


Introduction:	3
Sitemap, mockups, wireframe& requirements:	3
sitemap:	4
Mockups:.....	5
wireframes:	6
The characteristics of the appropriate algorithm base are:	9
Accuracy - The steps are completely clear (specific).	9
Uniqueness - the results of each step are unambiguously defined and relying solely on inputs, and therefore the results of the previous steps.	9
Collecting information entries and requirements of the Ministry of Finance	9
Outputs - screens that benefit Ministry of Finance employees based on their requirements	9
Generality - matches inputs to outputs	9
Effective to be effective without problems	9
Clarity - Define each step clearly and unambiguously	9
The appropriate algorithm rule properties are:	9
Programming steps:	24
Styles of programming:	25
Characteristics of the foremost forms of programming:.....	27
Common options of integrated development environments.....	29
Java Debugging with Eclipse	29
starting the Debugger	30
Uses of IDE:	30
Using IDE for development of applications contrasted and not.....	31
importance of a coding standard.....	31
Debugging in application	31
Standard code	32
Empty space	33
Comment code.....	33
Variable declarations:	33
Test Plan.....	33
Reference	34





complaints and suggestions page ○ ○ C

Type  title

text

send Back



Fill Data

Ministry of Finance

Month

Select Month

Spending

Earning

Save

Back



Monthly Reports

Ministry of Finance

Month	Spendings	Earning
<div> <div>Insert Data</div> <div>Back</div> </div>		

In general, it is the definition of the problem and the requirements for solving it with determining the data type in the form of variables and a default table for the shape of the output, then design it and write the code and read a line by line from start to end or design a flowchart using engineering shapes to clarify it and then convert it to code after tracking the main goal of it to solve Complex problems. Rocking might be near JFC (Foundation Java Classes), graphical user interface (graphical user interface) and AWT (summary window toolkit) it's a powerful and versatile useful program for graphic building in Java that wants to use Swings. Swing Framework contains a large collection of parts that allow for a high level of customization and provide deserving functionality, and are used to form window-based applications. Lightweight Java Hammock Parts Module, Free Platform, Powerful Parts Supply Like Tables, Scroll Panels, Buttons, List, Person Color, etc. With Eclipse for Java (Swing) and Windows Builder, to embrace SWT designers and Swing designer halves, it's very easy to configure Java GUI applications while not paying much of your time typing code. Use the application's visual design and layout tools to form direct forms for complex windows; Java code will be generated by adding controls for dragging and dropping misuse, adding event handlers to special controls, modifying the various properties of controls that use a feature editor, and internationalizing and adding your application to a large extent. Window Builder was created as a plug-in for the varied Eclipse (RAD, RSA, MyEclipse, JBuilder, etc.). The plugin creates an abstract syntax tree (AST) for navigation in the ASCII text file and uses GEF to show and manage the visual presentation.

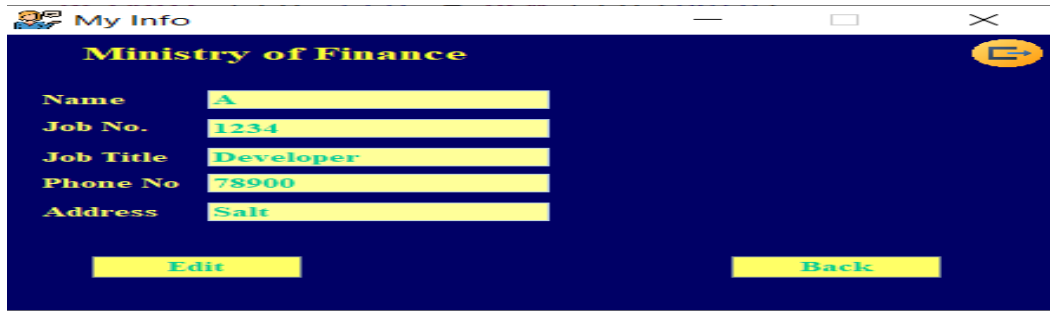


[illegible]

- [illegible]

Create the frame:

My Info:



I used many class to make My Info page like :

before any thing should use import to use any class

- JFrame : Used to add a window
- JPanel: Create a new JPanel with specified layout
- JTextField: enter the text box Job No, name, Job title, Phone, Address
- String: (job no)
- String result list & File: to read from new file and to write on new file
array list array result: to store split elements like [0,1,2,3,4] [A,1234, Developer, 78900, Salt]

```
String path = "C:\\MinistryOfFinanceFiles\\Login.txt"
```

Launch the application:

- I used object oriented
- `frame.setVisible(true)` to become visible
- `frame.setLocationRelativeTo(null)` to center the JFrame
- `frame.setResizable(false)` to disable the maximize button
- `addActionListener(new ActionListener()`

Create the frame:

from design window and source

- Try & catch Java Exceptions (throw an error)
- For statement to read file and read new file foreach to read line by line and split array list by if [0] for Name [1] for job No [2] for Job title [3] for Job no [4] for address. Like `if(resultList[1].equals(jobNo)textFieldName.setText(resultList[0])`
- setTitle to put title
- setIconImage(`Toolkit.getDefaultToolkit().getImage(MainPage.class.getResource("/images/money.png"))`) : from file I saved in it many images .
- setBond
- setDefaultCloseOperation(`JFrame.EXIT_ON_CLOSE`) to close
- setBackground, setBorder, setLayout
- Mouse pressed To return to the screen login page
- Action event to lock the existing screen and open the required
- setHorizontalAlignment(`SwingConstants.CENTER`) to setting frame like color and place in center



- setFont, JLabel to (password and Job No) , JButton, setFont to button setting.
 - In this screen the data will be filled automatically after enter login and you can edit Phone NO ,Address by the flowing screen by press Edit button .



Edit My Info:



I used many class to make Edit My Info page like :

before any thing should use import to use any class

- JFrame : Used to add a window
- JPanel: Create a new JPanel with specified layout
- JTextField:to enter the text box
- String: for (Address), (path to place storage files), (phone)
- File: to read from new file and to write on new file
- array list & array result: to store split elements like [0 ,1 ,2 ,3]

String path = "C:\\MinistryOfFinanceFiles\\Login.txt"

Launch the application:

- **I used object oriented**
- Try & catch Java Exceptions (throw an error)
- frame.setVisible(**true**) to become visible
- frame.setLocationRelativeTo(**null**) to center the JFrame
- frame.setResizable(**false**) to disable the maximize button
- addActionListener(**new** ActionListener())

Create the frame:

from design window and source

- Try & catch Java Exceptions (throw an error)
- For statement to read file and red new file foreach to read line by line and split



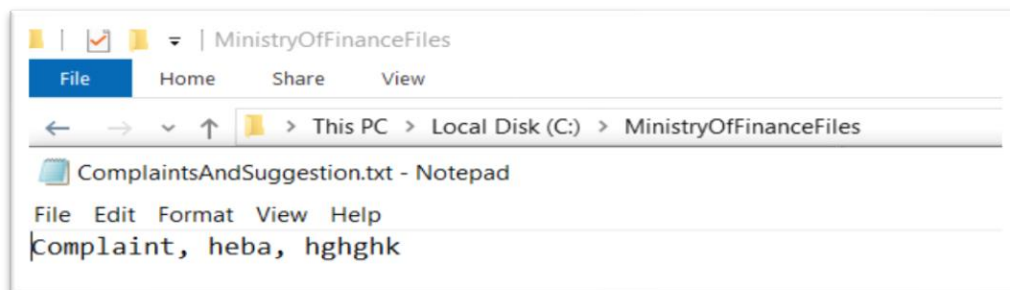
Complaints&Suggestions:



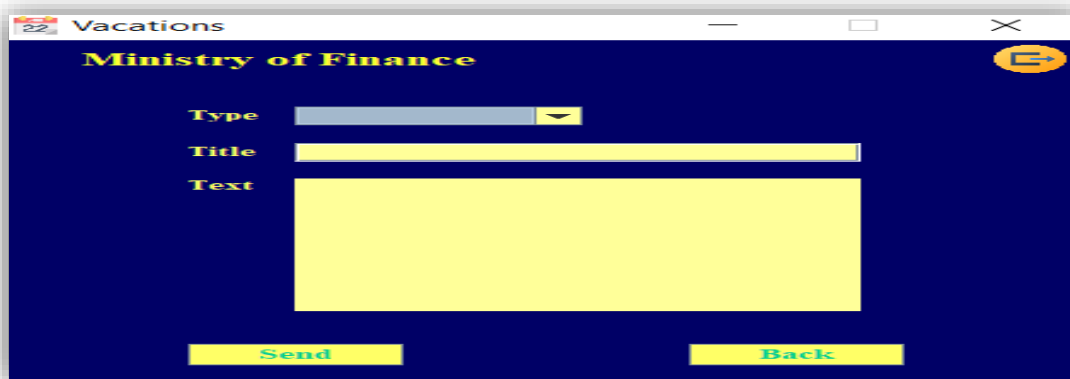
Create the frame:

from design window and source

- Set Title: to put title
- setIconImage (Toolkit.getDefaultToolkit().
getImage(MainPage.class.getResource("/images/money.png"))) : from file I saved in it many images .
- setBond
- setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE) to close
- setBackground,setBorder,setLayout
- setFont, JLabel to (password and Job No) , JButton, setFont to button setting
- Mouse pressed To return to the screen login page
- Action event to lock the existing screen and open the required.
- Try & catch Java Exceptions (throw an error)
- writer in file ("%s, %s, %s%n", type, title, text, "\n")) new line 8 spaces between them



Vacations:



I used many class to make vacations page like :

before any thing should use import to use any class

- JFrame : Used to add a window
- JPanel: Create a new JPanel with specified layout
- JTextField:to enter the text box
- JComboBox: to select type of complains & suggestions

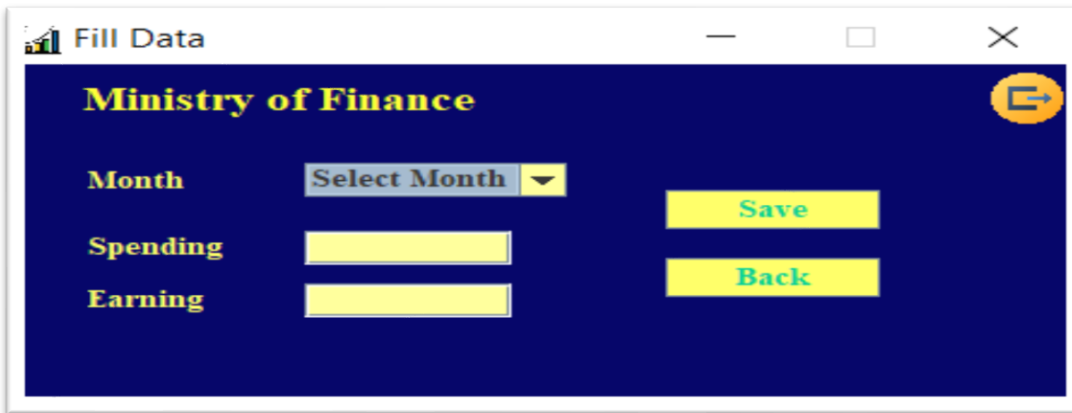


-



- Action event to lock the existing screen and open the required so, **set** to put and give but **get** to take.

Fill Data:



I used many class to make Fill Data page like :

before any thing should use import to use any class

- JFrame: Used to add a window
- JPanel: Create a new JPanel with specified layout
- JTextField: to enter the text box
- JComboBox: to select a month
- JLabel
- JButton: save & back
- String: "Select Month", "January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", "December".
- File: new file("C:\\MinistryOfFinanceFiles\\ComplaintsAndSuggestion.txt");
- File writer: to write it

Launch the application:

- I used **object oriented**
- Try & catch Java Exceptions (throw an error)
- frame.setVisible(**true**) to become visible
- frame.setLocationRelativeTo(**null**) to center the JFrame
- frame.setResizable(**false**) to disable the maximize button.

Create the frame:

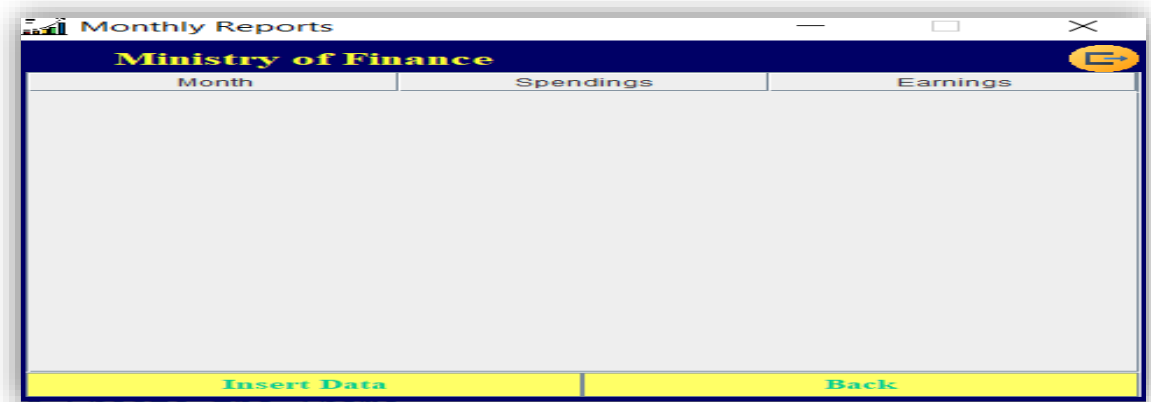
from design window and source

- Set Title: to put title
- setIconImage (Toolkit.getDefaultToolkit().
- getImage(MainPage.class.getResource("/images/money.png"))) : from file I saved in it many images .
- setHorizontalAlignment(SwingConstants.**RIGHT**)
- setBond
- setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE) to close

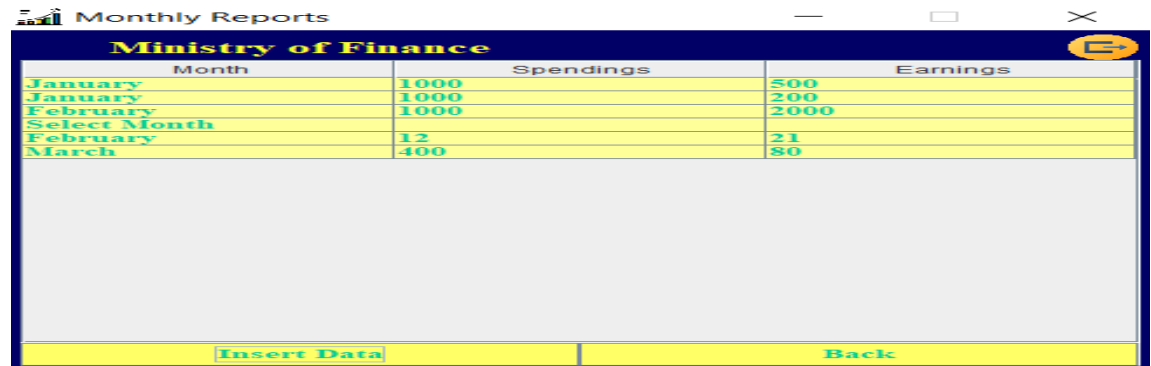


- setBackground,setBorder,setLayout
- setFont, JLabel , JButton, setFont to button setting.
- Mouse pressed To return to the screen login page
- Action event to lock the existing screen and open the required
- Try & catch Java Exceptions (throw an error)
- writer in file ("%s, %s, %s\n", type, title, text, "\n")) \n: new line. \t new tab 8spaces
- When you insert in Earning and Spending click on save then click on view report then click on insert data to see report that saved or back to return to report page screen.

Like it:



Monthly Report :



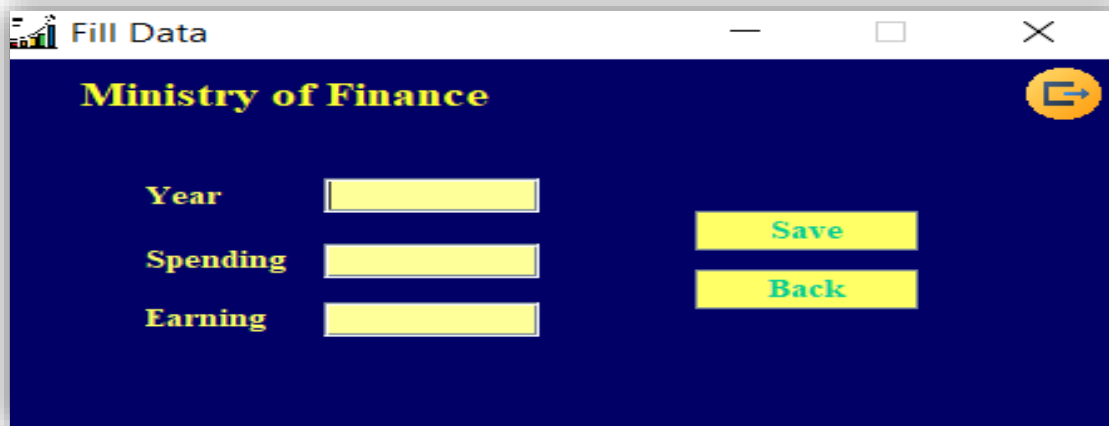
I used many class to make monthly reports page like :

before any thing should use import to use any class

- JFrame: Used to add a window
- JPanel: Create a new JPanel with specified layout
- JLabel
- JTable: to make table
- JButton: insert data & back
- String: ("C:\\MinistryOfFinanceFiles\\ComplaintsAndSuggestion.txt");



-

Fill Data:

I used many class to make Fill Data page like :

before any thing should use import to use any class

- JFrame: Used to add a window
- JPanel: Create a new JPanel with specified layout
- JTextField: to enter the text box
- JLabel
- JButton: save & back
- String: ("C:\\MinistryOfFinanceFiles\\ComplaintsAndSuggestion.txt");
- File: new file("C:\\MinistryOfFinanceFiles\\ComplaintsAndSuggestion.txt");

Launch the application:

- **I used object oriented**
- Try & catch Java Exceptions (throw an error)
- `frame.setVisible(true)` to become visible
- `frame.setLocationRelativeTo(null)` to center the JFrame
- `frame.setResizable(false)` to disable the maximize button.

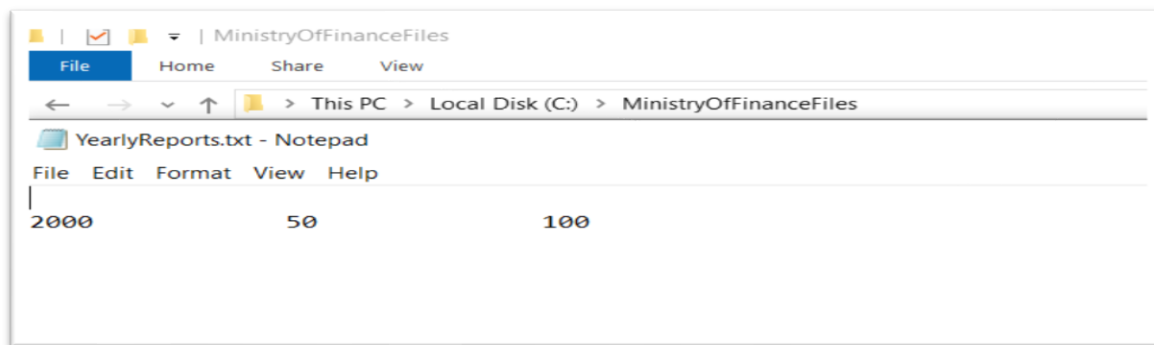
Create the frame:

from design window and source

- Set Title: to put title
- Set layout
- setIconImage (Toolkit.*getDefaultToolkit()*).
- getImage(MainPage.**class**.getResource("/images/money.png")) : from file I saved in it many images .
- setHorizontalAlignment(SwingConstants.**RIGHT**)
- setHorizontalAlignment(SwingConstants.**center**)
- setBond



- screen. Like it:





- 



4. Development & Implementation: the particular task of developing the package starts here with information recording occurring within the background. Once the package is developed, the stage of implementation comes in wherever the merchandise goes through a pilot study to visualize if it's functioning properly.
5. Testing: The testing stage assesses the package for errors and documents bugs if their area unit any.
6. Maintenance: Once the package passes through all the stages with none problems, it's to endure a maintenance method whereby it'll be maintained and upgraded from time to time to adapt to changes.

Styles of programming:

A programming paradigm is also an elementary type of building the structure and components of a program. The styles and capabilities of programming languages printed by their paradigms. as an example, a linguistic communication might entirely be designed to follow one paradigm (i.e. Haskell), whereas others can support multiple paradigms.

A Paradigm usually thought of as a philosophical or theoretical framework. This framework is then used as a result of the idea of a replacement approach of viewing problems.

This topic investigates the 3 major programming paradigms:

- Imperative (or procedural)
- Object oriented
- Event-driven


Main forms of programming


1. Procedural Programming-

Procedural language is a list of instructions telling a computer (a programming paradigm that uses a linear or top-down approach), Step-by-Step, what to do, in an order of how-to performance the first code to the

- Second and So forth which may contain loops. Procedural programming,
- languages include C, C++, Fortran, Pascal, these are very detailed and
- • takes tons of your time to put in writing. I.
- It relies on the programmer to provide a Solution to a problem or to
- provide an answer. It rotates around keeping code as Short as possible







- **Final result preview:** most IDE additionally build it simple to put in writing UI giving nice previews while not having to begin our application.

Debugging, in computer programming and engineering, may be a multistep method that involves distinguishing a drag, uninflected the supply of the matter, so either correcting the





