Notepad Application

MINOR PROJECT

Submitted in Partial Fulfilment of the Requirements for the Award of the Degree of

MASTER OF COMPUTER APPLICATIONS

(M.C.A.)

BY

RIYA KUMARI KUSHWAHA

(03735304420)

MCA – 1st SEMESTER

BHARATI VIDYAPEETH'S INSTITUTE OF COMPUTER APPLICATIONS AND MANAGEMENT (BVICAM)

(GURU GOBIND SINGH INDRAPRASTHA UNIVERSITY)

NEW DELHI

April, 2021

CANDIDATE'S DECLARATION

I hereby declare that the work which is being presented in this project work entitled "**Notepad Application**" in partial fulfilment of the requirements for the award of the degree of **Master in Computer Applications at Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM), New Delhi** is an authentic record of my own work carried out during the period January 2021 to April 2021 under the supervision and guidance of **Dr. Vaishali Joishi (Associate Professor, BVICAM)**.

I have not submitted the matter embodied in this project work anywhere for the award of any degree or diploma.

Riya Kumari Kushwaha
Enroll. No. 03735304420

ACKNOWLEDGEMENT

It is my proud privilege to express my profound gratitude to the entire management of Bharati Vidyapeeth's Institute of Computer Applications and Management and teachers of the institute for providing me with the opportunity to avail the excellent facilities and infrastructure. The knowledge and values inculcated have proved to be of immense help at the very start of my career. Special thanks to Hon'ble Founder, Bharati Vidyapeeth, Pune for having provided us an excellent infrastructure at BVICAM.

I am grateful to **Dr. Vaishali Joshi** (Project Guide, BVICAM, New Delhi) for her astute guidance, constant encouragement and sincere support for this project work.

I feel pride and privileged in expressing my deep sense of gratitude to all those who have helped me in presenting this assignment. I would be failing in my endeavour if I do not place my acknowledgement**.**

RIYA KUMARI KUSHWAHA

Enroll. No. 03735304420

ABSTRACT

In this minor project, I have developed a java-based text editor application that resembles a notepad. The application is developed using Java GUI technology with Java programming. No database is used in the development of this application. The development of application is carried out by following the waterfall model of software development. The application follows Model-View-Controller (MVC) architecture for providing robust and reliable services. There are mainly 3 modules of the system: file,edit and format.

TABLE OF CONTENTS

CHAPTER 1

Introduction

A text editor is a type of computer program that edits plain text. Such programs are sometimes known as "notepad" software, following the naming of Microsoft Notepad. Text editors are provided with operating systems and software development packages, and can be used to change files such as configuration files, documentation files and programming language source code.

a. Problem Description

Text Editor: - A text editor is a type of program used for editing plain text files. A plain text file is represented and edited by showing all the characters as they are present in the file. The only characters usable for 'mark-up' are the control characters of the used character set; in practice this is newline, tab and form feed. The most commonly used character set is ASCII, especially recently, as plain text files are more often being used for programming and configuration, and less frequently for documentation (e.g. detailed instructions, user guides) than in the past.

b. Proposed Solution

Features

- Open – Opening a presaved files.
- New – Creating a new file.
- Save and Print – To save text files and converting it into pdf.
- Copy,Cut, Paste,Undo,Redo and Select All – For copying and cutting a string of text , pasting and to select the entire text of the file.
- Format- Changing text colour ,background colour and font of the text.

c. Deliverables

This application provides simple uses of a text editor . It provides a writing space for the user and some operations like creating a new file , opening a new file , saving a file and operations like cut, paste and select all,etc and formatting options to change the appearance of text and editor.

CHAPTER 2

Project Description

2.1System Specification

2.1.1 Hardware requirements

Processor - Intel Pentium 600 MHz

Memory – 256 MB

Storage – 106.1GB

2.1.2 Software Requirements

OPERATING SYSTEM : Windows XP/7/8/10

Compiler used : Java version 15

IDE : Netbeans

Front End : Java Swings , AWT

2.2 Methodology Used

Java

Java is a class-based, object-oriented programming language that is designed to have as few implementation dependencies as possible. It is a general-purpose programming language intended to let application developers write once, run anywhere (WORA),[16] meaning that compiled Java code can run on all platforms that support Java without the need for recompilation.[17] Java applications are typically compiled to bytecode that can run on any Java virtual machine (JVM) regardless of the underlying computer architecture. The syntax of Java is similar to C and C++, but has fewer low-level facilities than either of them. The Java runtime provides dynamic capabilities (such as reflection and runtime code modification) that are typically not available in traditional compiled languages.

Java was originally developed by James Gosling at Sun Microsystems (which has since been acquired by Oracle) and released in 1995 as a core component of Sun Microsystems' Java platform. The original and reference implementation Java compilers, virtual machines, and class libraries were originally released by Sun under proprietary licenses. As of May 2007, in compliance with the specifications of the Java Community Process, Sun had relicensed most of its Java technologies under the GNU General Public License. Oracle offers its own HotSpot Java Virtual Machine, however the official reference implementation is the OpenJDK JVM which is free open source software and used by most developers and is the default JVM for almost all Linux distributions.

JFC

JFC is an abbreviation for Java Foundation classes that includes a set of characteristics to build Graphical User Interfaces (GUI) and add wealthy graphical characteristics and interactivity to Java apps. Java Swing belongs to the Java Classes Foundation (JFC).

Feature	Description
Swing GUI	Includes everything from buttons to split panes to tables.

Components	Many components are capable of sorting, printing, and drag and drop, to name a few of the supported features.
Pluggable Look-and-Feel Support	The look and feel of Swing applications is pluggable, allowing a choice of look and feel. For example, the same program can use either the Java or the Windows look and feel. Additionally, the Java platform supports the GTK+ look and feel, which makes hundreds of existing look and feels available to Swing programs. Many more look-and-feel packages are available from various sources.
Accessibility API	Enables assistive technologies, such as screen readers and Braille displays, to get information from the user interface.
Java 2D API	Enables developers to easily incorporate high-quality 2D graphics, text, and images in applications and applets. Java 2D includes extensive APIs for generating and sending high-quality output to printing devices.
Internationalization	Allows developers to build applications that can interact with users worldwide in their own languages and cultural conventions. With the input method framework developers can build applications that accept text in languages that use thousands of different characters, such as Japanese, Chinese, or Korean.

Java Swing

Swing Framework includes a number of classes that provide GUI components that are more strong and versatile than AWT. Swing is an official Java GUI tool kit published by Sun Microsystems that offers the look and feel of contemporary Java GUI. It is used to build Java's graphical user interface.

In javax.swing package and its sub-packages, swing classes are described.

Swing Components Are Lightweight

With very few exceptions, Swing components are lightweight. This means that they are written entirely in Java and do not map directly to platform-specific peers. Thus, lightweight components are more efficient and more

flexible. Furthermore, because lightweight components do not translate into native peers, the look and feel of each component is determined by Swing, not by the underlying operating system. As a result, each component will work in a consistent manner across all platforms.

Swing Supports a Pluggable Look and Feel

Swing supports a pluggable look and feel (PLAF). Because each Swing component is rendered by Java code rather than by native peers, the look and feel of a component is under the control of Swing. This fact means that it is possible to separate the look and feel of a component from the logic of the component, and this is what Swing does. Separating out the look and feel provides a significant advantage: it becomes possible to change the way that a component is rendered without affecting any of its other aspects. In other words, it is possible to "plug in" a new look and feel for any given component without creating any side effects in the code that uses that component. Moreover, it becomes possible to define entire sets of look-and-feels that represent different GUI styles. To use a specific style, its look and feel is simply "plugged in." Once this is done, all components are automatically rendered using that style. Pluggable look-and-feels offer several important advantages. It is possible to define a look and feel that is consistent across all platforms. Conversely, it is possible to create a look and feel that acts like a specific platform. For example, if you know that an application will be running only in a Windows environment, it is possible to specify the Windows look and feel. It is also possible to design a custom look and feel. Finally, the look and feel can be changed dynamically at run time.

System development life cycle (SDLC)

This is also known as Classic Life Cycle Model (or) Linear Sequential Model (or) Waterfall Method. This has the following activities.

System/Information Engineering and Modeling

As software is always of a large system (or business), work begins by establishing requirements for all system elements and then allocating some subset of these requirements to software. This system view is essential when software must interface with other elements such as hardware, people and other resources. System is the basic and very critical requirement for the existence of software in any entity. So if the system is not in place, the system should be engineered and put in place. In some cases, to extract the maximum output, the system should be re-engineered and spruced up. Once the ideal system is engineered or tuned, the development team studies the software requirement for the system.

Software Requirement Analysis

This is also known as feasibility study. In this phase, the development team visits the customer and studies their system. They investigate the need for possible software automation in the given system. By the end of the feasibility study, the team furnishes a document that holds the different specific recommendations for the candidate system. It also includes the personnel assignments, costs, project schedule, and target dates. The requirements gathering process is intensified and focused specially on software. To understand the nature of the program(s) to be built, the system engineer ("analyst") must understand the information domain for the software, as well as required function, behavior, performance and interfacing. The essential purpose of this phase is to find the need and to define the problem that needs to be solved.

System Analysis and Design

In this phase, the software development process, the software's overall structure and its nuances are defined. In terms of the client/server technology, the number of tiers needed for the package architecture, the database design, the data structure design etc are all defined in this phase. A

software development model is created. Analysis and Design are very crucial in the whole development cycle. Any glitch in the design phase could be very expensive to solve in the later stage of the software development. Much care is taken during this phase. The logical system of the product is developed in this phase.

Code generation

The design must be translated into a machine-readable form. The code generation step performs this task. If the design is performed in a detailed manner, code generation can be accomplished without much complication. Programming tools like Compilers, Interpreters, and Debuggers are used to generate the code. Different high level programming languages like C, C++, Pascal, and Java are used for coding. With respect to the type of application, the right programming language is chosen.

Testing

Once the code is generated, the software program testing begins. Different testing methodologies are available to unravel the bugs that were committed during the previous phases. Different testing tools and methodologies are already available. Some companies build their own testing tools that are tailor made for their own development operations.

Maintenance

Software will definitely undergo change once it is delivered to the customer. There are many reasons for the change. Change could happen because of some unexpected input values into the system. In addition, the changes in the system could directly affect the software operations. The software should be developed to accommodate changes that could happen during the post implementation period.

2.3 Modules and their Descriptions

File

New

It is used for opening a blank page of the text editor.

Open

The open feature allows the user to open a saved file.

Save

The save feature allows the user to save their work and can be reused again by opening the saved file using ‘Open’.

Print

Printing and converting text file to pdf files.

Exit

The exit feature allows the user to exit/close the current working editor.

Edit

Copy

The cut feature allows the user to cut a statement written on the screen.

Cut

The cut feature allows the user to cut a statement written on the screen.

Paste

The Paste feature allows the user to paste the statement the user has already cut.

Select All

This feature allow the user to select all the text written on the editor’s window.

Redo

Redo the Text we had undone

Undo

To remove the recently typed text.

Format

Background Colour

For changing the background color of the text editor

Text Colour

For changing the text color .

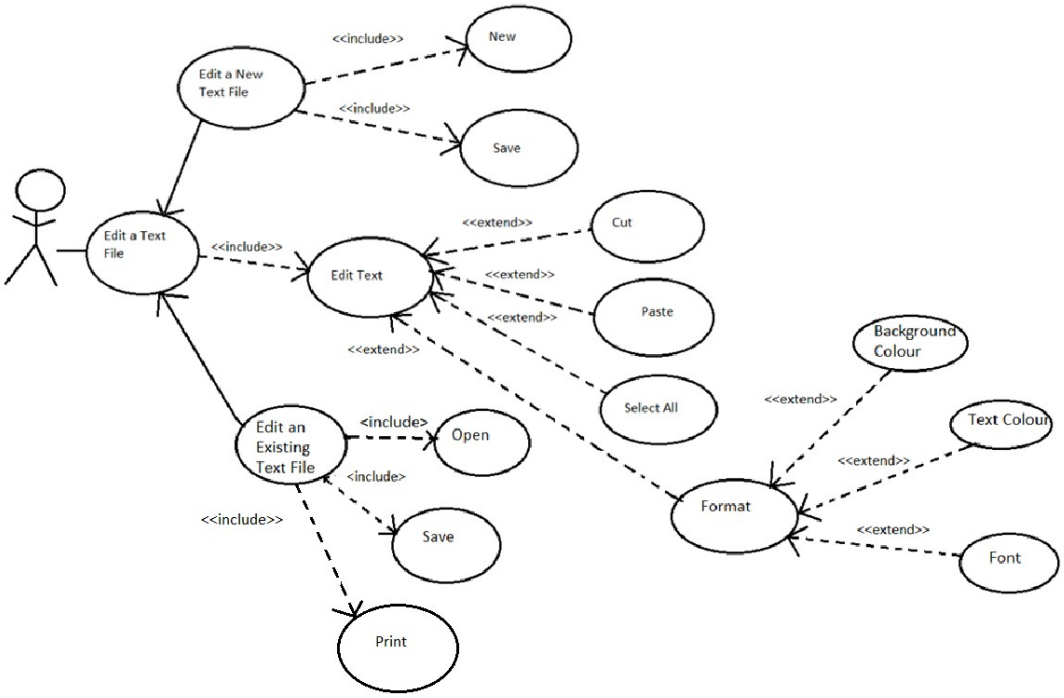
Font

For changing the style of the text, size and type (plain ,bold and italic).

CHAPTER 3

Functionalities

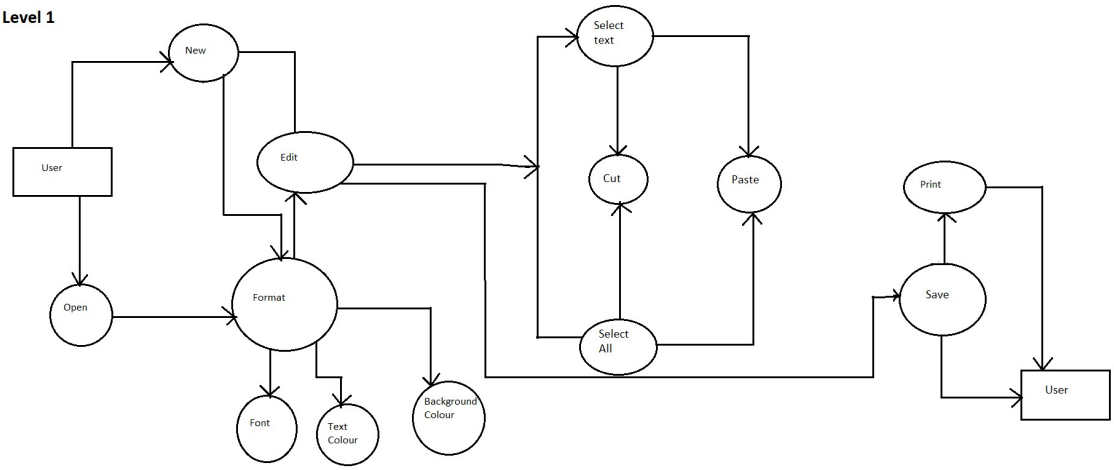
3.1Use Case Diagram



3.2Data Flow Diagram

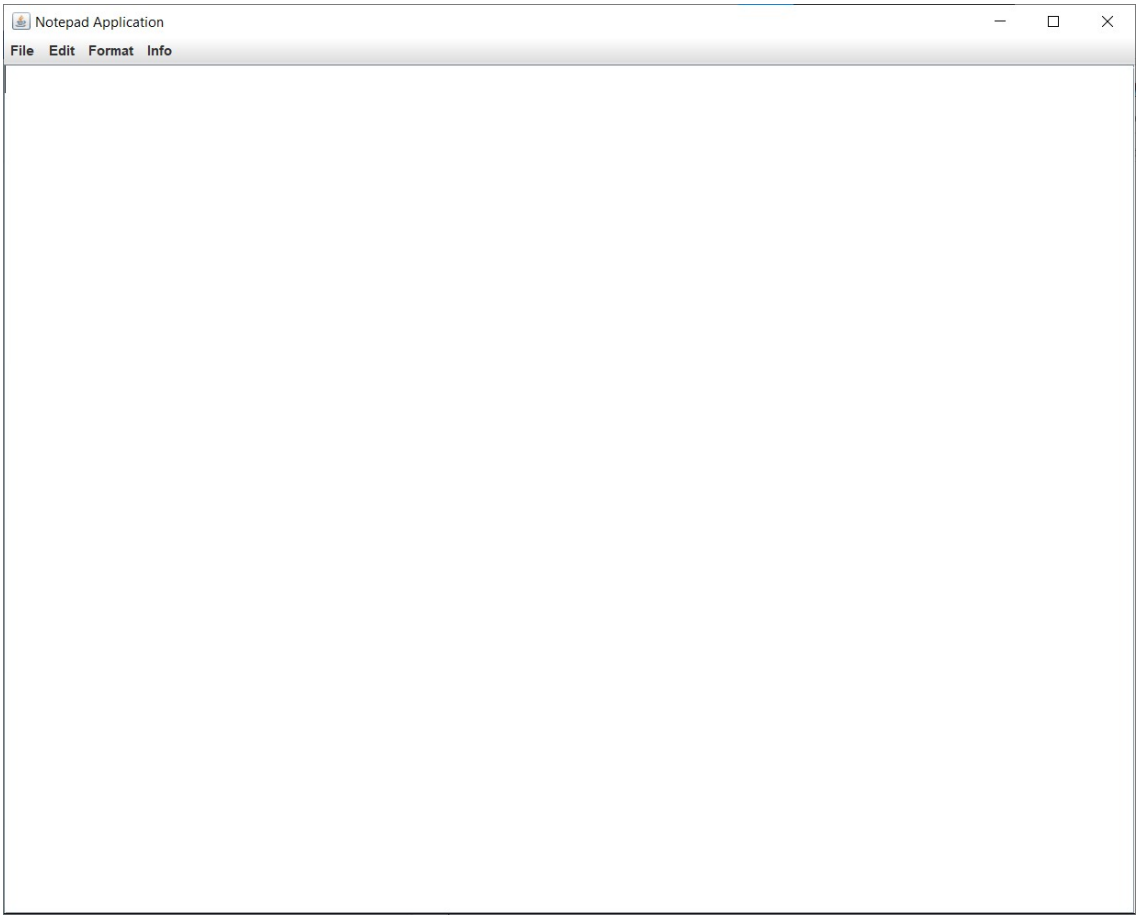
Level 0

Level 1

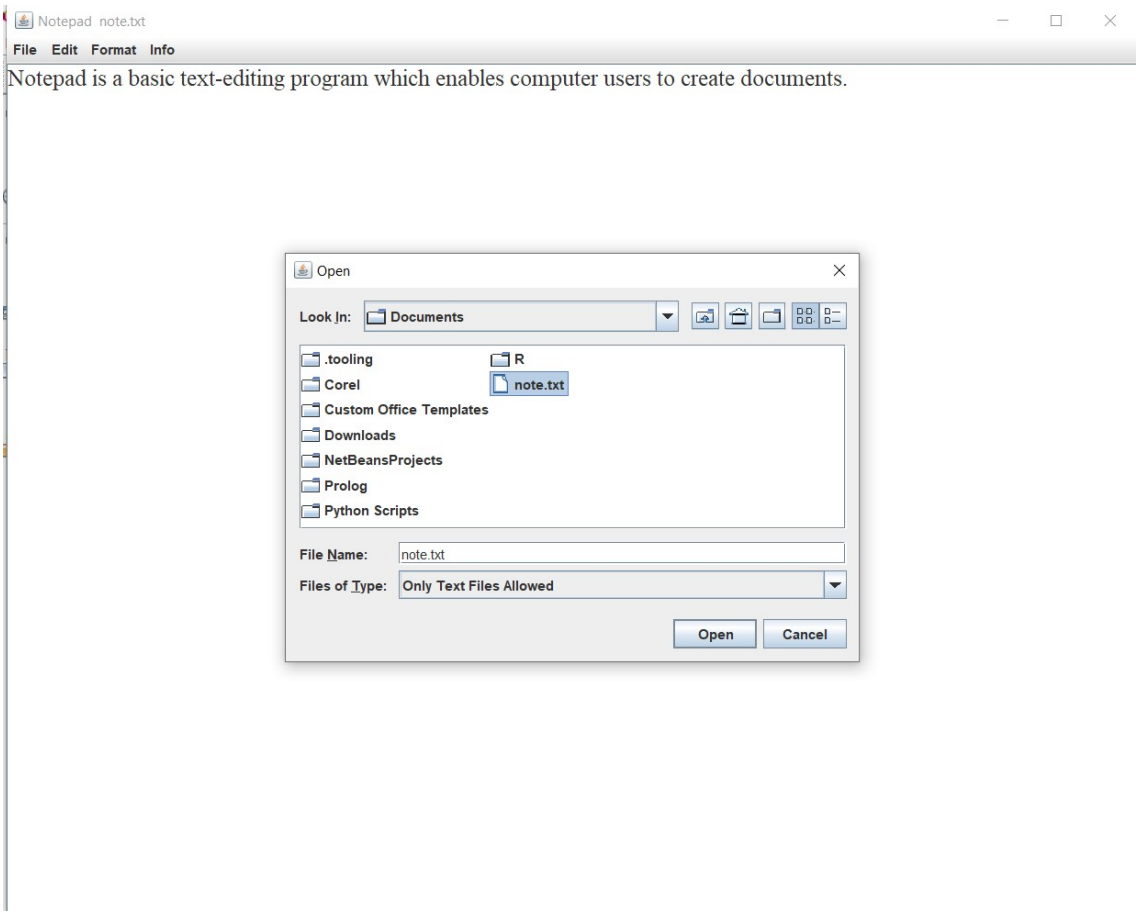


3.3 Screenshots of the running code

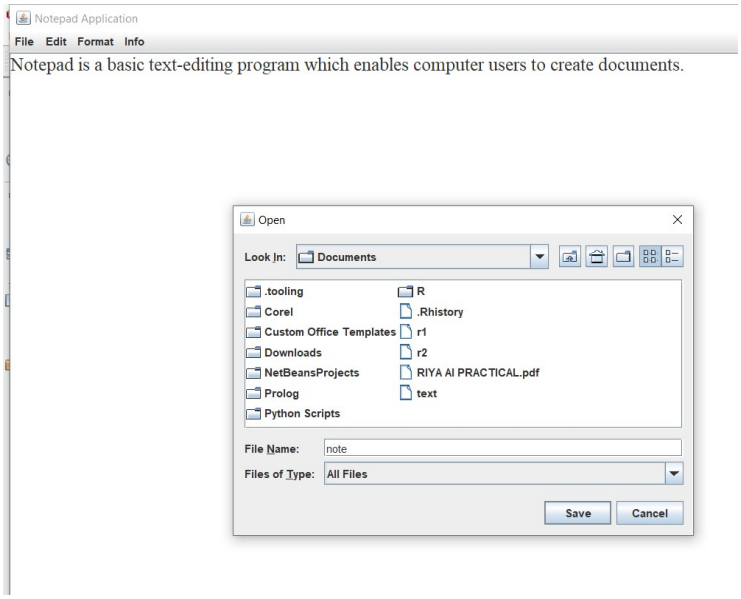
1. Main Window



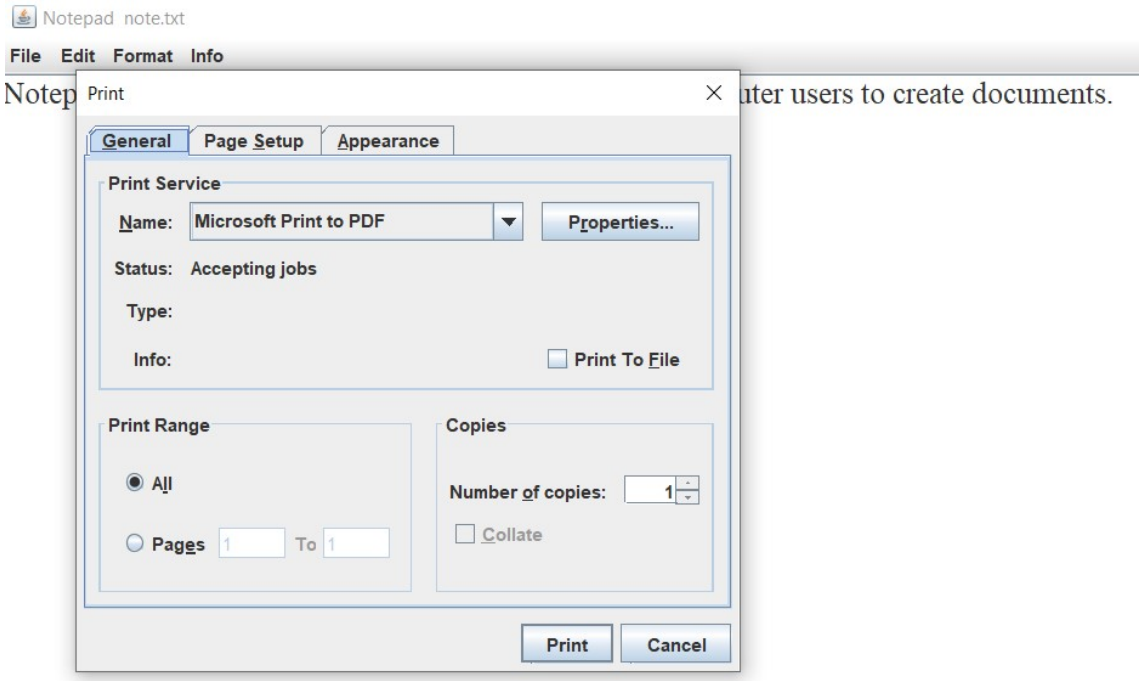
2. Opening a file



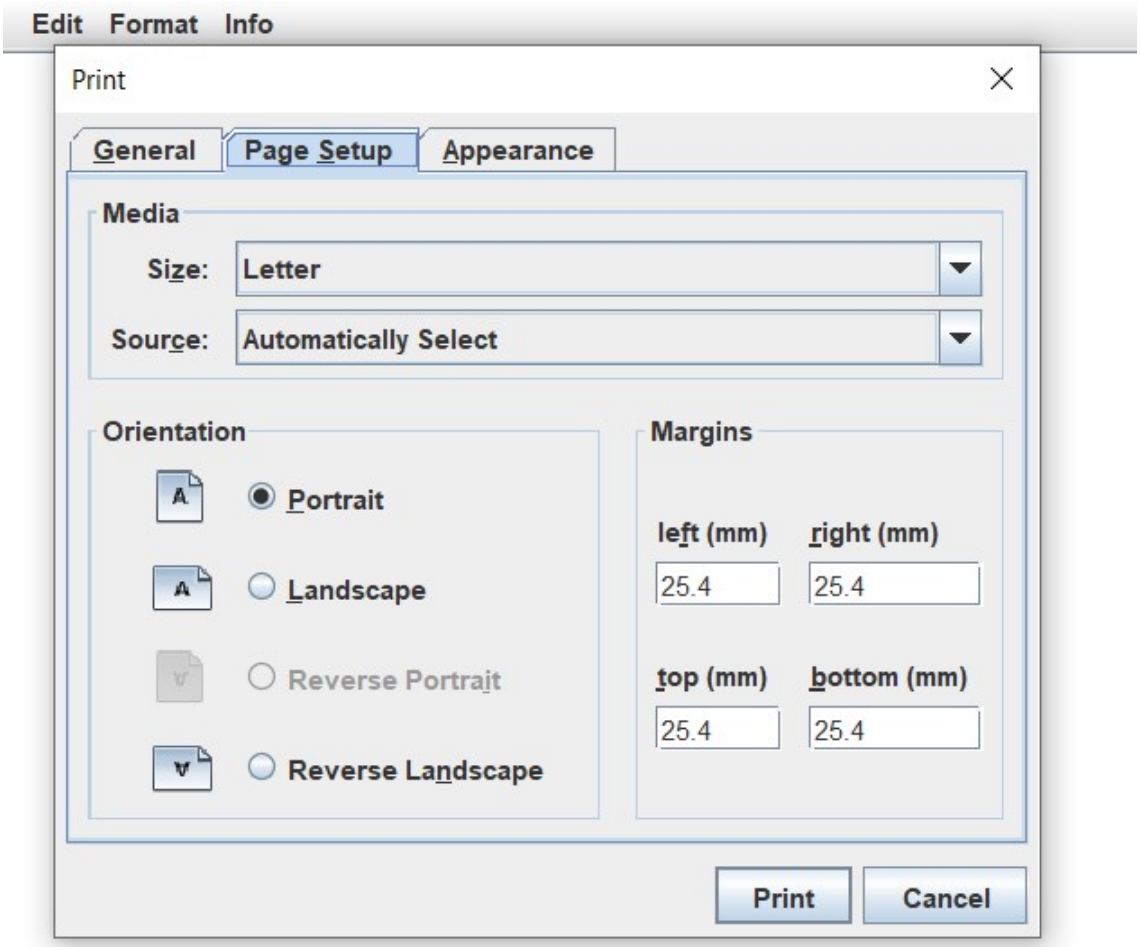
3. Saving a file


Notepad Application
File Edit Format Info
Notepad is a basic text-editing program which enables computer users to create documents.
Open
Look In:
Documents
.tooling
Corel
Custom Office Templates
Downloads
NetBeansProjects
Prolog
Python Scripts
R
.Rhistory
r1
r2
RIYA AI PRACTICAL.pdf
text
File Name:
note
Files of Type:
All Files
Save
Cancel

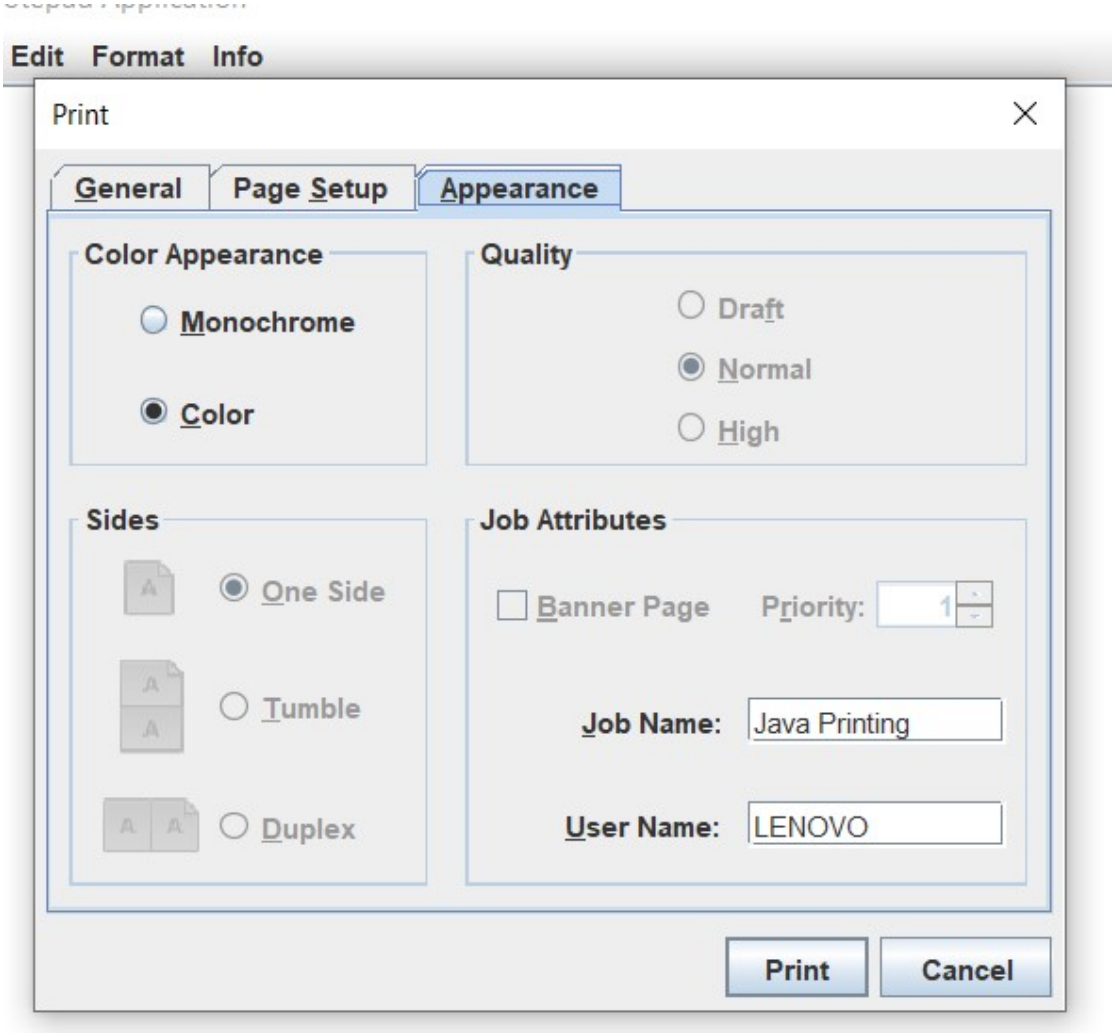
4. Print



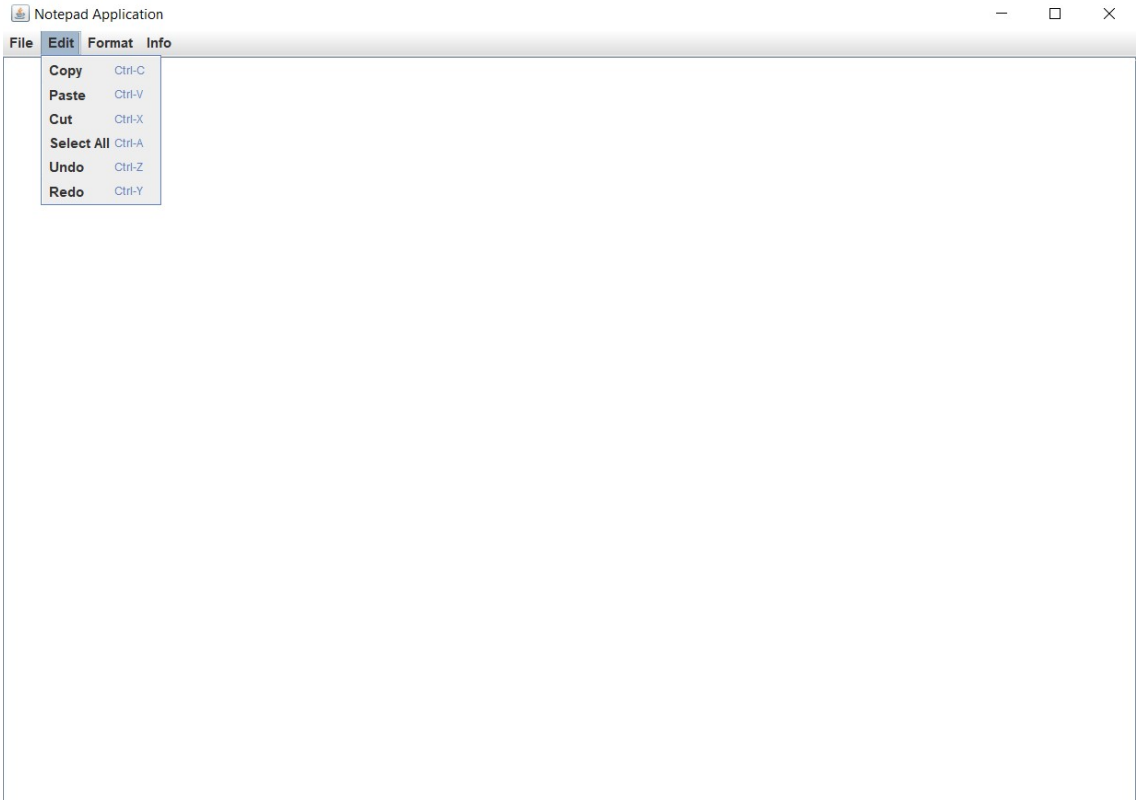
5. Print Page-setup



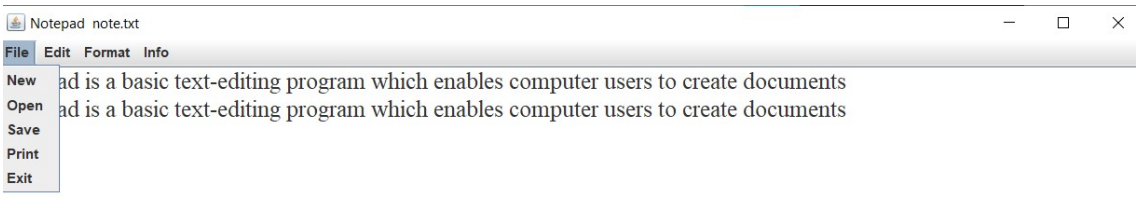
6. Print appearance



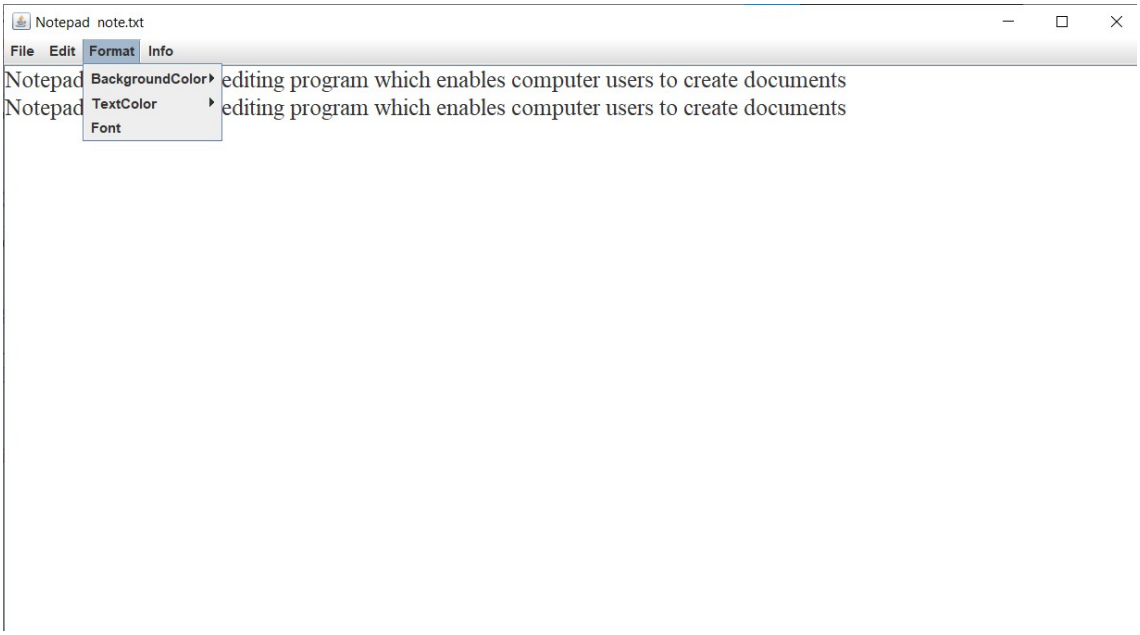
7. Edit Options



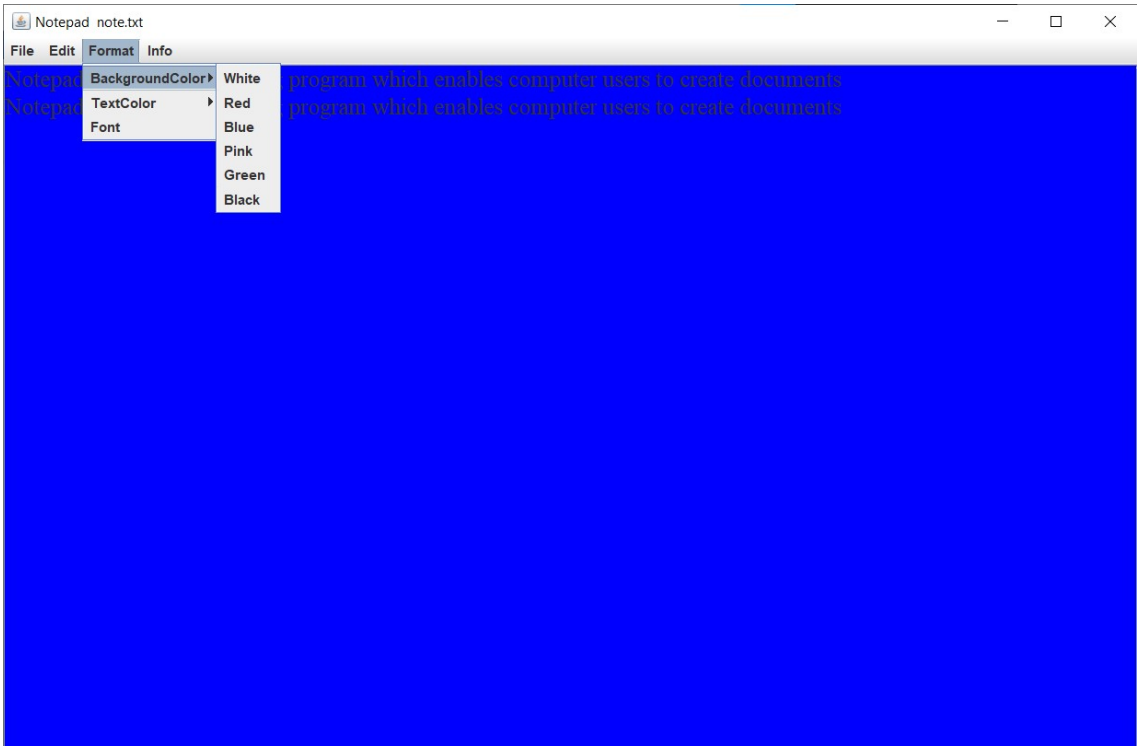
8. File Options



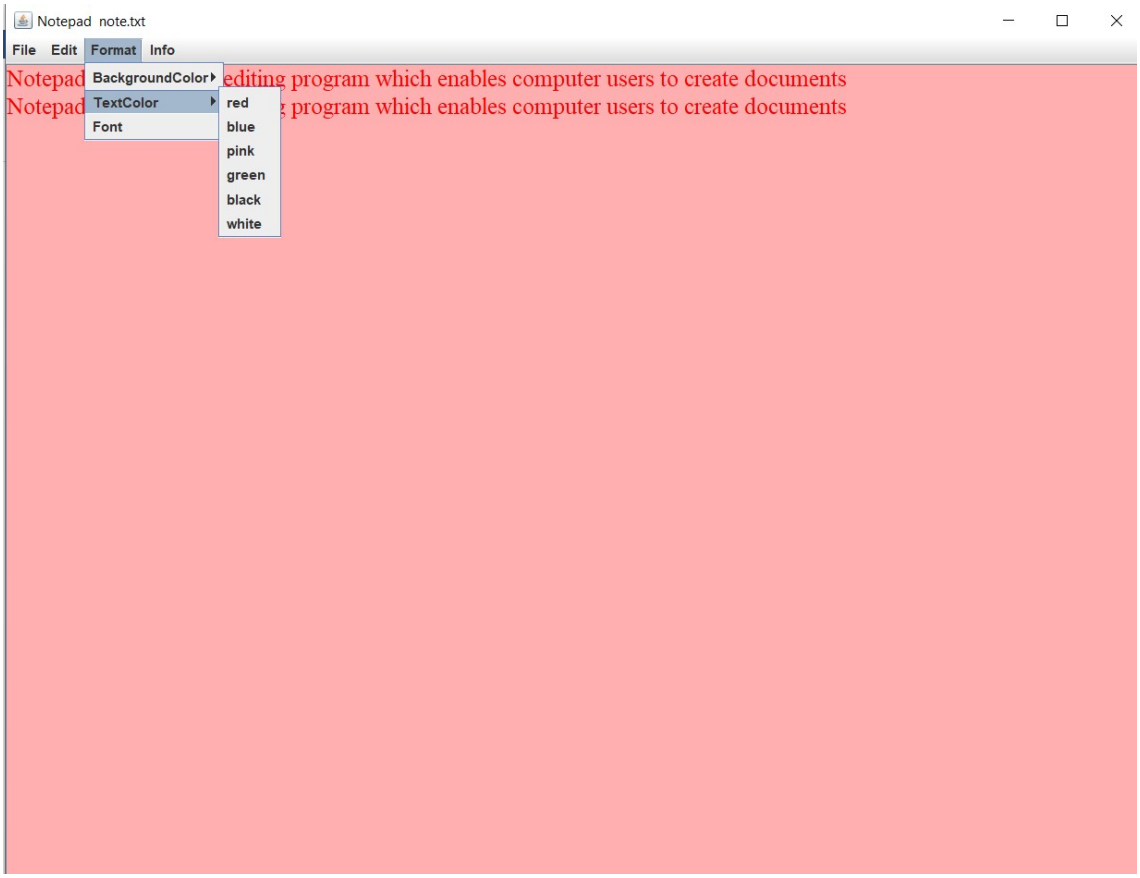
9. Format Options



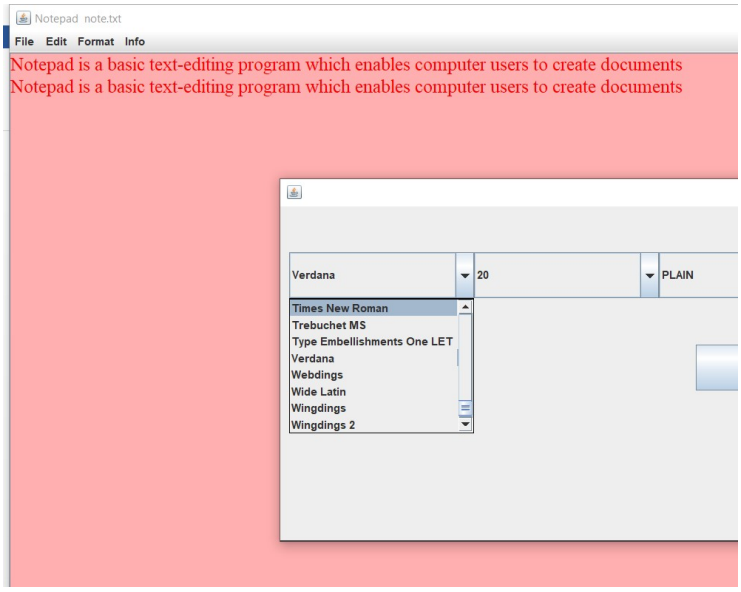
10. Background Colour Option



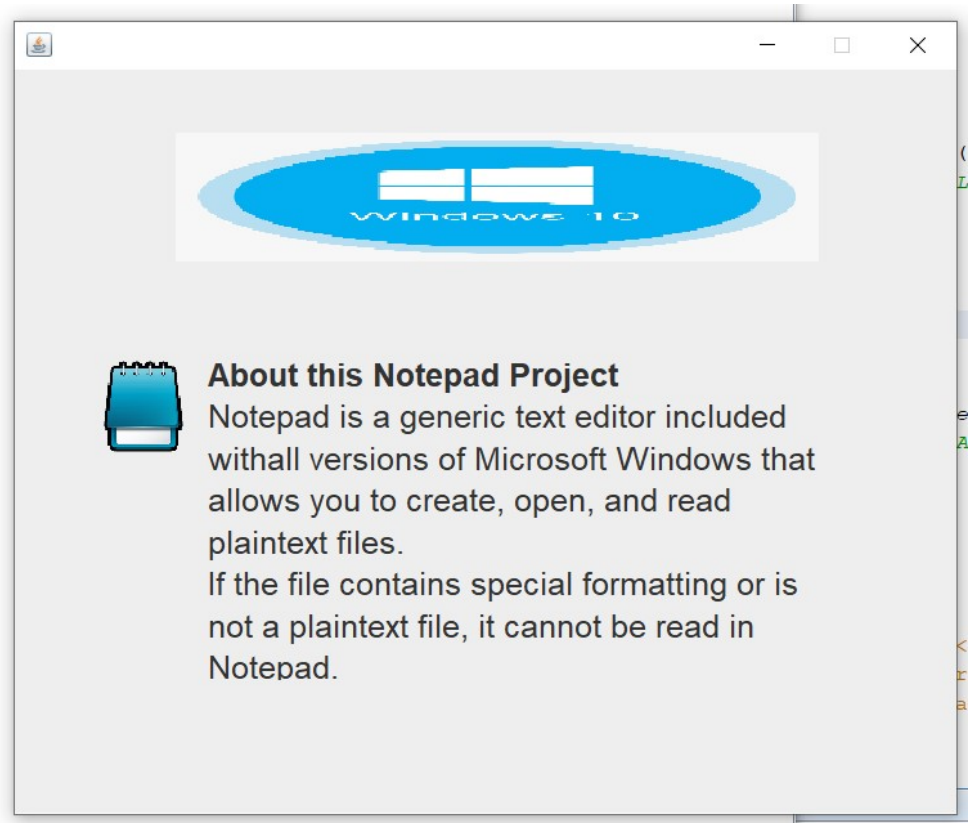
11.Text colour option


Notepad note.txt
File Edit Format Info
BackgroundColor
TextColor
Font
red
blue
pink
green
black
white
Notepad editing program which enables computer users to create documents
Notepad program which enables computer users to create documents

12. Font


Notepad note.txt
File Edit Format Info
Notepad is a basic text-editing program which enables computer users to create documents
Notepad is a basic text-editing program which enables computer users to create documents
Verdana
20
PLAIN
Times New Roman
Trebuchet MS
Type Embellishments One LET
Verdana
Webdings
Wide Latin
Wingdings
Wingdings 2

13.Info


About this Notepad Project
Notepad is a generic text editor included withall versions of Microsoft Windows that allows you to create, open, and read plaintext files.
If the file contains special formatting or is not a plaintext file, it cannot be read in Notepad.

CHAPTER 4

Conclusion and Future Scope

4.1 Conclusion

This project that I undertook is a simple implementation of Java Swing features and File Handling to created a plain text editor . We use plain text editor in our daily lifes to write an essay ,to take notes etc. Text editors make our work easier as we can read the saved notes , we don't need pen and paper to write down.

4.2 Future Scope

We already have plenty of advanced text editors such as Notepad++, Sublim , VS Code , etc. With the advancement of technologies we'll have more advanced text editors which will make human work easier .

Bibliography

The referred books were

- Java 2-Complete reference (9th edition) by Herbert Schildt.

And the following websites were also used for reference

- https://docs.oracle.com/javase/7/docs/api/java/awt/datatransfer
- https://www.javatpoint.com/event-handling-in-java
- https://www.tutorialspoint.com/java/io/java_io_file.htm
- https://docs.oracle.com/javase/7/docs/api/java/io/PrintWriter.html