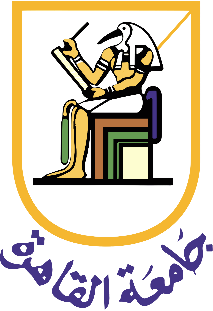
**Cairo University**

**Faculty of Computers and Artificial Intelligence**

**Software Engineering**

**Sent** to: Dr. Mohamed El-Ramly

CS251

* **Assignment: A1**
* **Task: T1 &T2**
* **Date: 2025/2/25**
* **Section: S14**
* **Team Programming Language: Java**
* **Team Leader Phone Number: +20 128 696 4627**
* **Name, IDs, and E-mails:**

|  |  |  |
| --- | --- | --- |
| Name | IDs | E-Mails. |
| Aly El-Deen Yasser Ali | 20231109 | ali.el.badry.747@gmail.com |
| Nagham Wael Mohamed | 20231189 | naghamw63@gmail.com |
| Fatema El-Zhraa Ahmed Mohamed El-Fiky | 20230280 | fatmaelfeky922@gmail.com |

Table of Content

**The Process of Learning 3**

Aly ElDeen Program3

Nagham Program 9

Fatema El-Zhraa Ahmed Program 14

**Low Code No Code Tool4**

Evaluation for LCNC Tools5

The potential of LCNC Tools5

What LCNC Tools Can Do 5

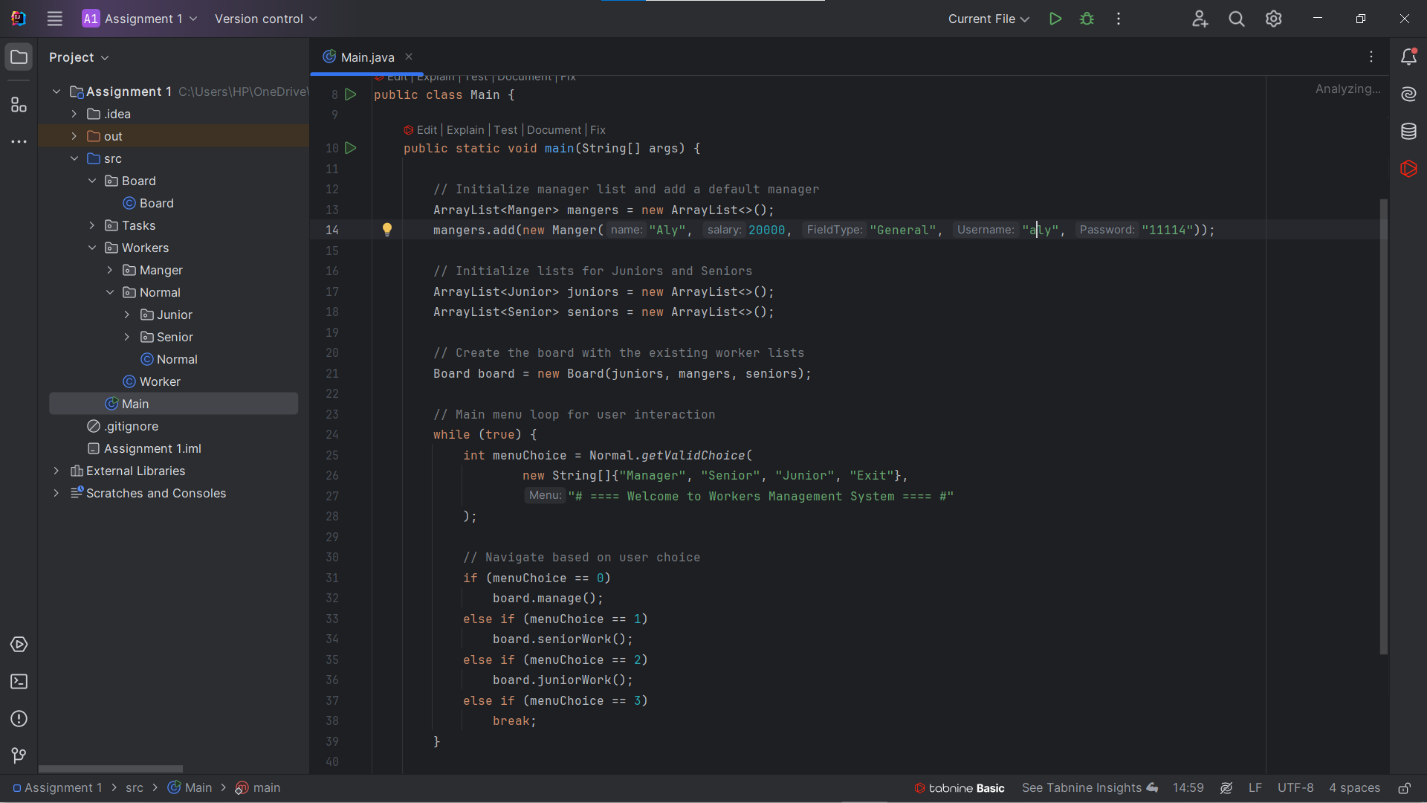
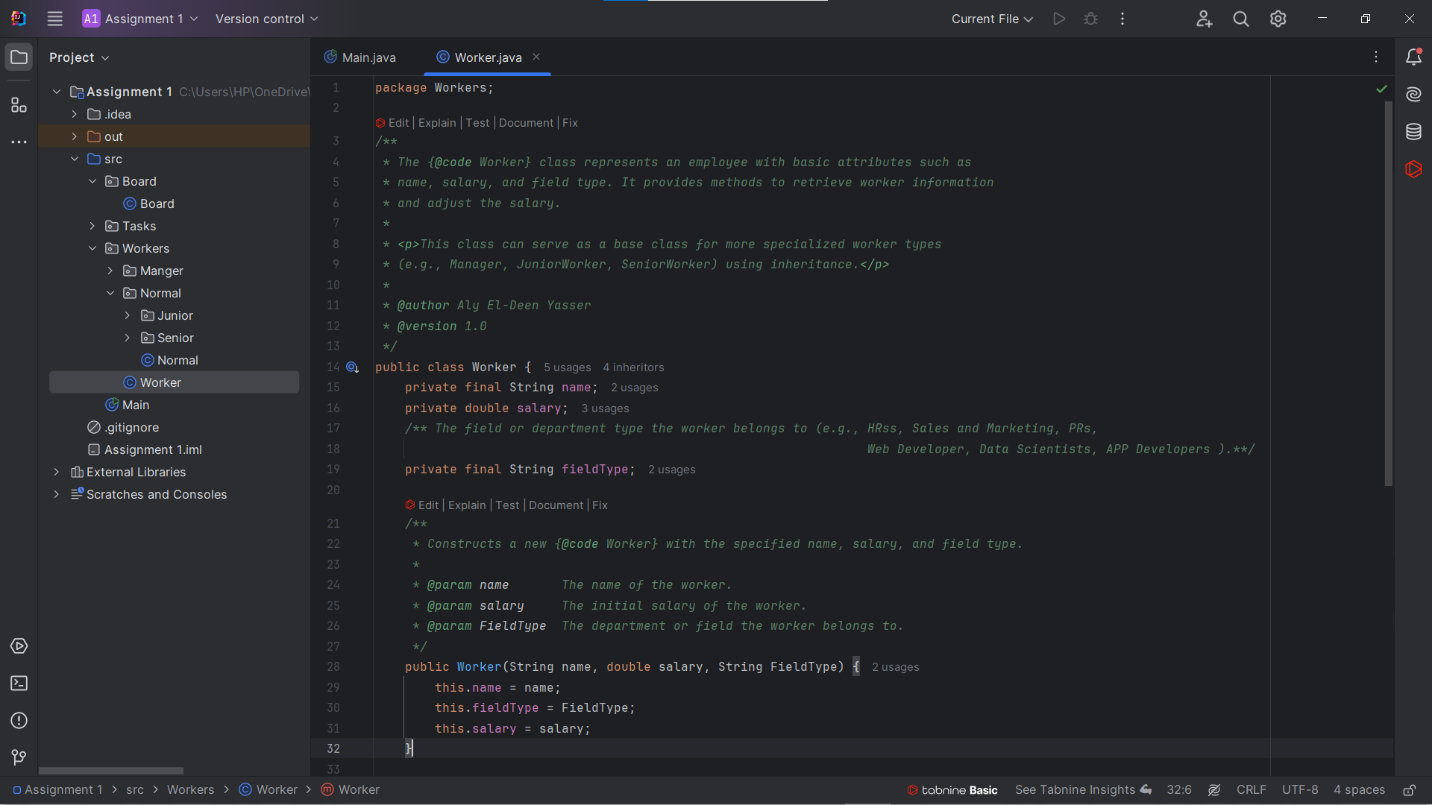
Benefits of LCNC Tools5

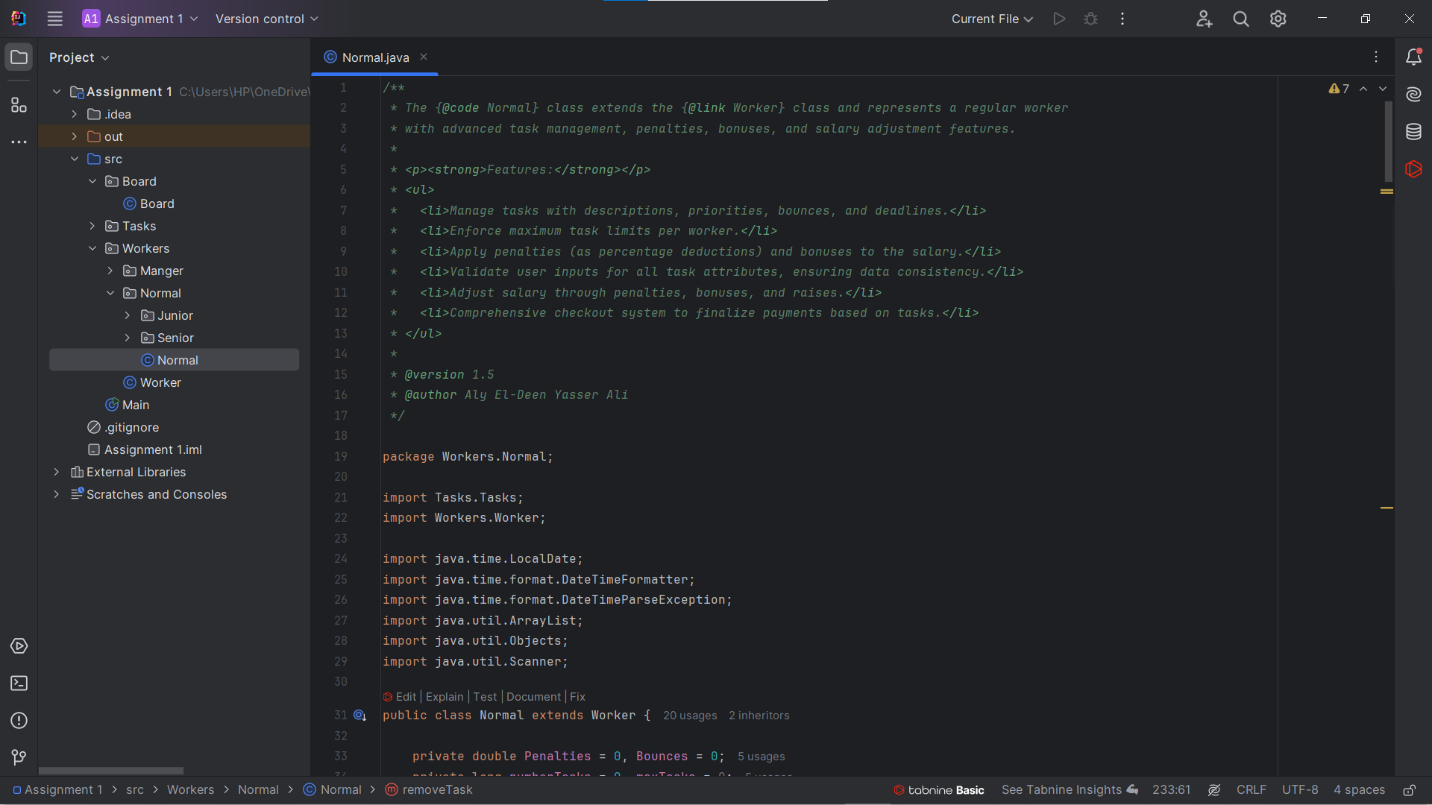
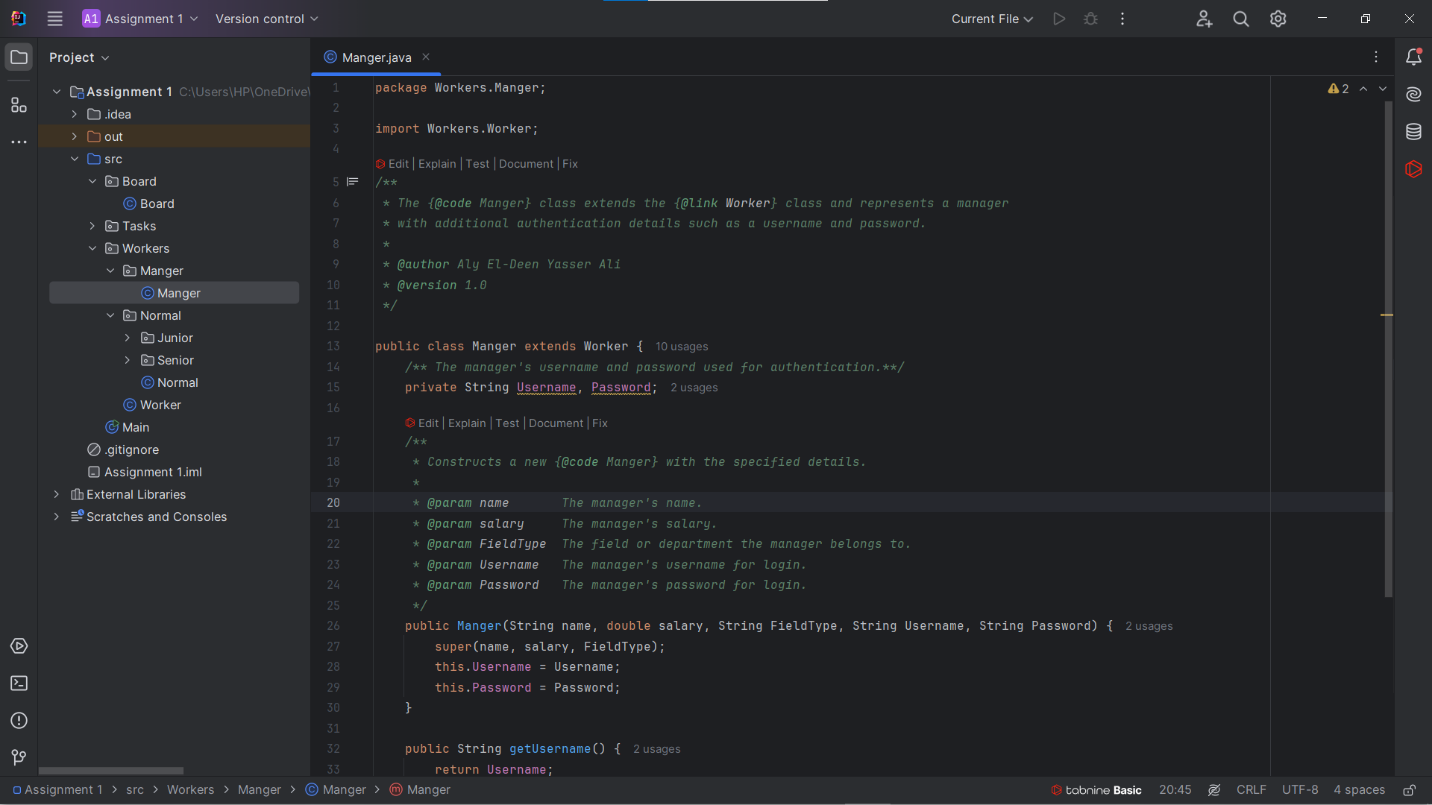
Will LCNC Tools Take the Job of Developers? 5

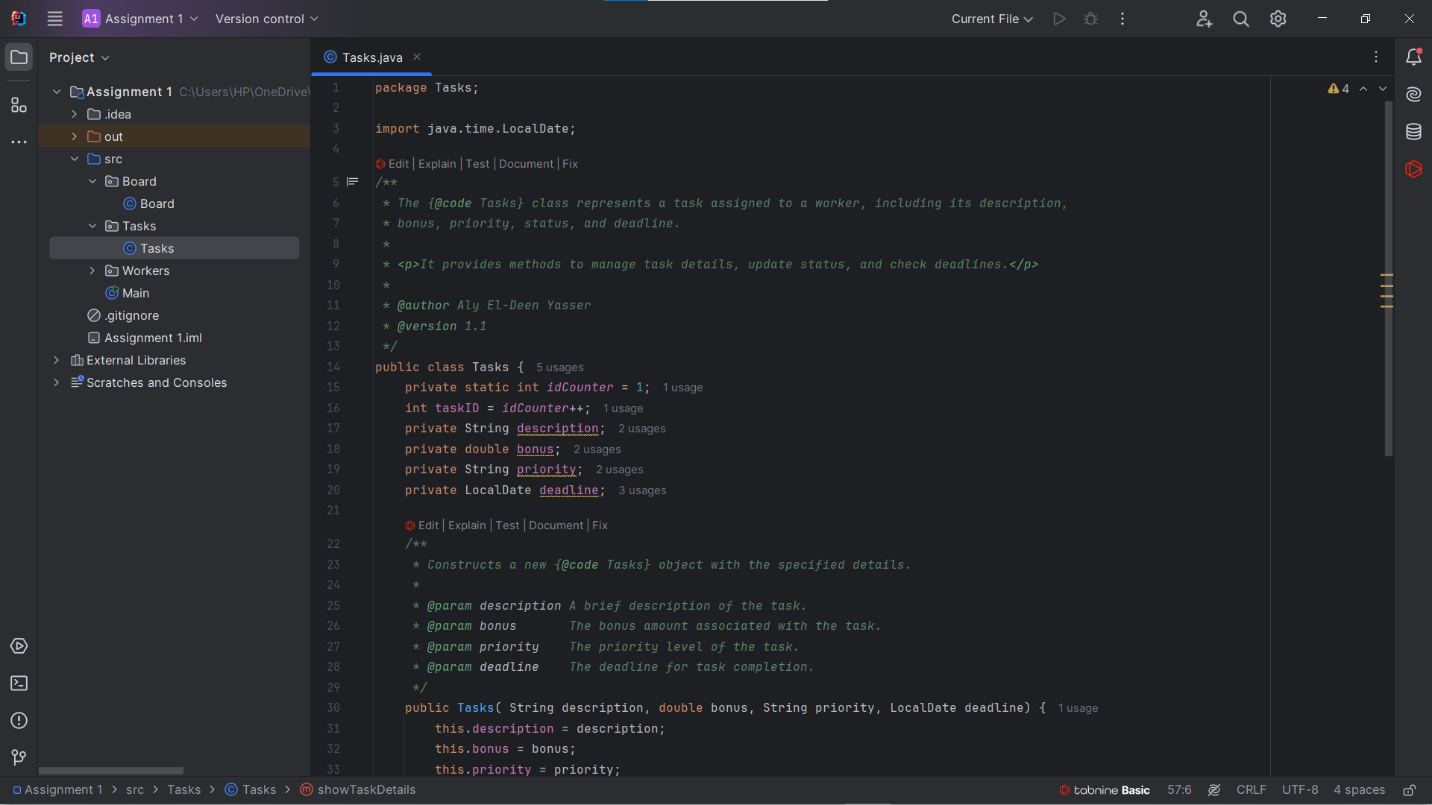
**Compare between The two LCNC Tools4**

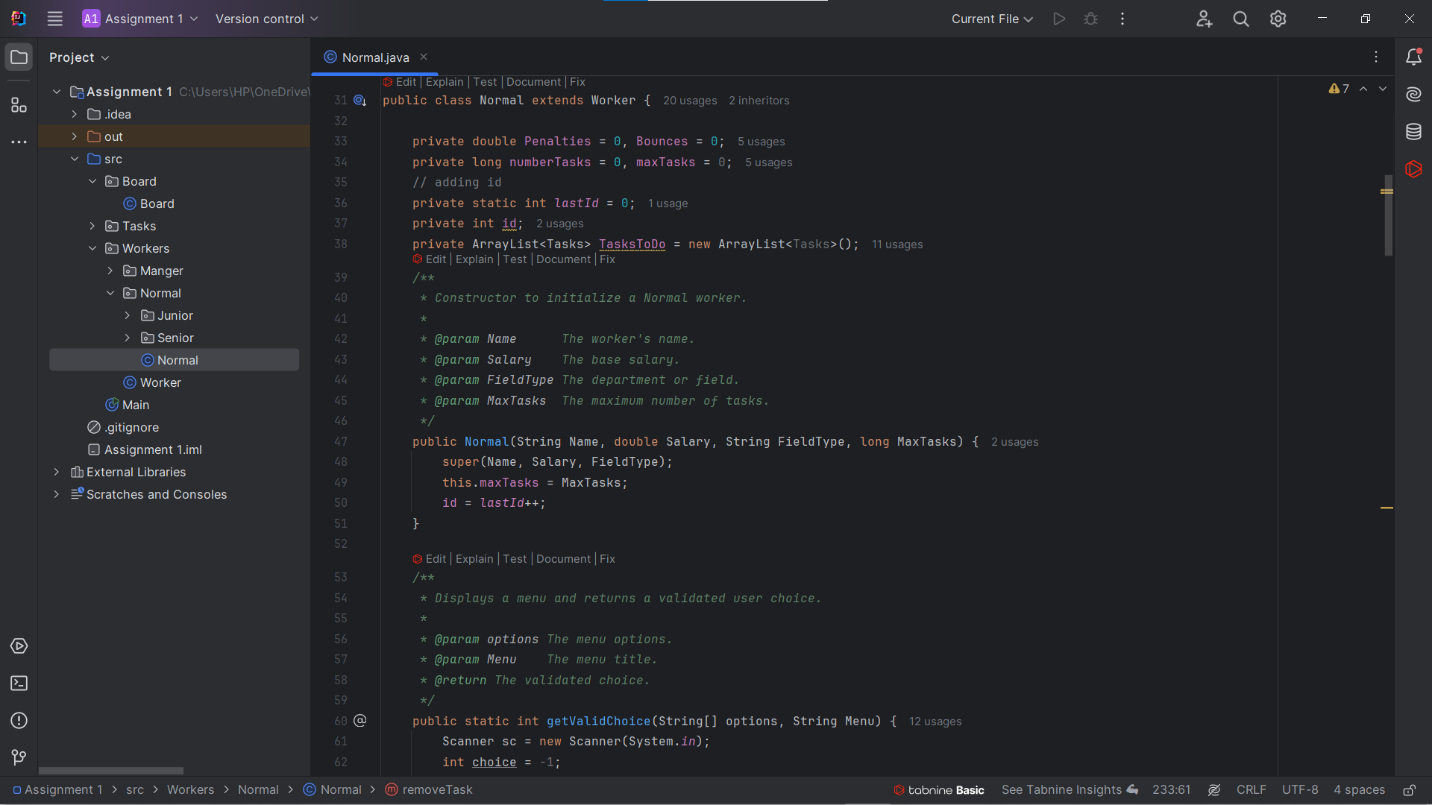
**The Process of Learning :**

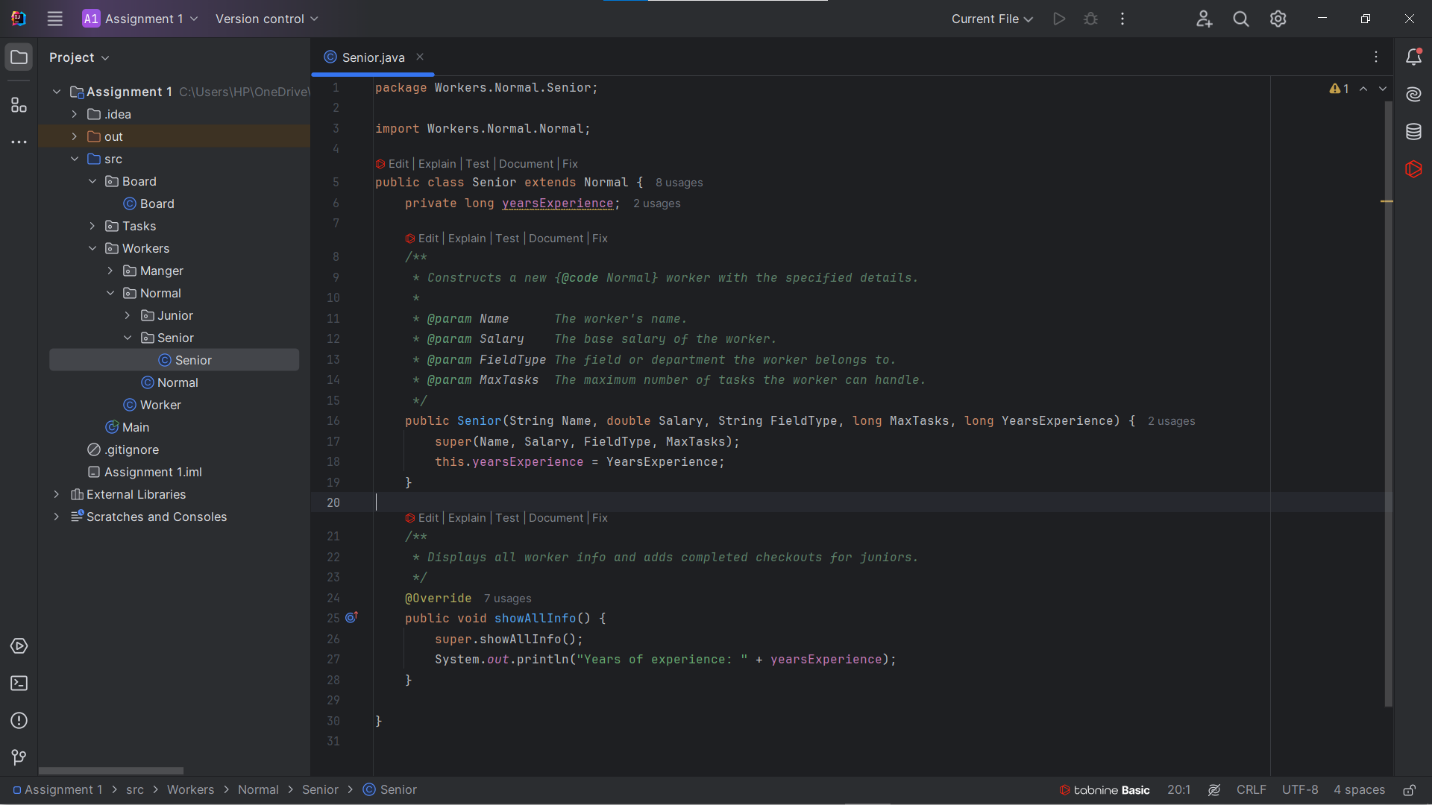
**Aly El-Deen Yasser Aly:**

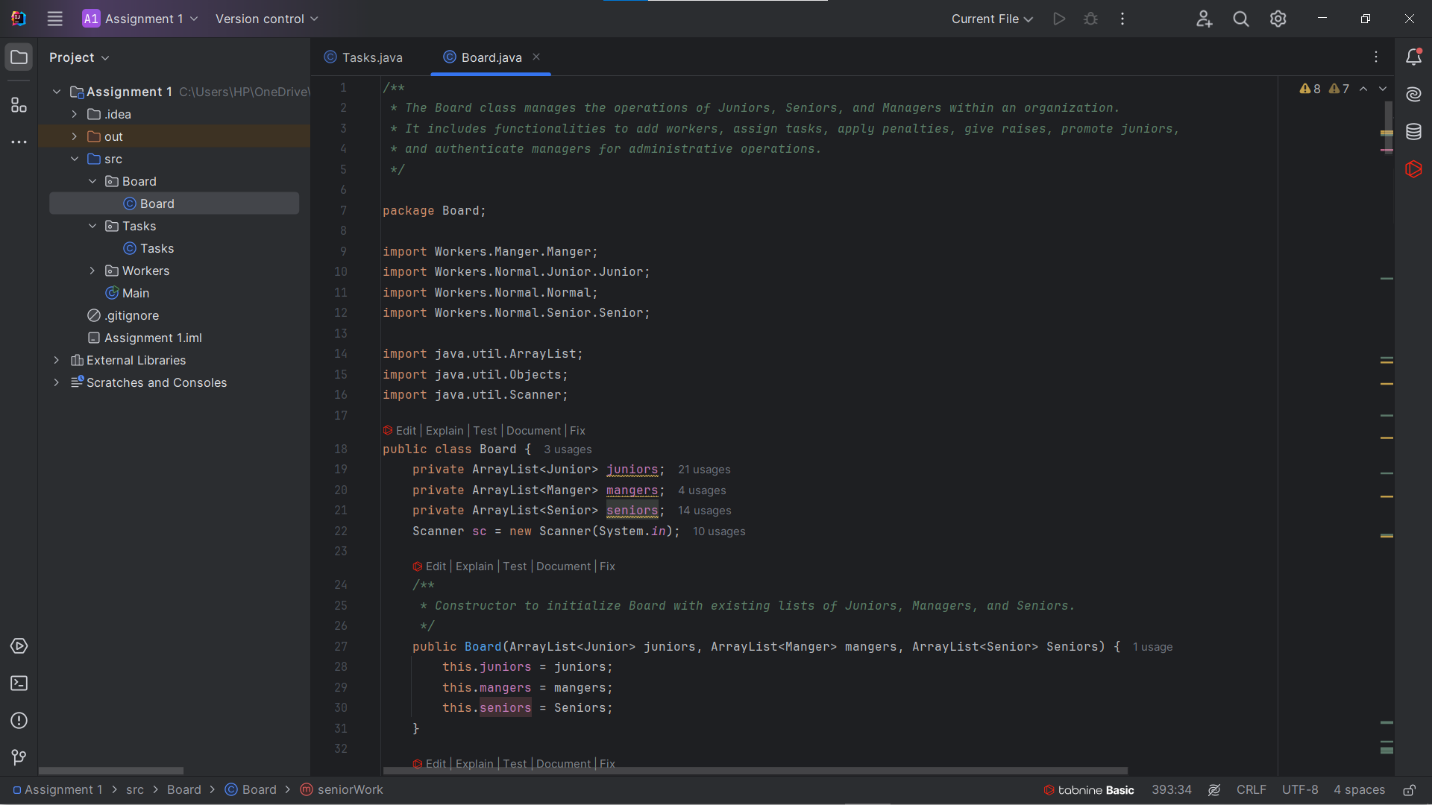
* Hours of Study: 18 hours
* Source of study https://youtube.com/playlist?list=PLCInYL3l2AajYlZGzU\_LVrHdoouf8W6ZN&si=tfRlF2iNxOtJehum
* Main Logic of code:
* import java.util.ArrayList;  
  import Workers.Manger.Manger;  
  import Workers.Normal.Junior.Junior;  
  import Workers.Normal.Normal;  
  import Workers.Normal.Senior.Senior;  
  import Board.Board;  
    
  public class Main {  
    
   public static void main(String[] args) {  
    
   // Initialize manager list and add a default manager  
   ArrayList<Manger> mangers = new ArrayList<>();  
   mangers.add(new Manger("Aly", 20000, "General", "aly", "11114"));  
    
   // Initialize lists for Juniors and Seniors  
   ArrayList<Junior> juniors = new ArrayList<>();  
   ArrayList<Senior> seniors = new ArrayList<>();  
    
   // Create the board with the existing worker lists  
   Board board = new Board(juniors, mangers, seniors);  
    
   // Main menu loop for user interaction  
   while (true) {  
   int menuChoice = Normal.*getValidChoice*(  
   new String[]{"Manager", "Senior", "Junior", "Exit"},  
   "# ==== Welcome to Workers Management System ==== #"  
   );  
    
   // Navigate based on user choice  
   if (menuChoice == 0)  
   board.manage();  
   else if (menuChoice == 1)  
   board.seniorWork();  
   else if (menuChoice == 2)  
   board.juniorWork();  
   else if (menuChoice == 3)  
   break;  
   }  
    
   System.*out*.println("\nThanks For Using Our Program");  
   }  
  }\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
* Images For the code:

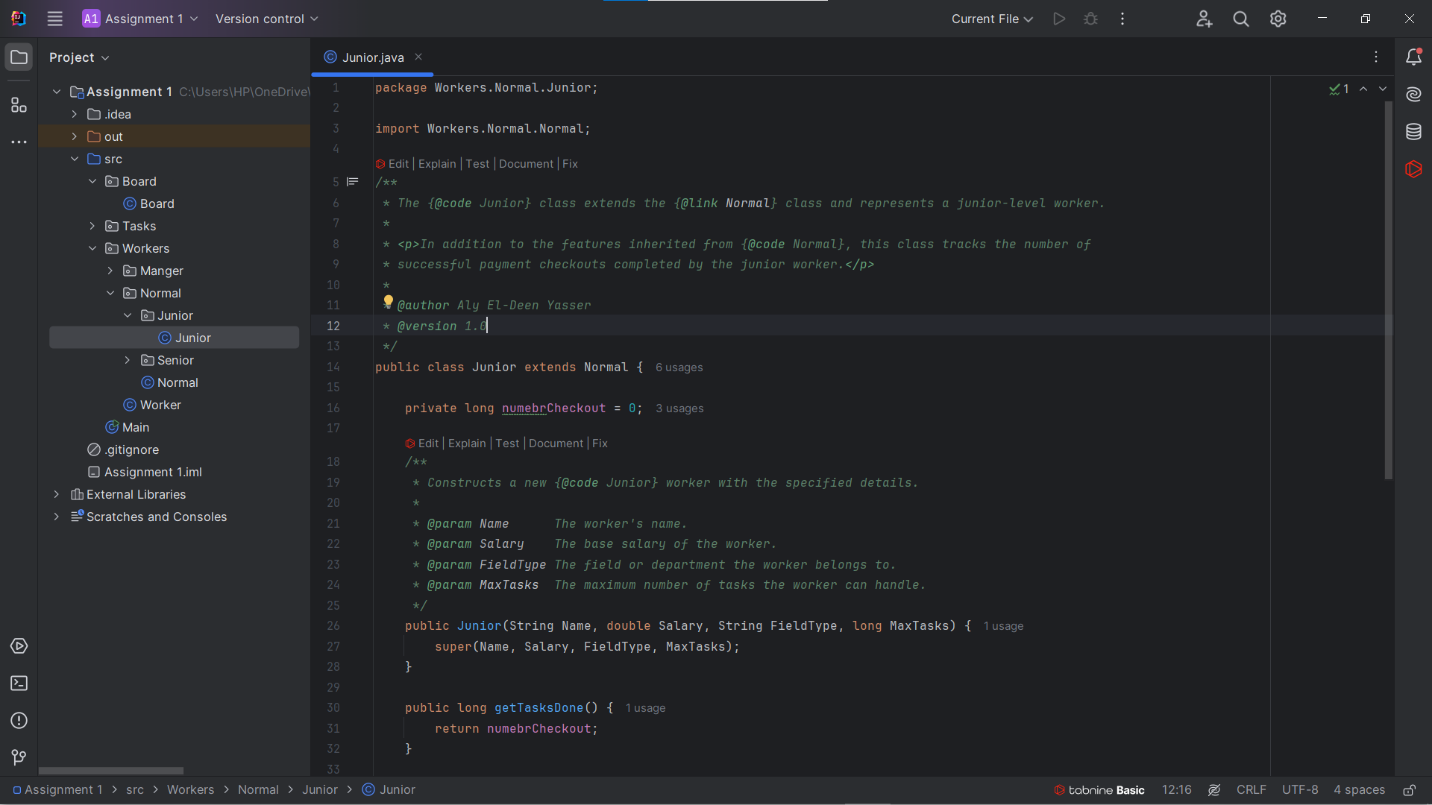












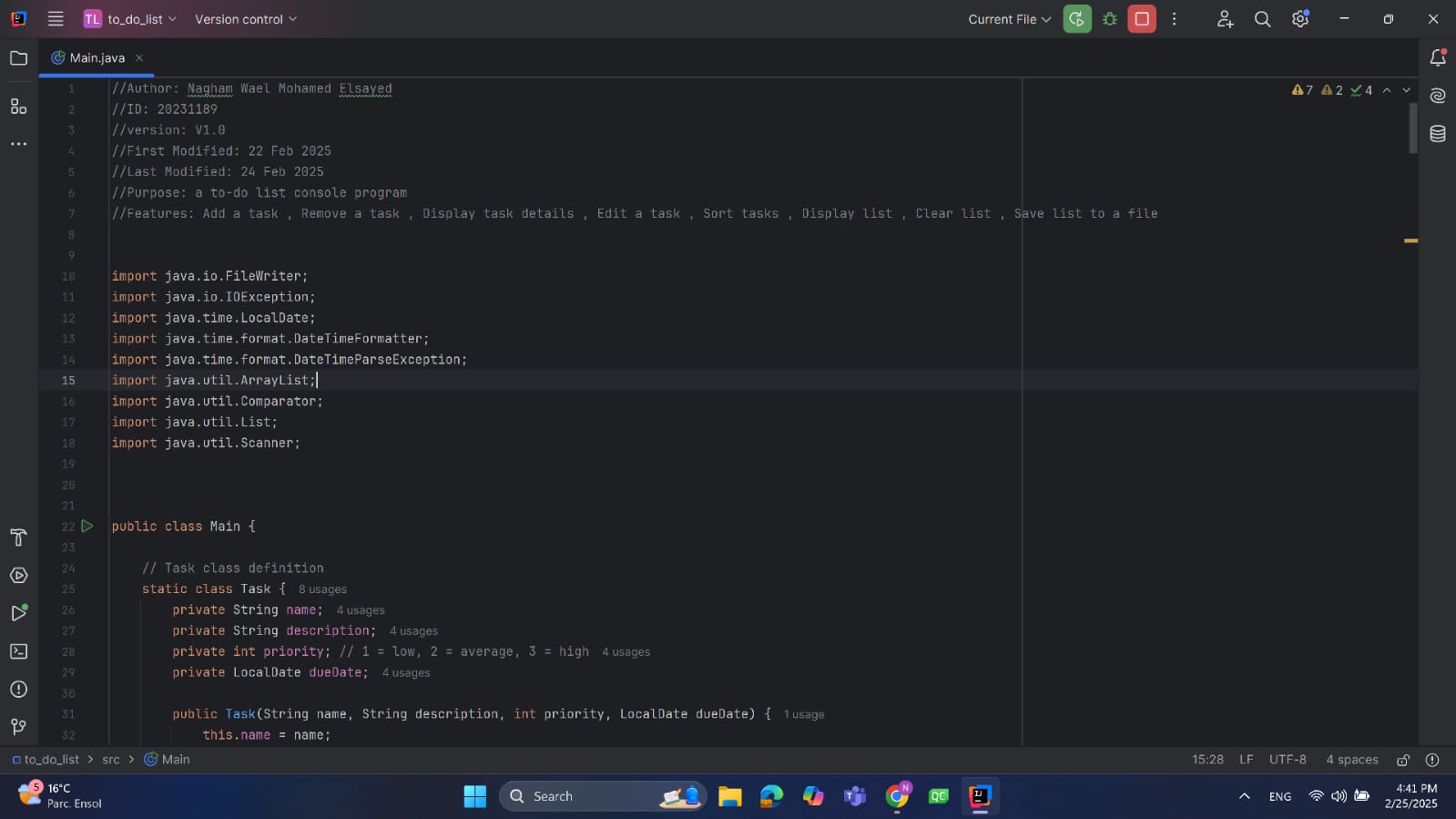
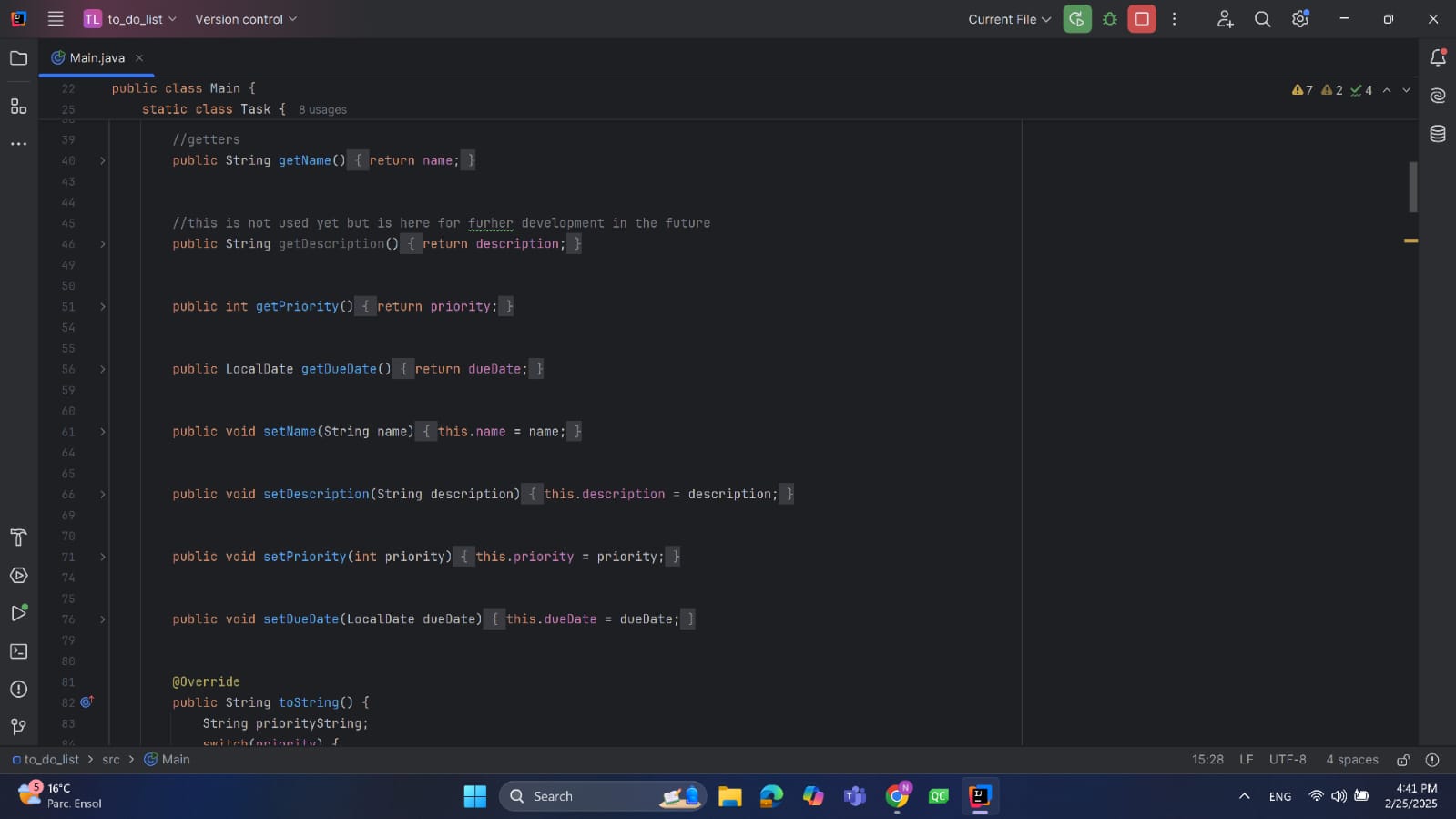
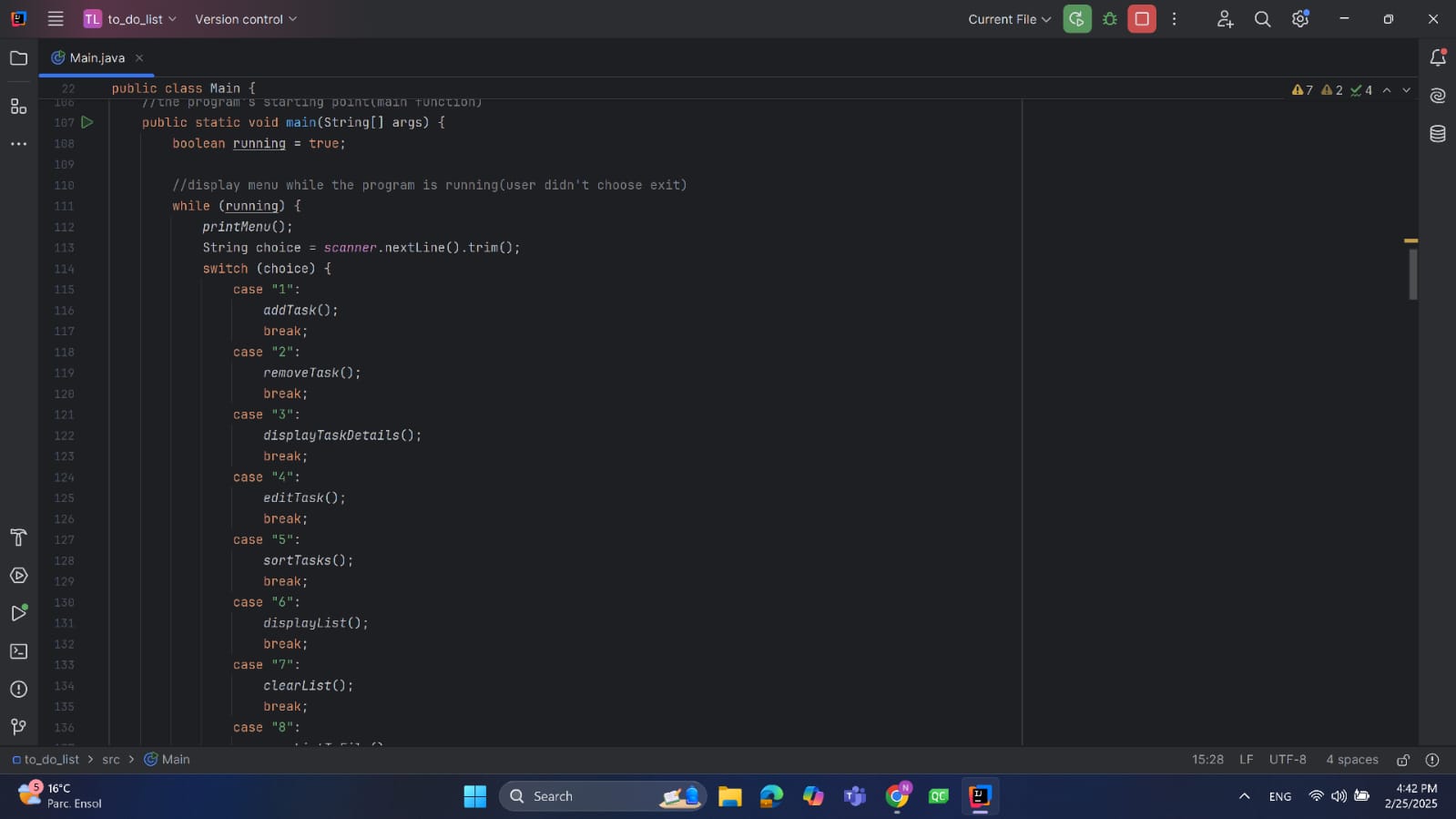
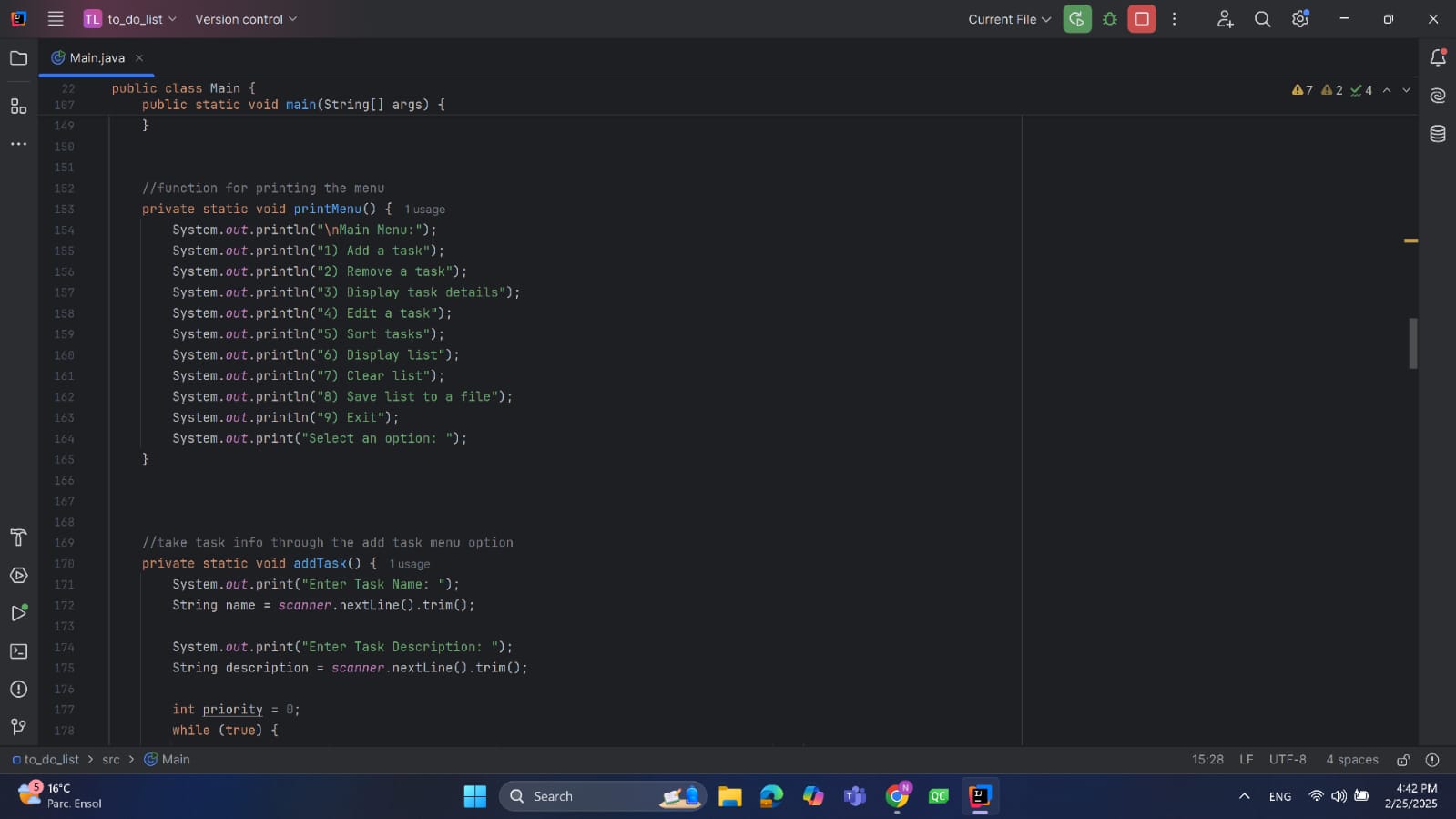
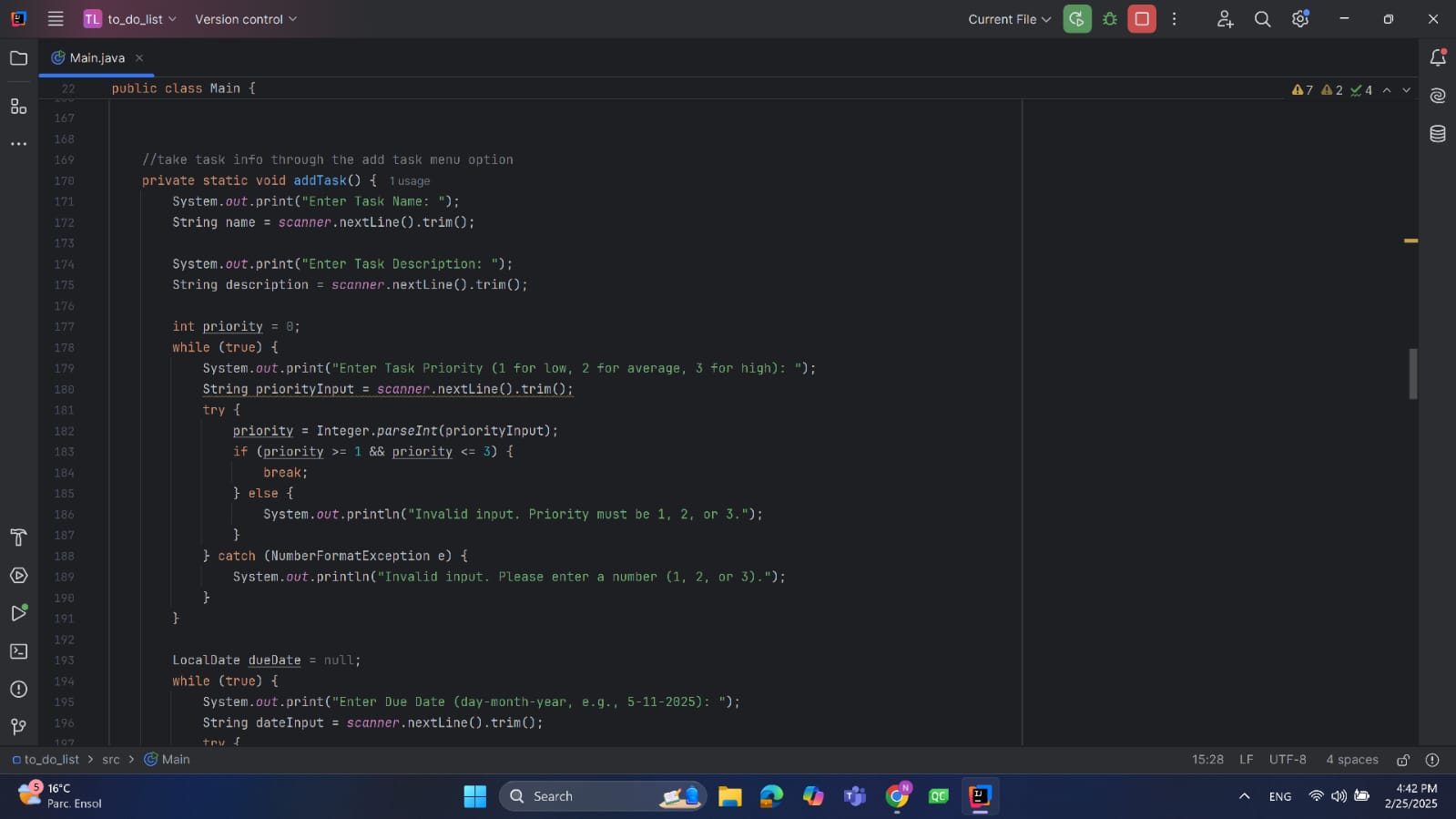
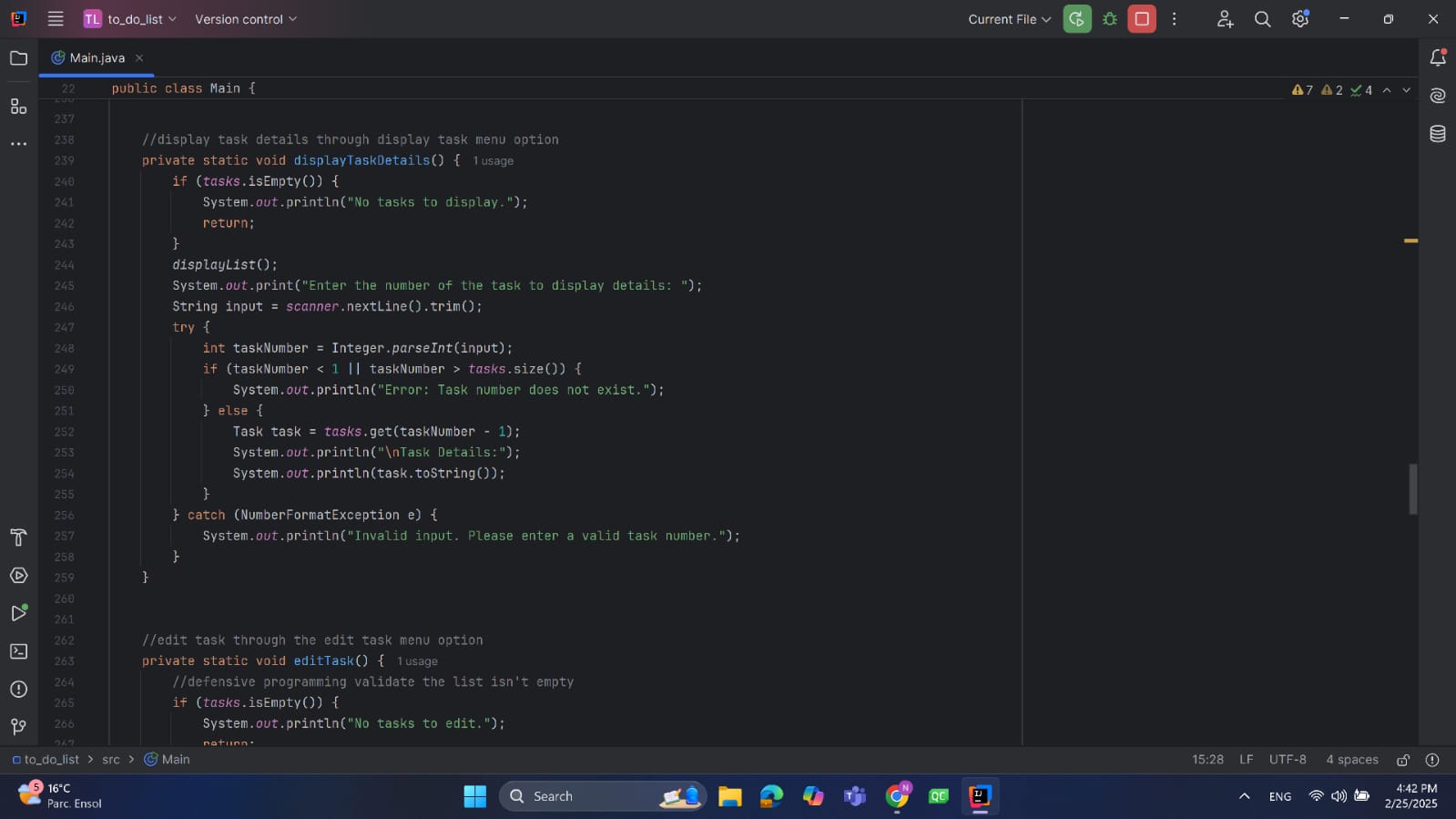
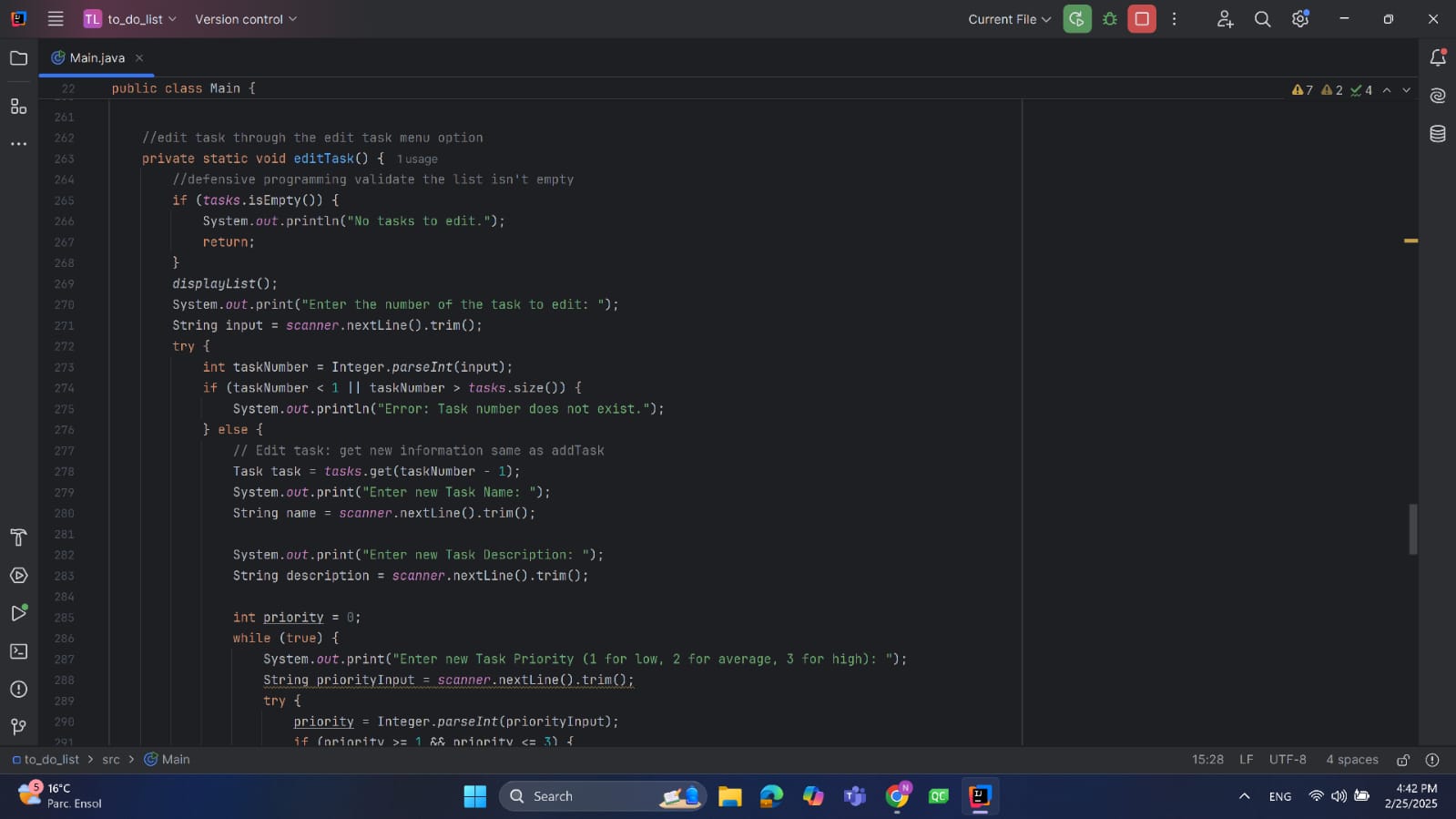
Video Link:

**Nagham Wael:**

* Hours of Study: 12h
* Source of Study: Geeks For Geeks + https://youtube.com/playlist?list=PLCInYL3l2AajYlZGzU\_LVrHdoouf8W6ZN&si=tfRlF2iNxOtJehum
* Main Code Logic:
* public static void main(String[] args) {  
   boolean running = true;  
    
   //display menu while the program is running(user didn't choose exit)  
   while (running) {  
   printMenu();  
   String choice = scanner.nextLine().trim();  
   switch (choice) {  
   case "1":  
   addTask();  
   break;  
   case "2":  
   removeTask();  
   break;  
   case "3":  
   displayTaskDetails();  
   break;  
   case "4":  
   editTask();  
   break;  
   case "5":  
   sortTasks();  
   break;  
   case "6":  
   displayList();  
   break;  
   case "7":  
   clearList();  
   break;  
   case "8":  
   saveListToFile();  
   break;  
   case "9":  
   running = false;  
   System.out.println("Exiting program...");  
   break;  
   default:  
   System.out.println("Invalid option. Please choose from 1 to 9.");  
   }  
   }  
    
   scanner.close();  
  }

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* Images for code :



Video Link:

https://youtu.be/VqOBuBvfRiw?si=WCep92XFi12x7LnN

**Fatema El-Zhraa Ahmed:**

* Hours of Study:12 hours
* Source of Study :
* Main Code Logic:
* public boolean MainMenu(){  
    
   Scanner in = new Scanner(System.*in*);  
    
   if (*isFirst*)  
   manager=new ContactManager();  
    
    
   ArrayList<String> choices=new ArrayList<String>(Arrays.*asList*("1","2","3","4","5","6"));  
   String choice=check\_menu("Choose from the following : \n\n1.Add a new contact\n2.Delete a contact\n3.Search for a contact\n4.View all contacts\n5.Save contacts in a file\n6.Exit\n\nEnter your answer : ",choices );  
    
    
   if(choice.equals("1")){//Add  
    
   boolean isRepeated=manager.Add();  
   if(!isRepeated)  
   System.*out*.println("Sorry , it will not be added for repeating phone number or email");  
   else  
   *isSaved*=false;  
    
   }  
   else if( choice.equals("2")){//Delete  
    
   if(manager.Delete())  
   *isSaved*=false;  
    
   }  
   else if( choice.equals("3")){//search for (by name , number)  
    
   ArrayList<String> Choices=new ArrayList<String>(Arrays.*asList*("1","2"));  
   String Choice =check\_menu("you want to search by:\n1.phone number\n2.name\n\nEnter your choice : ",Choices);  
    
   if(Choice.equals("1")) {  
   contact person = manager.Search(true);  
   if(person!=null)  
   System.*out*.println(person.tostring());  
   else  
   System.*out*.printf("Sorry the contact is not found\n\n");  
    
   }  
   else {  
   contact person = manager.Search(false);  
   if(person!=null)  
   System.*out*.println(person.tostring());  
   else  
   System.*out*.printf("Sorry the contact is not found\n\n");  
    
   }  
    
   }  
   else if( choice.equals("4")){//view all contacts  
    
   manager.Display();  
    
   }  
   else if( choice.equals("5")){//Save contacts in file  
    
   if(!*isSaved*)  
   manager.Save();  
    
   *isSaved*=true;  
    
   }else{//Exit  
    
   if(!*isSaved*){  
    
   ArrayList<String>Choices=new ArrayList<String>(Arrays.*asList*("1","2"));  
   String Choice=check\_menu("Do you want to save changes before exiting?\n1.Yes\n2.No\n\nEnter your choice : ",Choices);  
    
   if(Choice.equals("1"))// as task 5  
   manager.Save();  
    
   }  
    
   return false;  
   }  
    
   *isFirst*=false;  
   return true;  
    
  }

Screen Shots :

Video Link : <https://www.youtube.com/watch?v=fuV9C_FRDf8>

**Low-Code/No-Code (LCNC) Tools: Evaluation, Potential, and Impact**

**Evaluation of LCNC Tools:**

**As for Thunkable:**

Thunkable is a popular LCNC platform for building mobile applications with a visual drag-and-drop interface.

It allows users to create cross-platform iOS and Android apps without deep programming knowledge. While it excels in simplicity and rapid development, it has limitations in customization and performance for highly complex applications.

**As for Webflow:**

Webflow is a powerful LCNC tool for building responsive websites without writing code. It provides extensive design customization, CMS integration, and e-commerce capabilities.

However, for more advanced backend functionality, developers may still need to use custom code or external integrations.

**The potential of LCNC Tools:**

**As for Thunkable:**

Thunkable has great potential for facilitating mobile app development, particularly for non-developers and small businesses.

It enables rapid prototyping and deployment of functional mobile applications, making app development more accessible.

**As for Webflow:**

Webflow empowers designers and entrepreneurs to create sophisticated websites without relying on developers.

It is particularly useful for startups and businesses that require dynamic, responsive websites with CMS capabilities but lack coding expertise.

**What LCNC Tools Can Do:**

**As for Thunkable:**

* Create cross-platform mobile applications.
* Integrate APIs and external services.
* Implement UI/UX elements with a drag-and-drop editor.
* Directly publish apps to the Apple App Store and Google Play Store.

**As for Webflow:**

* Build highly customizable, responsive websites.
* Integrate CMS and e-commerce features.
* Use pre-built animations and interactions.
* Export clean HTML, CSS, and JavaScript for further customization.

**Benefits of LCNC Tools:**

**As for Thunkable:**

* Simplifies mobile app development.
* Reduces costs and time-to-market.
* Provides a user-friendly interface for non-technical users.
* Allows for quick prototyping and iteration.

**As for Webflow:**

* Enables high-quality web design without coding.
* Offers powerful customization and design flexibility.
* Integrates seamlessly with CMS and third-party tools.
* Generates clean and exportable code for further modifications.

**The Quality of the Systems They Produce:**

**As for Thunkable:**

Thunkable produces functional and user-friendly mobile applications suitable for MVPs, small business apps, and educational purposes.

However, apps developed with Thunkable may face performance and scalability challenges compared to those built with native development.

**As for Webflow:**

Webflow websites are visually stunning and optimized for performance. The platform ensures high-quality front-end development, but for complex backend functionalities, additional integrations or custom coding may be required.

**Will LCNC Tools Take the Job of Developers?**

**As for Thunkable:**

While Thunkable simplifies app development, it will not replace developers. Advanced mobile applications still require custom coding, optimization, and backend functionalities that LCNC tools cannot fully provide.

**As for Webflow:**

Webflow empowers designers and marketers to build and maintain websites without developers, but for highly dynamic and database-driven applications, developers are still essential. Webflow acts as a tool that enhances productivity rather than replacing professional developers.

**Compare Between Two LCNC Tools:**

|  |  |  |
| --- | --- | --- |
| **Key of Compare** | **WebFlow** | **Thunkable** |
| **Purpose** | Website design and development | Mobile app development (iOS & Android) |
| **Target Users** | Designers, developers, and businesses looking to create responsive websites | Individuals and businesses looking to create mobile apps without coding |
| **Drag-and-Drop Interface** | Yes, for building websites visually | Yes, for building mobile apps using a block-based interface |
| **Export Code** | Can export HTML, CSS, and JavaScript | No direct code export; apps are built within Thunkable’s environment |
| **Use Cases** | Landing pages, business websites, e-commerce stores, blogs, and portfolios | Prototyping, MVPs, mobile app development for business, education, and personal projects |