



Words	753
Characters	8642

Exclude URL

None

```
# include "Files.h" /** * @brief save_files Funcion crea un archivo * @param persona de donde se obt
Files::save_files(Persona persona) { fstream archivo; fstream usuarios; FILE nuevo; check_files(); ar
(archivo.fail() || usuarios.fail()) { cout << "Ha ocurrido un error al verificar el archivo" << endl; exit(1); } else
persona.setNombre(auxPalabra);    archivo    >>    auxPalabra;    persona.setApellido(auxPalabra);
persona.setDireccion(auxPalabra); archivo >> auxNum; persona.setTelefono(auxNum); archivo >> auxPal
archivo >> auxPalabra; persona.setPlaca(auxPalabra); } } archivo.close(); system("pause"); } /** * @brief re
los Usuarios * @param person donde leemos datos del cliente y prestamo */ void Files::read_files(Auto
std::string auxPalabra = ""; std::string auxUsuario = ""; long auxNum = 0; double saldoAux = 0; che
(archivo.fail()) { cout << "Error, no se pudo encontrar el archivo" << endl; exit(1); } else { while (!arc
vehiculo.setPlaca(auxPalabra); archivo >> auxPalabra; vehiculo.setColor(auxPalabra); archivo >> auxPal
archivo >> auxPalabra; vehiculo.setModelo(auxPalabra); } } archivo.close(); system("pause"); } /** * @br
archivo de los Usuarios */ void Files::check_files() { fstream archivo; fstream usuarios; archivo.open("U.tx
"Error al verificar Banco.txt" << endl; system("pause"); create_files(); system("cls"); check_files(); } else { } a
@brief craete_files Funcion que crea archivos a los Usuarios */ void Files::create_files() { fstream ar
"Usuarios.txt"; archivo.open(ruta.c_str(), ios::out); if (archivo.fail() || usuarios.fail()) { cout << "Ha ocurrido
system("cls"); exit(1); } archivo.close(); usuarios.close(); } void Files::create_pdf(Persona persona,Auto vehic
"" "" "" : html << "
```

NO	FECHAS DE PAGO	DIA	CAPITAL	INTERES
" +	" + node-	" + node-	" +	" +
std::to_string(i-	>get_data().get_date()	>get_data().get_weekday()	Utils::Generator::to_string(node-	Utils::Generator::to_string(noc
1) + "	+	+	>get_data().get_capital()) + "	>get_data().get_interest()) +
"; // std::ostringstream plain; std::string html_filename = "data.html"; std::string pdf_filename = "data.pdf"; std::ofstream out_html(html_				
out_txt("data.txt", std::ios::trunc); out_html << html.str(); out_txt.close(); out_html.close(); if (std::ifstream(pdf_filename.c_str()).good()				
system((std::string("wkhtmltopdf.exe ") + html_filename + " " + pdf_filename).c_str()); std::cout << std::endl << "archivo pdf generado!"; :				



[Home](#)
[Blog](#)
[Testimonials](#)
[About Us](#)
[Privacy](#)

Copyright © 2022 Plagiarism Detector. All right reserved