de la clase:** Contiene la función *getValue*, que devuelve el valor absoluto del número pasado como parámetro
- **Cardinalidad:** Una función de valor absoluto aplicada por celda
- **Descripción de los atributos:** No tiene atributos
- **Descripción de las relaciones:**
  - Subclase de la clase *Function*

**Nombre de la clase:** Floor

- **Breve descripción de la clase:** Contiene la función *getValue*, que devuelve el redondeo hacia abajo de un número decimal
- **Cardinalidad:** Una función de redondeo hacia abajo por celda
- **Descripción de los atributos:** No tiene atributos
- **Descripción de las relaciones:**
  - Subclase de la clase *Function*

**Nombre de la clase:** Increment

- **Breve descripción de la clase:** Contiene la función *getValue*, que devuelve el número pasado como parámetro incrementado en uno
- **Cardinalidad:** Una función de incremento por celda
- **Descripción de los atributos:** No tiene atributos
- **Descripción de las relaciones:**
  - Subclase de la clase *Function*

**Nombre de la clase:** Mean

- **Breve descripción de la clase:** Contiene la función *getValue*, que devuelve el promedio o tendencia central de un conjunto de números
- **Cardinalidad:** Una función de media por celda
- **Descripción de los atributos:** No tiene atributos
- **Descripción de las relaciones:**
  - Subclase de la clase *Function*

**Nombre de la clase:** Identity

- **Breve descripción de la clase:** Contiene la función *getValue*, que devuelve el mismo número de una celda determinada
- **Cardinalidad:** Una función de identidad por celda
- **Descripción de los atributos:** No tiene atributos
- **Descripción de las relaciones:**
  - Subclase de la clase *Function*

**Nombre de la clase:** Median

- **Breve descripción de la clase:** Contiene la función *getValue*, que devuelve el valor de la variable de posición central en un conjunto de datos ordenados
- **Cardinalidad:** Una función de mediana por celda
- **Descripción de los atributos:** No tiene atributos
- **Descripción de las relaciones:**
  - Subclase de la clase *Function*

**Nombre de la clase:** Variance

- **Breve descripción de la clase:** Contiene la función *getValue*, que devuelve la suma de los residuos al cuadrado divididos entre el total de observaciones.
- **Cardinalidad:** Una función de varianza por celda
- **Descripción de los atributos:** No tiene atributos
- **Descripción de las relaciones:**
  - Subclase de la clase *Function*

**Nombre de la clase:** Covariance

- **Breve descripción de la clase:** Contiene la función *getValue*, que indica el grado de variación conjunta de dos variables aleatorias respecto a sus medias
- **Cardinalidad:** Una función de covarianza por celda

- **Descripción de los atributos:** No tiene atributos
- **Descripción de las relaciones:**
  - Subclase de la clase *Function*

**Nombre de la clase:** StandardDeviation

- **Breve descripción de la clase:** Contiene la función *getValue*, que devuelve la cuantificación de la dispersión del conjunto de números pasado como parámetro
- **Cardinalidad:** Una función de desviación estándar por celda
- **Descripción de los atributos:** No tiene atributos
- **Descripción de las relaciones:**
  - Subclase de la clase *Function*

**Nombre de la clase:** PearsonCorrelation

- **Breve descripción de la clase:** Contiene la función *getValue*, que mide el grado de relación de dos variables siempre y cuando ambas sean cuantitativas y continuas.
- **Cardinalidad:** Una función de coeficiente de correlación de Pearson por celda
- **Descripción de los atributos:** No tiene atributos
- **Descripción de las relaciones:**
  - Subclase de la clase *Function*

**Nombre de la clase:** ReplaceText

- **Breve descripción de la clase:** Contiene la función *getValue*, que sustituye la parte indicada de un String por otra
- **Cardinalidad:** Una función de reemplazar texto por celda
- **Descripción de los atributos:** No tiene atributos
- **Descripción de las relaciones:**
  - Subclase de la clase *Function*

**Nombre de la clase:** LengthText

- **Breve descripción de la clase:** Contiene la función *getValue*, que devuelve la longitud de un String contenido en una celda
- **Cardinalidad:** Una función de longitud de texto por celda
- **Descripción de los atributos:** No tiene atributos
- **Descripción de las relaciones:**
  - Subclase de la clase *Function*

**Nombre de la clase:** ElementExtraction

- **Breve descripción de la clase:** Contiene la función *getValue*, que devuelve el número del día, mes o año respectivamente, de una fecha
- **Cardinalidad:** Una función de extracción de elementos, ya sea de día, mes o año, por celda
- **Descripción de los atributos:** No tiene atributos
- **Descripción de las relaciones:**
  - Subclase de la clase *Function*

**Nombre de la clase:** DayOfWeek

- **Breve descripción de la clase:** Contiene la función *getValue*, que devuelve un *String* del día de la semana de la fecha indicada
- **Cardinalidad:** Una función de día de la semana por celda
- **Descripción de los atributos:** No tiene atributos
- **Descripción de las relaciones:**
  - Subclase de la clase *Function*

**Nombre de la clase:** Parameter

- **Breve descripción de la clase:** interface parámetro y cada elemento que implementa es un parámetro
- **Cardinalidad:** Varios parámetros en una función (mínimo uno)
- **Descripción de los atributos:** No tiene atributos
- **Descripción de las relaciones:**
  - Relación de asociación con *Function*: una función tiene diferentes parámetros en su entrada

**Nombre de la clase:** ArrayDoubleP

- **Breve descripción de la clase:** Parámetro de tipo Array de doubles que implementa la interfaz *Parameter*
- **Cardinalidad:** Implementa *Parameter*, por lo tanto varios parámetros o ninguno de tipo *ArrayDouble* por función.
- **Descripción de los atributos:** No tiene atributos
- **Descripción de las relaciones:**
  - Implementa *Parameter*

**Nombre de la clase:** DoubleP

- **Breve descripción de la clase:** Parámetro de tipo double que implementa la interfaz *Parameter*
- **Cardinalidad:** Implementa *Parameter*, por lo tanto varios parámetros o ninguno de tipo *Double* por función.
- **Descripción de los atributos:** No tiene atributos
- **Descripción de las relaciones:**
  - Implementa *Parameter*

**Nombre de la clase:** StringP

- **Breve descripción de la clase:** Parámetro de tipo String que implementa la interfaz *Parameter*
- **Cardinalidad:** Implementa *Parameter*, por lo tanto varios parámetros o ninguno de tipo *String* por función.
- **Descripción de los atributos:** No tiene atributos
- **Descripción de las relaciones:**
  - Implementa *Parameter*