**Projet ScIn1B**

Dijon 16/12/2019

Intégrants :

Cartaya Agustin

Moussa Hassan

Enseignant :

Bricq Stephanie

***Version ES***

Índex

[Projet 3](#_Toc27300180)

[Funcionalidad del proyecto 3](#_Toc27300181)

[Modularizacion del código 3](#_Toc27300182)

[Fichero shell (main.sh) 3](#_Toc27300183)

[Ficheros perl (Ex#.pl) 4](#_Toc27300184)

[Ex1.pl 4](#_Toc27300185)

[Ex2.pl 4](#_Toc27300186)

[Ex3.pl 5](#_Toc27300187)

[Ex4.pl 5](#_Toc27300188)

[Ex5.pl 6](#_Toc27300189)

[Ficheros Gnuplot (g1.gp y g1.gp) 7](#_Toc27300190)

[g1.gp 7](#_Toc27300191)

[g2.gp 7](#_Toc27300192)

[Guia de uso 8](#_Toc27300193)

[Remarcas: 8](#_Toc27300194)

[Pasos 8](#_Toc27300195)

[Resultados 8](#_Toc27300196)

# Projet

## Funcionalidad del proyecto

Todo el codigo escrito en este proyecto tiene como objetivo, leer, clasificar, interpretar y graficar una tabla de datos que contiene información de objetos proveniente del Metropolitan Museum of Art, datos cuyos significado y presentación fueron explicados en el enunciado propuesto.

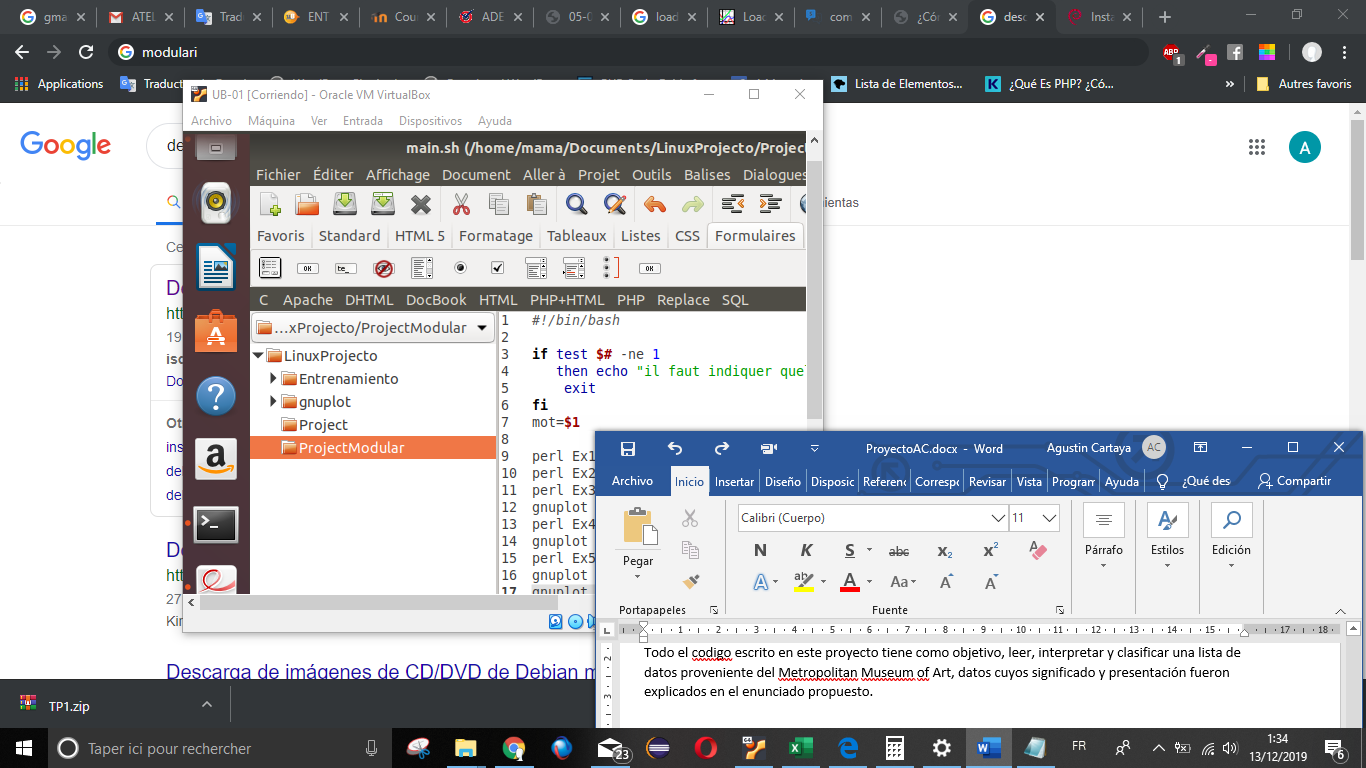
## Modularizacion del código

El código que será presentado está dividido en:

* 1 fichero Shell (.sh)
* 5 ficheros Perl (.pl)
* 2 ficheros Gnuplot (.gp)

### Fichero shell (main.sh)

Es el fichero principal encargado de ejecutar todos los demás y pasarle automáticamente los datos necesarios. Este fichero es el único que se llamara por consola al momento de ejecutar el proyecto y necesita como parámetro el nombre del fichero que contiene los datos a estudiar.

(Linea 1) Cabecera sirve para especificar el camino completo y los parámetros del intérprete de que ejecutará el programa.

(líneas 3-7) condicional encargado de verificar si se ha pasado como parámetro el nombre del fichero que contiene la tabla de datos, en caso de que ese parámetro no exista imprime en consola un error (línea 4) y sale del programa (línea 5). En caso de que si exista dicho parámetro este se guarda en la variable $table (línea 7) y se continuara con la ejecución del programa.

(líneas 9-17) ejecución de ficheros .pl y .gp encargados de clasificar los datos.

* Los archivos perl son ejecutados escribiendo la palabra “perl” y luego el nombre del fichero a ejecuta ejemplo *perl Ex2.pl* si el archivo a ejecutar necesita que se le pase un parámetro (caso único Ex1.pl) este se coloca luego del nombre del fichero ejemplo *perl Ex1.pl $table*
* Los archivos gnuplot se ejecutan anteponiendo la palabra “gnuplot” luego con el uso del interruptor “-e” se indica que se ingresaran variables y por ultimo el nombre del archivo .gp a ejecutar ejemplo *gnuplot -e “fn1 = ‘nomFichier\_dep’” g1.gp*

### Ficheros perl (Ex#.pl)

Existen 5 ficheros .pl los cuales son los encargados de leer y clasificar la tabla de datos según los ejercicios, todos los ficheros tienen una estructura similar, la cual fue dividida en bloques para facilitar la comprensión.

En cada fichero existen 6 bloques enumerado de A-F, si no se indica lo contrario cada bloque tiene la siguiente función

Bloque #A: creación de variables comunes las cuales sirven para la lectura de los datos

* $fichier: variable donde se guardara el fichero de lectura
* $ligne: variable donde se la línea de turno en el bucle
* @par: arreglo donde se guardaran cada una de la columnas de la línea de turno en el bucle
* $sorti: variable donde se guardara el fichero de escritura

Bloque #B: creación de variables propias para el ejercicio

Bloque #C: abertura de ficheros de lectura y escritura

Bloque #D: inicio del bucle while

Bloque #D1: obtención de línea en curso y división de columnas con el método split()

Bloque #D2: cuerpo del ejercicio (condicionales, contadores y escritura de ficheros)

Bloque #E: Escritura de resultados en ficheros o en consola

Bloque #F: cierre de ficheros

#### Ex1.pl

Bloque #B (variables)

* $valides: contador de líneas validas
* $publics: contador de objetos públicos
* $date: contador de objetos con fecha
* $p contador de objetos cuyos nombres comienzan por p

Bloque #D2

|  |  |
| --- | --- |
| Condicion | Función |
| if(@par[2] =~ /^[0-9]+$/) | Verificar que la columna 3 solo contenga números |
| if(@par[1] =~ /true/i) | Verifica que la columna 2 contenga la palabra true (sin distinción de mayúsculas) |
| if( @par[7] =~ /[a-zA-z0-9]+/ && @par[7] !~ /n\.d/i) | Verificar que la columna 8 contenga al menos un carácter alfanumérico y que sea distinto a “n.d” (sin distinción de mayúsculas) |
| if( @par[4] =~ /^P/) | Verificar que la columna 5 comience por P (Mayúscula) |

#### Ex2.pl

Bloque #B (variables)

* $sd: contador de objetos sin dimensión

Bloque #D2

|  |  |
| --- | --- |
| Condicion | Función |
| If ( @par[9] !~ /[0-9]+/) | Verificar que la columna 10 no contenga numeros |

#### Ex3.pl

Bloque #B (variables)

* @dCtt: contadores de objetos por departamento
  + dCtt[0] = Contador de objetos de American Decorative Arts
  + dCtt[1] = Contador de objetos de Drawings and Prints
  + dCtt[2] = Contador de objetos de Medieval Art
  + dCtt[3] = Contador de objetos de Egyptian Art

Bloque #D2

|  |  |
| --- | --- |
| Condicion | Función |
| if (@par[3] =~ /(American Decorative Arts)/i) | Verificar que la columna 4 contenga “American Decorative Arts” (sin distinción de mayúsculas) |
| elsif (@par[3] =~ /(Drawings and Prints)/i) | Verificar que la columna 4 contenga “Drawings and Prints” (sin distinción de mayúsculas) |
| elsif (@par[3] =~ /(Medieval Art)/i) | Verificar que la columna 4 contenga “Medieval Art” (sin distinción de mayúsculas) |
| elsif (@par[3] =~ /(Egyptian Art)/i) | Verificar que la columna 4 contenga “Egyptian Art” (sin distinción de mayúsculas) |

#### Ex4.pl

Bloque #B (variables)

@mCtt: contadores de objetos por clasificación

* + dCtt[0] = Contador de objetos con clasificación metal
  + dCtt[1] = Contador de objetos con clasificación silver
  + dCtt[2] = Contador de objetos con clasificación prints
  + dCtt[3] = Contador de objetos con clasificación drawings
  + dCtt[4] = Contador de objetos con clasificación ceramics
  + dCtt[5] = Contador de objetos con clasificación ephemera
  + dCtt[6] = Contador de objetos con alguna otra clasificación
  + dCtt[7] = Contador de objetos sin clasificación

Bloque #D2

|  |  |
| --- | --- |
| Condicion | Función |
| if (@par[11] =~ /metal/i) | Verificar que la columna 12 contenga “metal” (sin distinción de mayúsculas) |
| if (@par[11] =~ / silver /i) | Verificar que la columna 12 contenga “silver” (sin distinción de mayúsculas) |
| if (@par[11] =~ / prints /i) | Verificar que la columna 12 contenga “prints” (sin distinción de mayúsculas) |
| if (@par[11] =~ / drawings /i) | Verificar que la columna 12 contenga “drawings” (sin distinción de mayúsculas) |
| if (@par[11] =~ / ceramics /i) | Verificar que la columna 12 contenga “ceramics” (sin distinción de mayúsculas) |
| if (@par[11] =~ / ephemera /i) | Verificar que la columna 12 contenga “ephemera” (sin distinción de mayúsculas) |
| if (@par[11] !~ /[a-z0-9]/i) | Verificar que la columna 12 no este vacia |
| elsif (@par[11] !~ /(metal|silver|prints|Drawings|ceramics|Ephemera) | Verifica que la columna 12 no contenga ninguna de las categorías anteriores |

#### Ex5.pl

Bloque #B (variables)

* $h: variable de altitud
* $w: variable de longitud
* $ar: variable de area

Bloque #D2

|  |  |
| --- | --- |
| Condicion | Función |
| if(@par[4] =~ /Print/ && @par[9] =~ /[0-9]+/) | Verificar que la columna 5 contenga “Print” y que la columna 10 tenga al menos un numero |
| if( @par[9] =~ /\(\D\*(\d+[.,]?\d\*)\D\*\).\*\(\D\*(\d+[.,]?\d\*)\D\*\)/) | Verificar que la columna 10 tenga una estructura parecida a “(#.#) (#.#)” los # indican una cifra, y guardando esos 2 valores numéricos en las variables $1 y $2 (luego esos valores serán trasladados a las variables $h y $w) |
| elsif( @par[9] =~ /\(\D\*(\d+[.,]?\d\*)\D\*(\d+[.,]?\d\*)\D\*\)/) | Verificar que la columna 10 tenga una estructura parecida a “(#.# x #.#)” |
| if(@par[4] =~ /Drawing/ && @par[9] =~ /[0-9]+/) | Verificar que la columna 5 contenga “Drawing” y que la columna 10 tenga al menos un numero |

Nota

se utilizaron las variables $h y $w para cambiar las “,” por “.” Ya que no está claro el separador de decimales

### Ficheros Gnuplot (g1.gp y g1.gp)

Esisten 2 ficheros gnuplot los cuales se encargan de hacer las graficas correspondientes a los resultados obtenidos en algunos ejercicios

#### g1.gp

Este fichero esta dividido en 5 bloques enumerados de A-E

* Bloque #A: Se encarga de resetear Gnuplot para no tener conflictos con parámetros establecidos anteriormente, además establece un titulo y activa las líneas guias
* Bloque #B: Se encarga de establecer los parámetros para crear un histograma
* Bloque #C: Establece la forma de mostrar las etiquetas en el eje de las x
* Bloque #D: Establece la salida como archivo png con el nombre pasado como argumento guardado en la variable nf
* Bloque #E: se crea la grafica usando la segunda columna como datos y la primera como labels para el eje de las x

#### g2.gp

Este fichero esta dividido en 5 bloques enumerados de A-E

* Bloque #A: Se encarga de resetear Gnuplot para no tener conflictos con parámetros establecidos anteriormente, además establece un titulo y activa las líneas guias
* Bloque #B: Se encarga de establecer los parámetros para crear un Boxplot
* Bloque #C: Establece la forma de mostrar las etiquetas en el eje de las x
* Bloque #D: Establece la salida como archivo png con el nombre pasado como argumento guardado en la variable nf
* Bloque #E: se crea la gráfica usando la segunda columna como datos y la primera como labels para el eje de las x

## Guia de uso

### Remarcas:

* Se debe tener instalado, Perl y Gnuplot para la ejecución del proyecto
* Se debe de tener un lista de datos cuyo nombre se pasara por parámetro al archivo main.sh

### Pasos

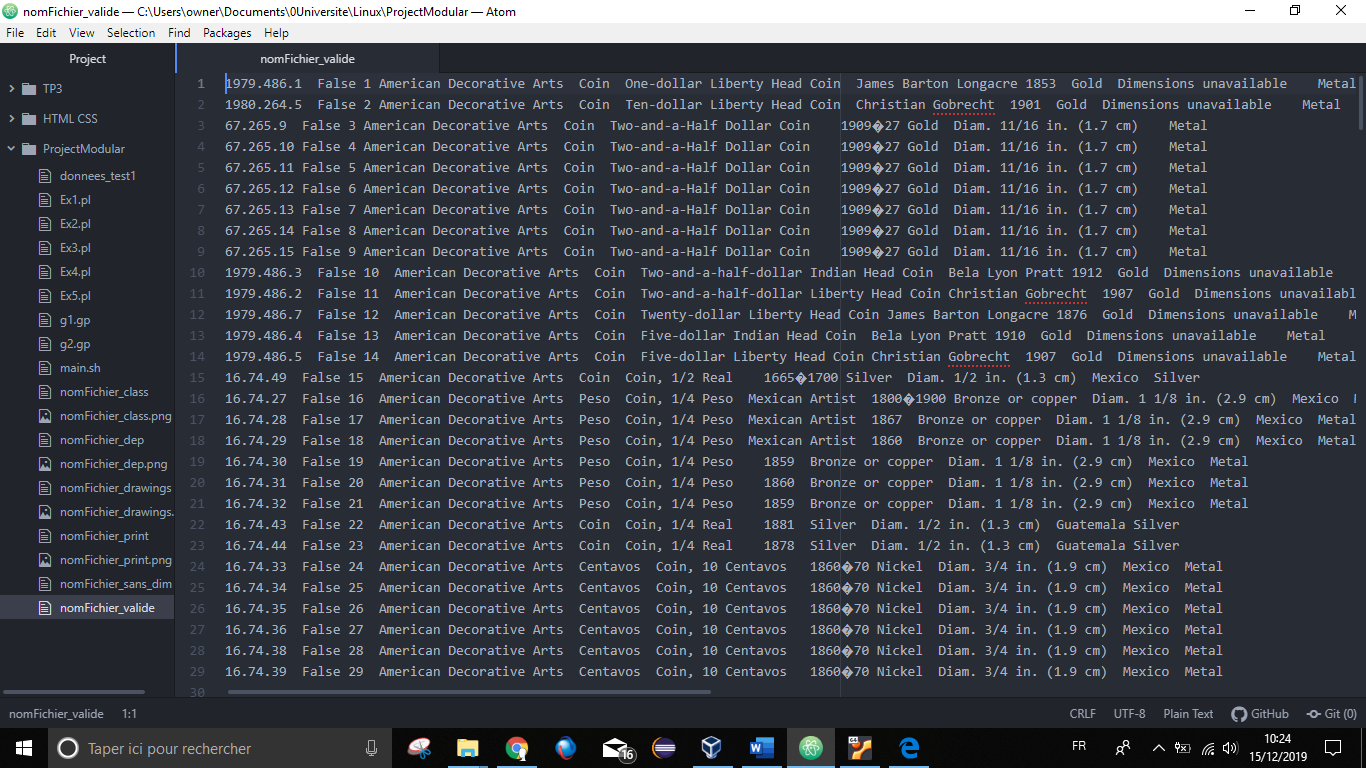
* Descomprimir el archivo projet\_ScIn1B\_Cartaya\_Moussa.zip
* Entrar en la carpeta projet\_ScIn1B\_Cartaya\_Moussa
* Abrir un la terminal en esa ubicación
* Dar permisos de ejecución al archivo main.sh para eso se ejecutará el comando chmod u+x en el terminal
* Ejecutar el programa, para eso se escribirá ./main.sh seguido del nombre del fichero que contiene la lista de datos Ejemplo *./main.sh donnees\_test1*

## Resultados

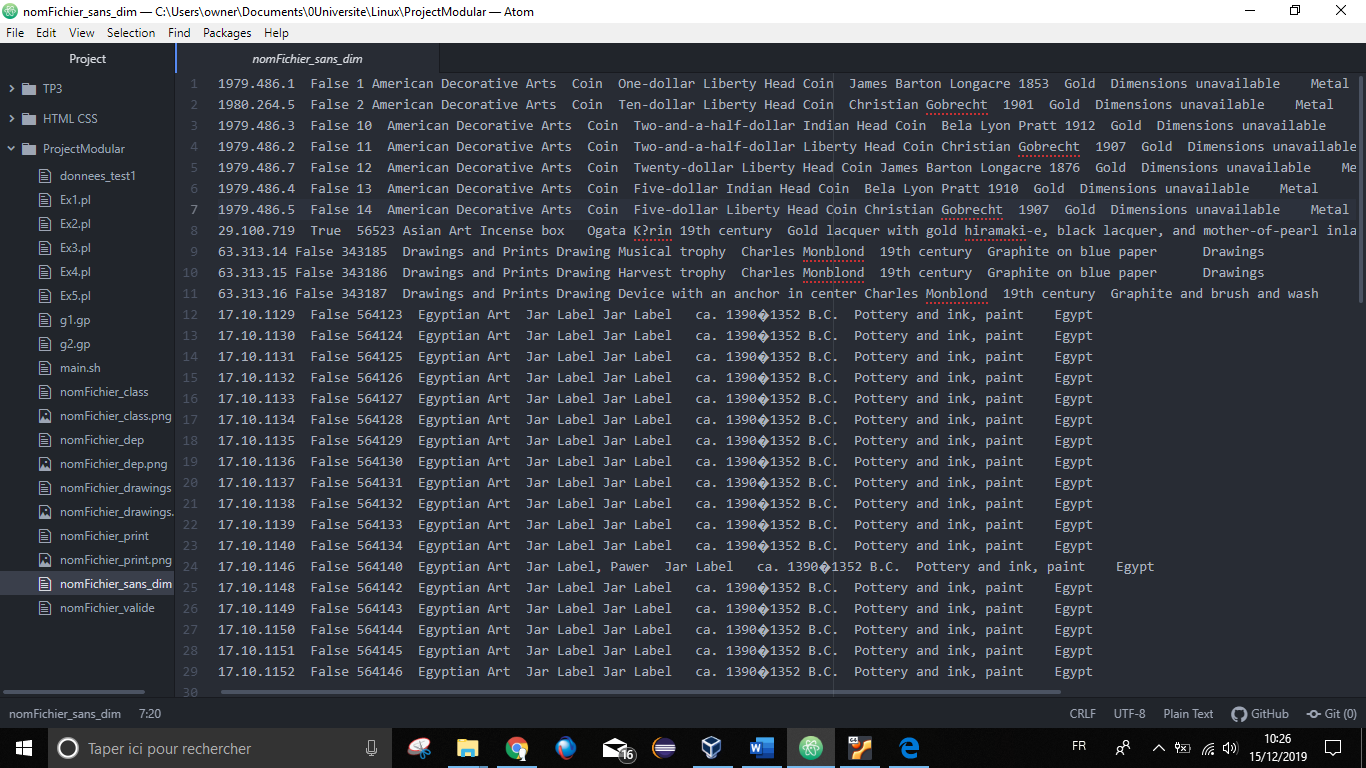
Al ejecutar el proyecto este imprimirá por consola

* Numero de líneas vailidas
* Numeroo de objetos publicos
* Numeroo de objetos con fecha
* Numeroo de objetos cuyo nombre comienza por P
* Numeroo de objetos sin dimensión

Ademas este creara 6 ficheros de texto correspondientes a cada ejercicio propuesto (para el ejercicio Nro 5 se crearon 2 ficheros) y 4 graficas correspondientes a los ejercicios 3, 4, 5. A continuación se presentan los resultados:

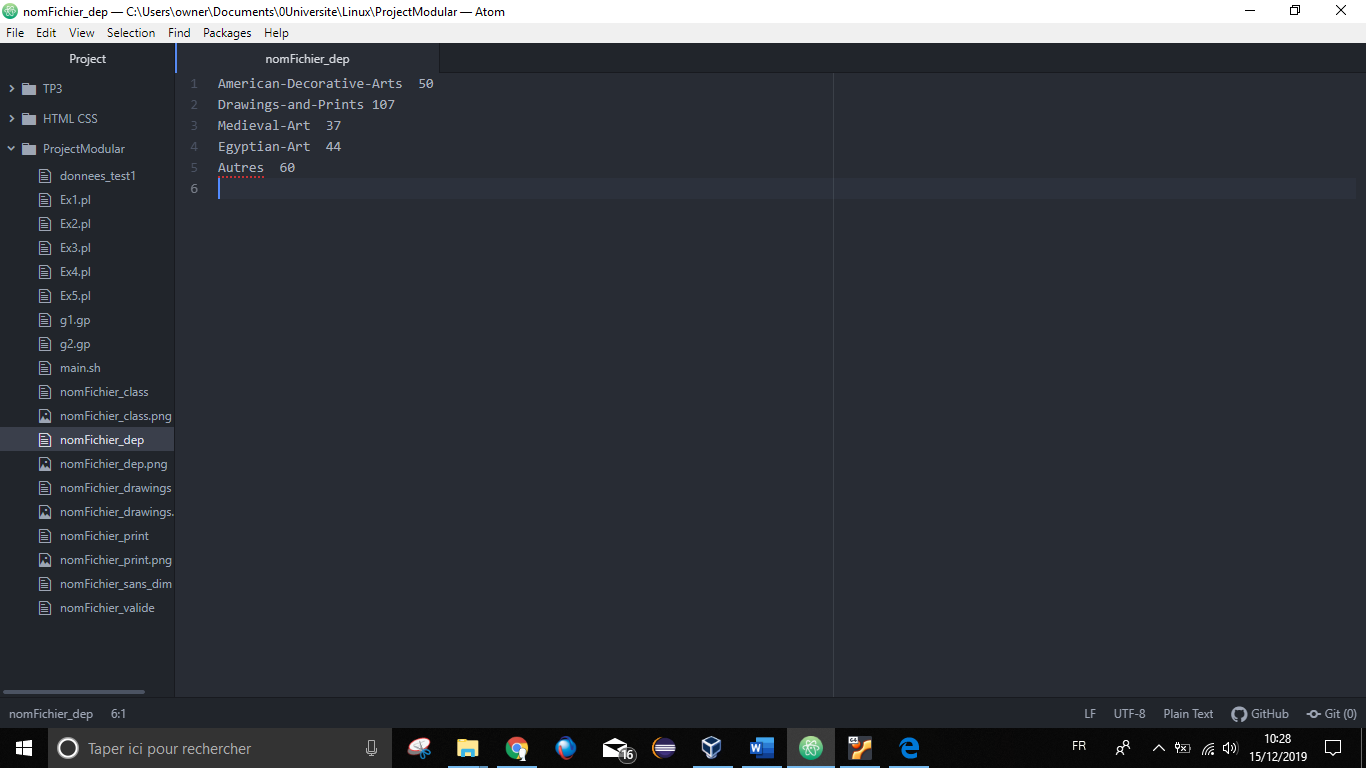
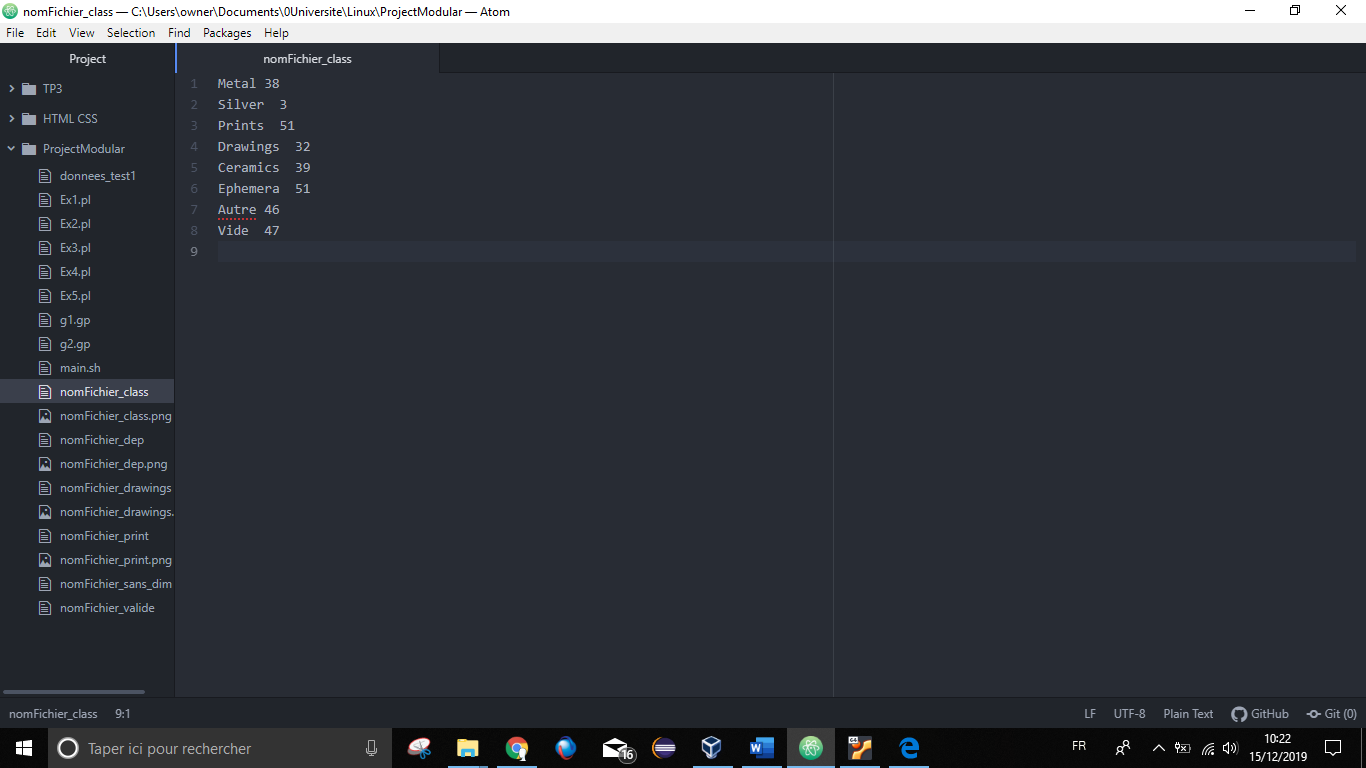
Ex1.pl (nomFichier\_valide)

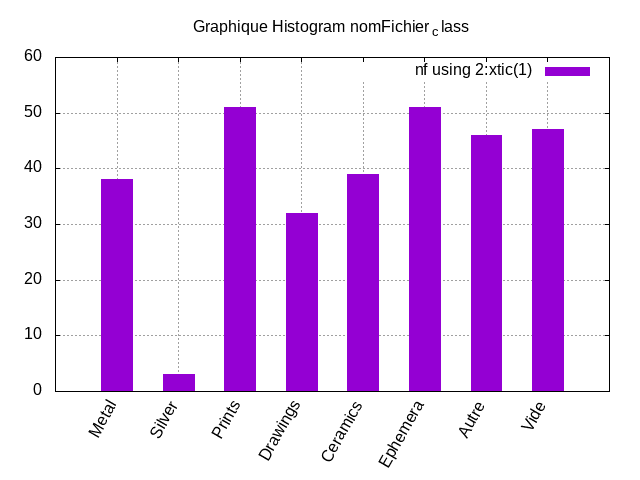
Ex2 (nomFichier\_sans\_dim)

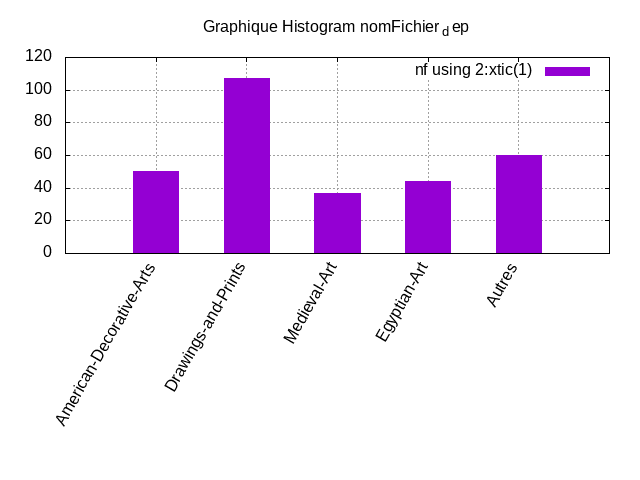


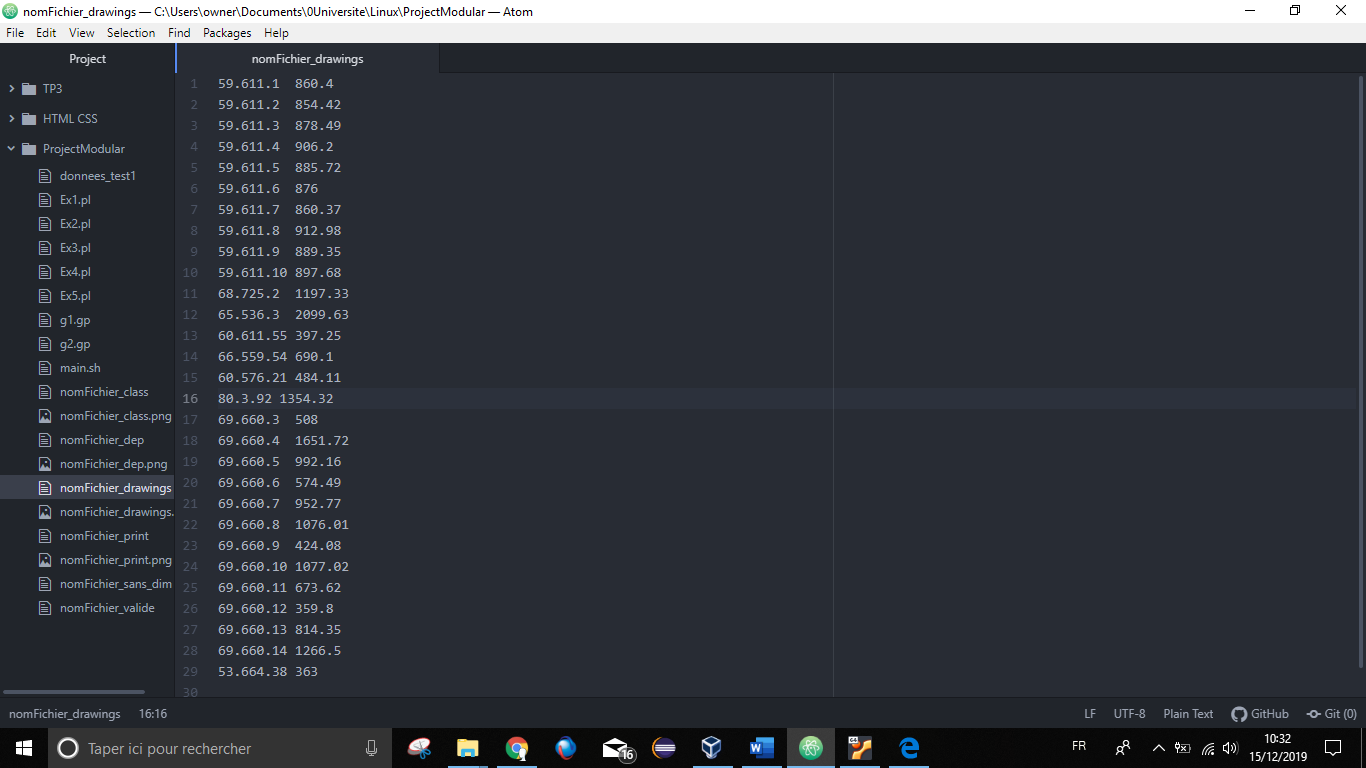
Ex4 (nomFichier\_class)

Ex3 (nomFichier\_dep)







Ex5 (nomFichier\_print, nomFichier\_drarings)

