****

**Programming**

**40201100**

**L/618/7398**

**Section (0)**

**Submitted to**

Dr. Huthaifa AL-Omari.

**Submitted by**

Hamza AL-Risheq.

**Fall 2022 – 2023**

Table of Contents

[**Part 1:** 3](#_Toc128516248)

[1.1: The process of building an application from writing to execution: 3](#_Toc128516249)

[1.2: Algorithms: 7](#_Toc128516250)

[1.3: ASB system algorithm: 8](#_Toc128516251)

[1.4: code submitted in the GitHub: 10](#_Toc128516252)

[1.5: Evaluate the implementation of my algorithm: 10](#_Toc128516253)

[**Part 2:** 12](#_Toc128516254)

[2.1: Define what is meant by a programming paradigm: 12](#_Toc128516255)

[2.2: The codes submitted on GitHub: 13](#_Toc128516256)

[2.3: Comparing between the programming diagrams: 13](#_Toc128516257)

[First: the procedural programming diagram 13](#_Toc128516258)

[Second: the OOP diagram (object-oriented programming) 14](#_Toc128516259)

[Third: the event driven code 19](#_Toc128516260)

[2.4: Critically evaluate the code examples that you have developed: 21](#_Toc128516261)

[**Part 3:** 22](#_Toc128516262)

[3.1: Evidence on how IDE was used to manage the development process of my code implementations: 22](#_Toc128516263)

[3.2: Evaluation of using IDE for development of applications contrasted with not using an IDE: 22](#_Toc128516264)

[3.3: Debugging: 25](#_Toc128516265)

[3.3.1: Debugging process and different error types that I have dealt with: 25](#_Toc128516266)

[3.3.2: different error types that you have dealt with and how you and how solved them: 27](#_Toc128516267)

[3.4: Evaluate how the debugging process can be used to help develop more secure, robust applications: 28](#_Toc128516268)

[3.5: coding standards: 29](#_Toc128516269)

[References: 32](#_Toc128516270)

[Part 1 references: 32](#_Toc128516271)

[Part 2 references: 32](#_Toc128516272)

[Part 3references: 33](#_Toc128516273)

[Declaration form: 34](#_Toc128516274)

# **Part 1:**

## 1.1: The process of building an application from writing to execution:

There are many stages must be taken in order to build an application, and these steps help the programmer in developing a strategy that enables him or her to be aware of what they are doing, what the next step must be, and what the application's final look will be.

The steps:

1. **Defining the problem:**

As a programmer, when a client asks for an application, we must start by defining the problem (analyzing the problem). which entails understanding the application's output (results), data that will be used as input, software interfaces, and what are the requirements for the application.

1. **Planning the solution (Algorithm):**

In this step as a programmer, you must create a plan step by step for your solution. And we have two common ways which are flowchart and pseudocode.

1. **Coding:**

After defining the problem and planning for the solution, you must start implementing your solution; based on the plan you created in the previous step (“You'll translate the logic from the flowchart, pseudocode, or another tool to a programming language.”). While you are writing your code, you must follow some guidelines for efficient code, such as:

* Use of meaningful names for variables.
* Make an indentation when you are opening a curly bracket.
* Using comments for unknown codes.
* Using the camel case when we are declaring the variables.

1. **Debugging:**

In this step, as a programmer, you must check your code line by line to ensure that the code is working properly, and if there are errors you must fix them.

1. **Testing:**

A variety of appropriate test cases are used to evaluate the program.

It is required to create a test plan for the program even before it is designed.

This ensures that the specs are well understood. It is important to find and test the simplest and most unique scenarios. The maximum and minimum values of each variable should always be included as test data.

1. **Documentation:**

A written full explanation of the programming cycle and specific program information constitutes. The cause and characteristics of the issue are typically included in program documentation papers. Also, this documentation includes the planning solution, testing results, and comments on the program.

1. **Maintenance:**

Patching, updating, and correcting the application. While actual execution of the program. (While the customers are using the application).

How I used these seven steps on building (ASB) system:

As junior developer for a software development company that produces software

for digital platforms. And part of the R&D team was to build an ASB system. So, we have used the following steps:

1. **Defining the problem:**

We have to meet the Jordan Collegiate Programming Contest (JCPC) to know what they want in their system and what is the output and inputs so, so we got that the system must include the following:

* That the ASB system includes 5 rows first one: Rank, second: Team ID, third: Team Name, fourth: Solved Question, and fifth: Elapsed Time.
* When a certain team solves a question correctly, the ASB system records the score of that question for the team and records the calculated time taken by the team to solve the question.
* The ASB system is supposed to display the results of the different competing teams in a sorted way, showing the team with the highest score first then the team with the next highest score, and so on.
* The winning team is the one that solves the maximum number of questions correctly within a minimum amount of time.
* The contest is supposed to last for 5 hours (300 minutes), where 10 competing teams are trying to correctly solve 10 questions.
* Following is an output sample of the ASB system: A picture containing diagram

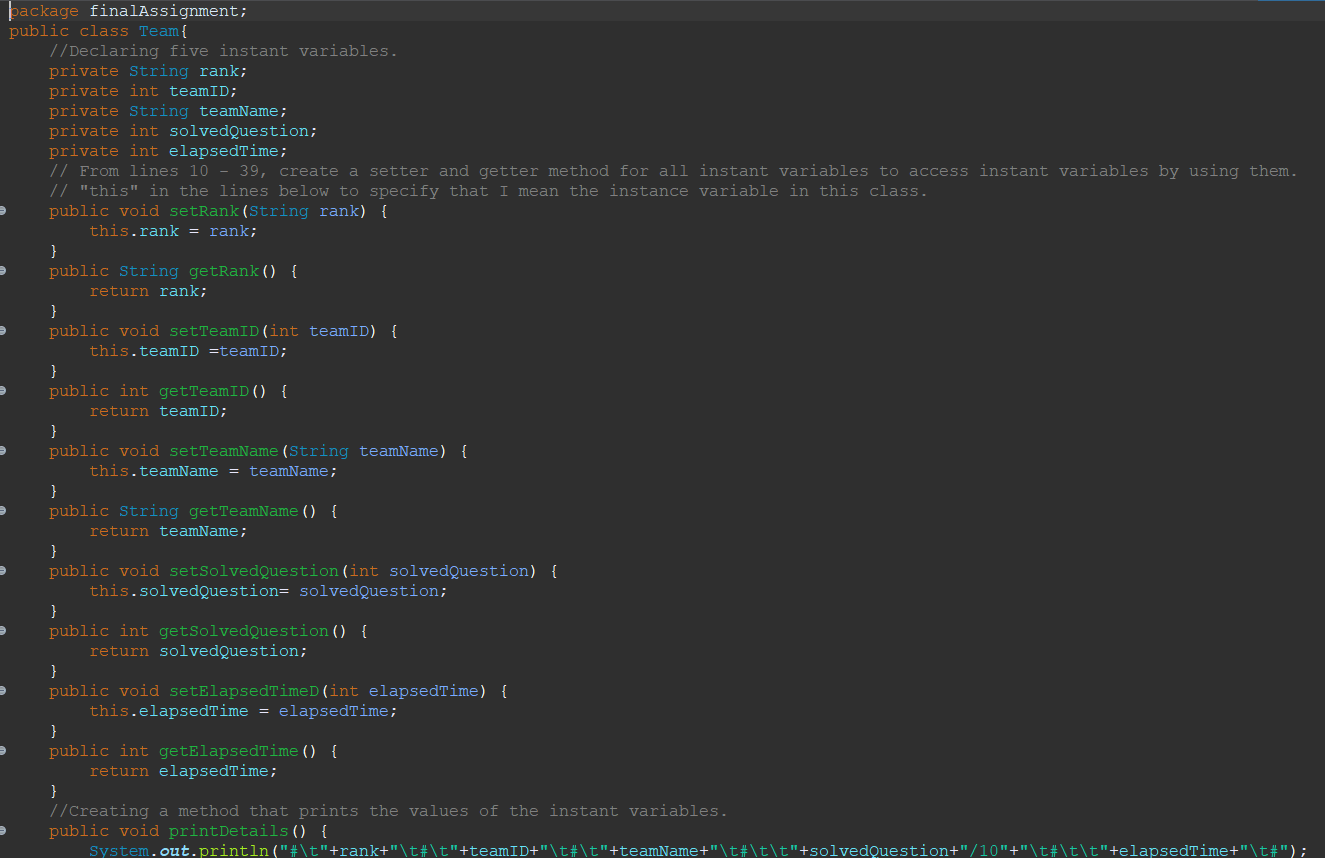
  Description automatically generated

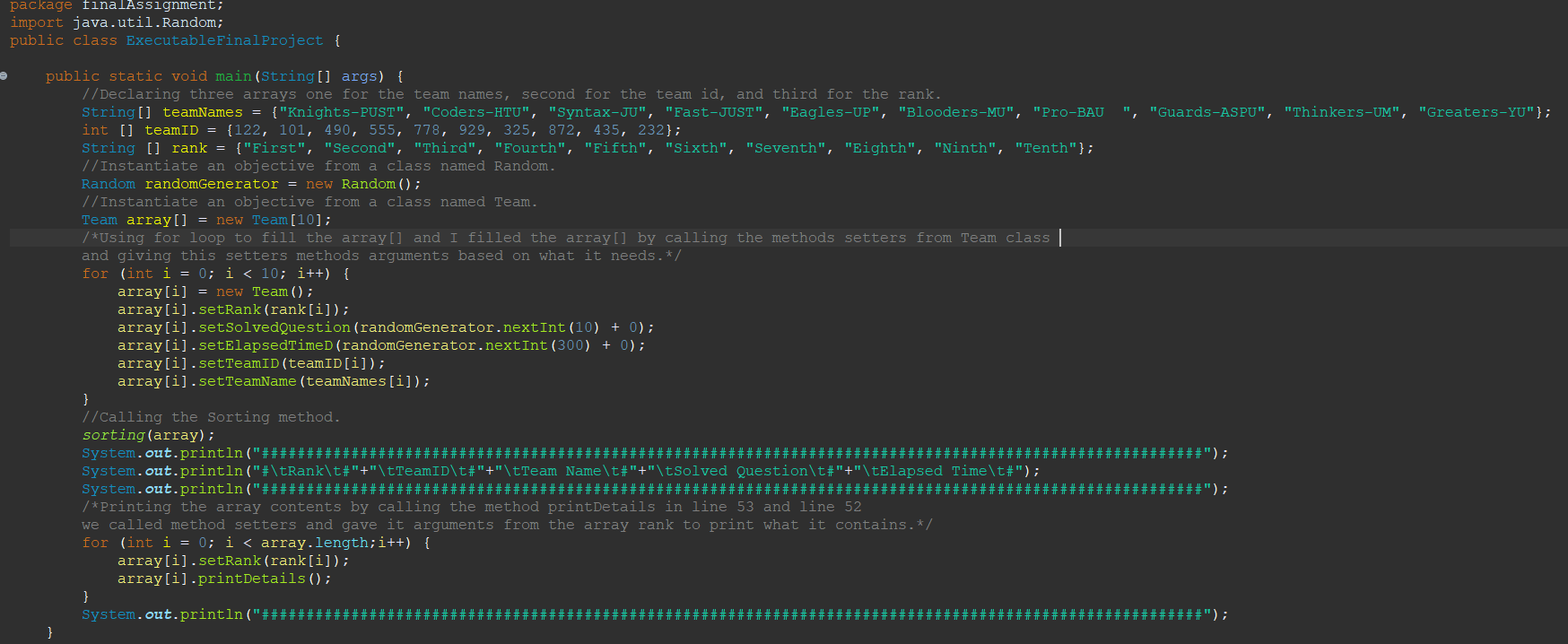
1. Planning the Solution (Algorithm):

And we chose to use the pseudocode to plan for the solution:

* we have to create a class named Team (inexecutable file).
  + declaring 6 instant variables.
  + Creating setter and getter methods for the 6 instant variables.
  + Creating a method to print the value of the instant variables.
* Then creating executable file named ExecutableFinalProject
  + Then creating the main method.
  + Declaring 3 array these arrays for the team’s name, team id, rank.
  + Instantiate2 objects, one for the random generator from a class Random, and one for creating array from a class named ASBrecord (the one we created above).
  + Start to fill the ASbrecord array by calling the setters and give it arguments based on what it needs.
  + Creating a sorting method by using the bubble method.
  + Calling the sorting method in main function.
  + Start to print the table as they asked for in the previous step.

1. Coding:

****

****

**Text

Description automatically generated**

As we can see I created a code based on what I wrote in pseudocode that was created in the previous step and my code is based on the guidelines.

1. **Debugging:**

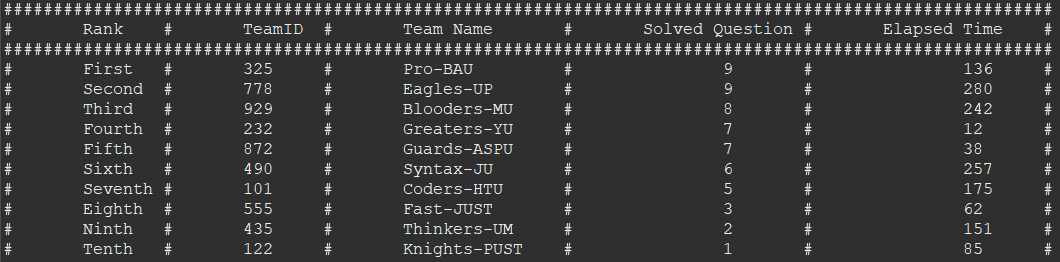
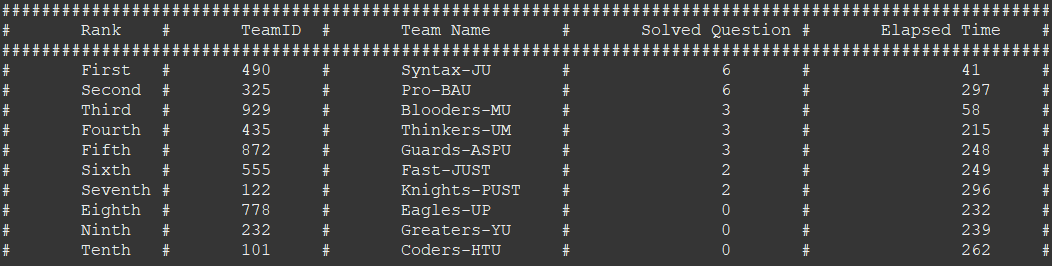
When code is written, the debugging process begins, and it is done in stages as the code is combined with other programming units to create a software product.

Unit tests, code reviews, and pair programming are all strategies that can help you troubleshoot a large application with tens of thousands of lines of code. To detect flaws, check the code's logging and use a standalone debugger tool or an integrated development environment's debug mode (IDE).

At this point, the developer's experience with standard error messages may be beneficial.

Even the cleanest code might be difficult to debug if developers do not adequately note their code.

1. **Testing:**

We have tested the code many times in many ways and situations to make sure that it is working correctly. And in the next below figures are samples of the results we got from the testing, and it was exactly as we wanted.

1. **Documentation:**

As we can see in the figures above, I wrote comments, and I documented the solution plan in step two as we can find on the pages (4-5).

1. **Maintenance:**

Our program (application) is created very well, and as we see, it does not need any maintenance at this moment, but if any problem is found in the future, we will be ready to fix it.

## 1.2: Algorithms:

An algorithm is a set of rules and guidelines that must be followed in order to solve a problem and get the results we desire. We are using algorithms in our daily life such as bedtime routines, as a junior program I am using algorithms for coding, and recipes.

To achieve an efficient algorithm, we must do some things:

* The algorithm must be clear and unambiguous in all the steps.
* The inputs must be well-defined.
* The outputs must be well-defined.
* The algorithm must have a start point and end point (Finiteness).
* The algorithm must be simple as much as possible and practical (Feasible).
* The algorithm must be understandable for anyone (Language independent).

Comparing their efficiency against brute forcing:

Brute forcing: “is a general problem-solving technique that involves generating a list of all the possible candidates for a solution”.

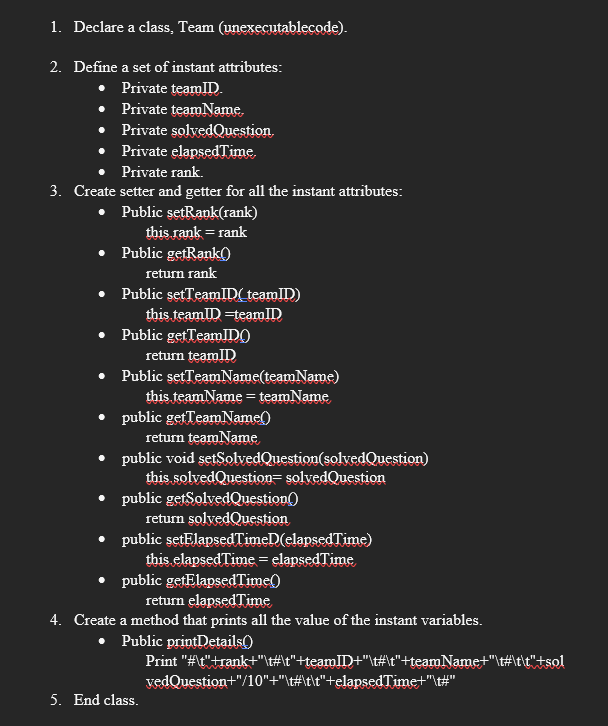
If we compared brute force to the algorithm. We shall discover that brute force is one of the least efficient methods because it takes more time than an algorithm and requires a powerful computer to examine all possible solutions (wasting time and effort). Whereas an algorithm can be used by any computer because it does not waste the computer's power.

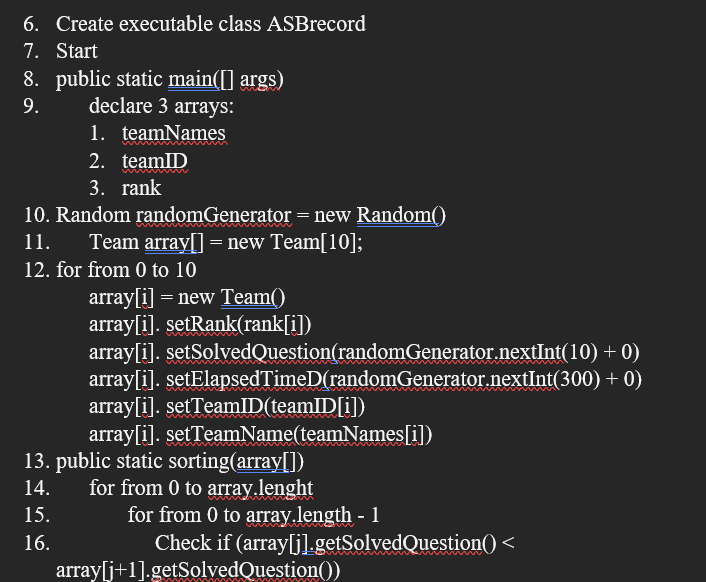
How the algorithm relates to the application development process and to implement the algorithm to the code:

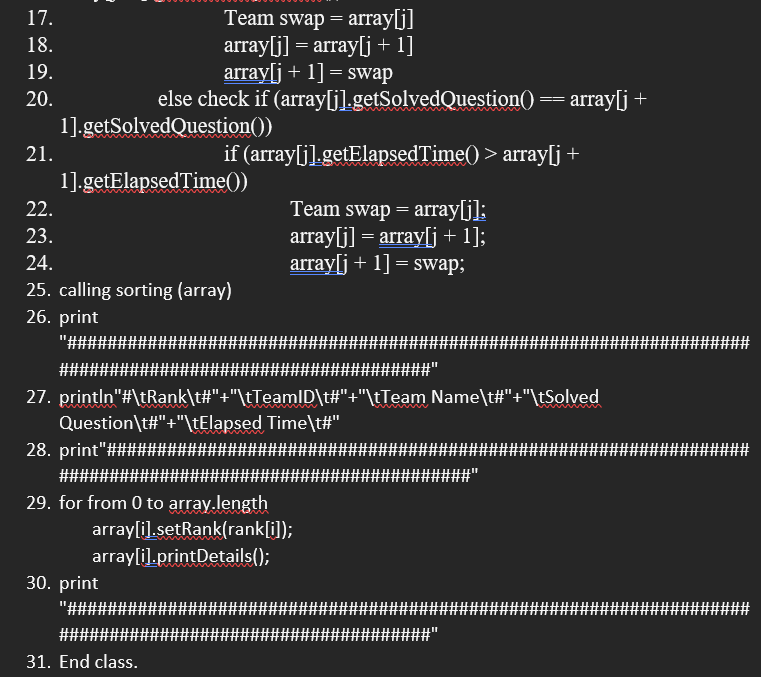
It is relevant to the application development process in all steps, but it is most visible in the first three. In the first step, when we had to identify the problem, we had to use an algorithm to comprehend and ensure that everything the customer desired was discussed. In the second stage, we utilized an algorithm to create a solution plan to determine how we will solve our problem and what steps we must take (some people are calling the second step in the application development process algorithm). In the third step, when coding, we used many algorithms to develop an efficient code that met the needs of the customer. Also, the algorithm helped us to know what to do while we are coding because when we have an algorithm it will be easier to know what to write while we coding.

## 1.3: ASB system algorithm:

1. Create an inexecutable class.
2. Declare 5 instant variables which are rank, team id, team name, solved question, and elapsed time.
3. Create setter and getter methods for each of the instant variables (5 setters method­s and 5 getters methods).
4. Create a method that prints all the value of the instant variables.
5. Create an executable class.
6. Start.
7. Import the java.util.Random.
8. Create a main method.
9. Declare 3 arrays which are array for team names, team id, and rank.
10. Instantiate 2 objects the first one type is Random, and the second from type is Team .
11. Start to fill the array typed Team by a loop from 1 to 10.
12. Inside the loop from the previous step calling all the setters for the instant variables from the ASBrecord call to give the instant variables values.
13. Create a new method in the executable file for sorting by using the bubble algorithm for sorting and pass the values for the sorting method. This method takes an array from Team type.
    1. Create a for loop from 0 to arry.length.
    2. Inside the first loop create another loop from 0 to array. length-1 to sort array based on the solved question and elapsed.
    3. Inside the second loop create an if statement to check if the index on the left is smaller than the right index based on the solved question so if the condition is true the code will swap between the two indexes.
    4. And again, inside the second for loop create else if statement to check if the solved question time is equal if they are it will enter the code and inside the else if statement create an if statement to check if the index on the left is bigger than the right index based on the elapsed time. If the condition is true, the code will swap between the two indexes.
14. Call the sorting method and pass the array to sort it.
15. Print the topside of the table.
16. Print the contents of the array by creating a loop that checks 0 to the array length.
17. Inside the loop from the previous step calling two methods from the Team class, first one is setter method for the instant variable rank and give it argument the array for the rank that has been created inside the main method. Second one is calling that method print all the value of the instant variables.
18. End







## 1.4: code submitted in the GitHub:

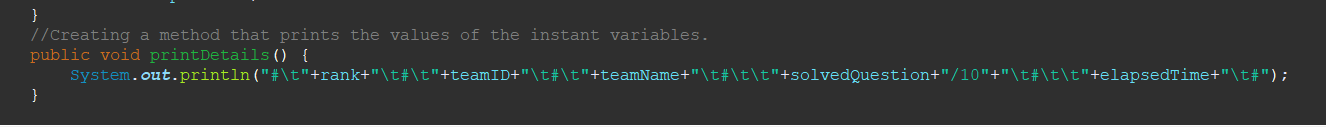
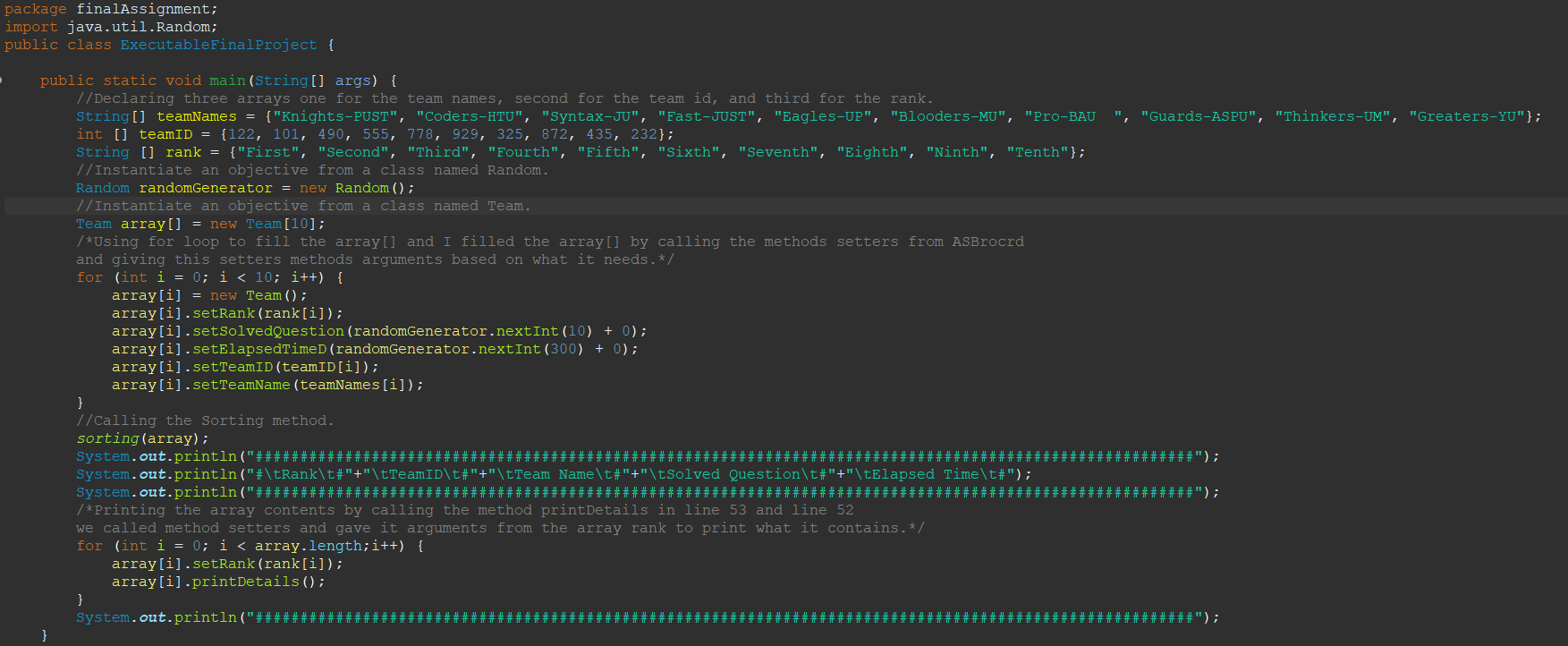
## 1.5: Evaluate the implementation of my algorithm:

The relationship between the algorithm and the code that my code written based on the algorithm as we can see in the figures:

The algorithm is a high-level language for coding. So, if you looked at the code and algorithm I created. You will see that I used every step of the algorithm in my code. And you will also be able to understand everything I wrote in it because the algorithm I created was regarded step by step to follow when implementing the algorithm into code.

Is my algorithm a good one? Returning to the guidelines for a good algorithm, let's see if my algorithm corresponds with them or not: there are seven standards for a good algorithm as I mentioned them in point 1.2 (7):

As I see that my algorithm is a good one because in clear and unambiguous in all the steps, the inputs and outputs are well defined, the algorithm has a start point and end point, it’s simple as much its possible, and it understandable for anyone.



Step 7

Step 12

Step 11

Step 5

Step 9

Step 8

Step 4

Step 3

Step 2

Step 1

Step 10

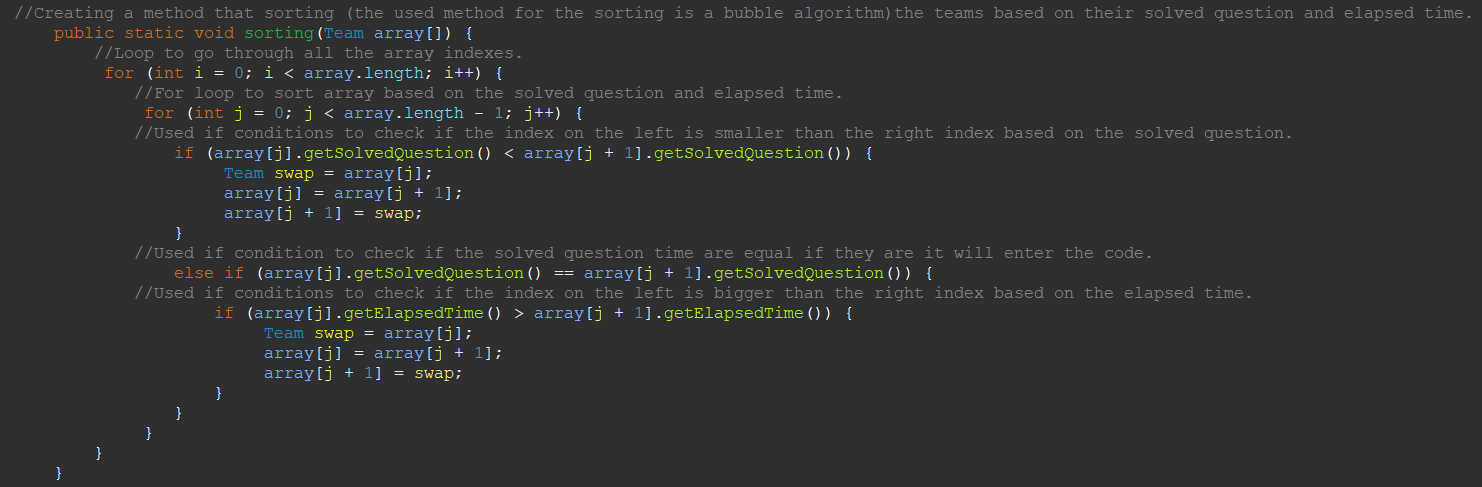
Step 16

Step 14

Step 15

Step 18

Step 17

****

Step 13 - d

Step 13 - c

Step 13 - b

Step 13 - a

Step 13

# **Part 2:**

## 2.1: Define what is meant by a programming paradigm:

Programming paradigm: A particular program or programming language may be organized in a variety of ways or styles. Each paradigm has a unique set of features, structures, and approaches to solving common programming problems. Using a paradigm can be challenging depending on the language. Programming paradigms come in a variety of forms, including procedural, object-oriented, and event driven. Declarative and imperative paradigms are the two primary categories of programming paradigms. Functional, and logic paradigm are both declarative. Procedural object-oriented programming (OOP), procedural, and event-driven are all examples of the imperative paradigm.

Procedural uses a linear top-down paradigm and views data and procedures as separate objects. Based on the concept of a procedure call, procedural programming divides the program into procedures, also referred to as scripts or functions and consisting basically of a set of activities to be performed. It uses subroutines or procedures for calculations. It places a lot of emphasis on dividing programs into named, function-like collections of instructions called procedures.

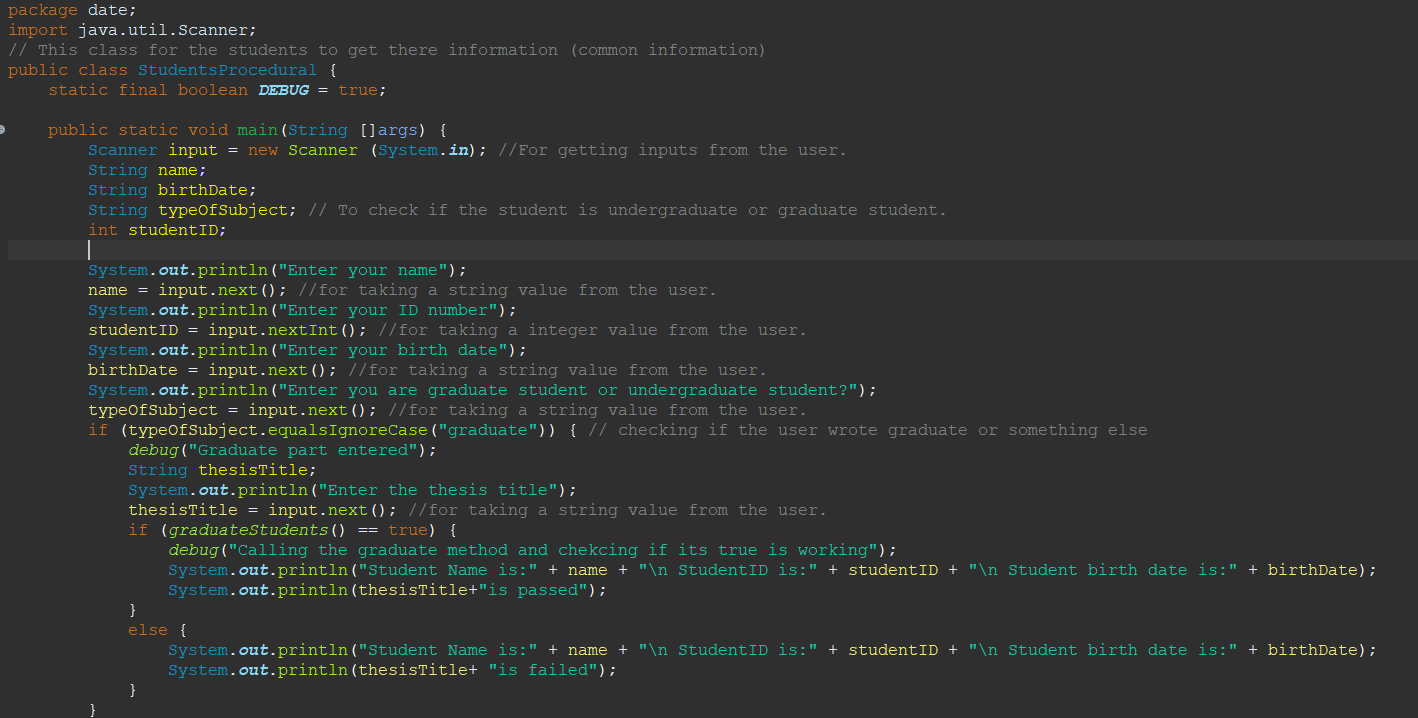
Object-oriented is a style to programming that is based on the ideas of classes and objects. It's used to divide up a piece of software into straightforward, reusable code blueprints, commonly referred to as classes, from which individual instances of objects can be created. Data fields called objects have unique attributes and behaviors. Three fundamental ideas in OOP are inheritance, encapsulation, and classes and instances.

Event driven is one of the programming paradigms written to respond to actions such as user actions (mouse clicks, key presses), sensor outputs, or message passing from other programs or threads. An event is any identifiable occurrence that has significance for system hardware or software. And it works with OOP (object-oriented programming). One of the most known applications based on the event driven is GUI (graphical user interface).

## 2.2: The codes submitted on GitHub:

## 2.3: Comparing between the programming diagrams:

First: the procedural programming diagram: the code is divided into small parts (methods) but in same class. In my procedural code I used 3 methods main meth, undergraduate method, and graduate method.



Text

Description automatically generated

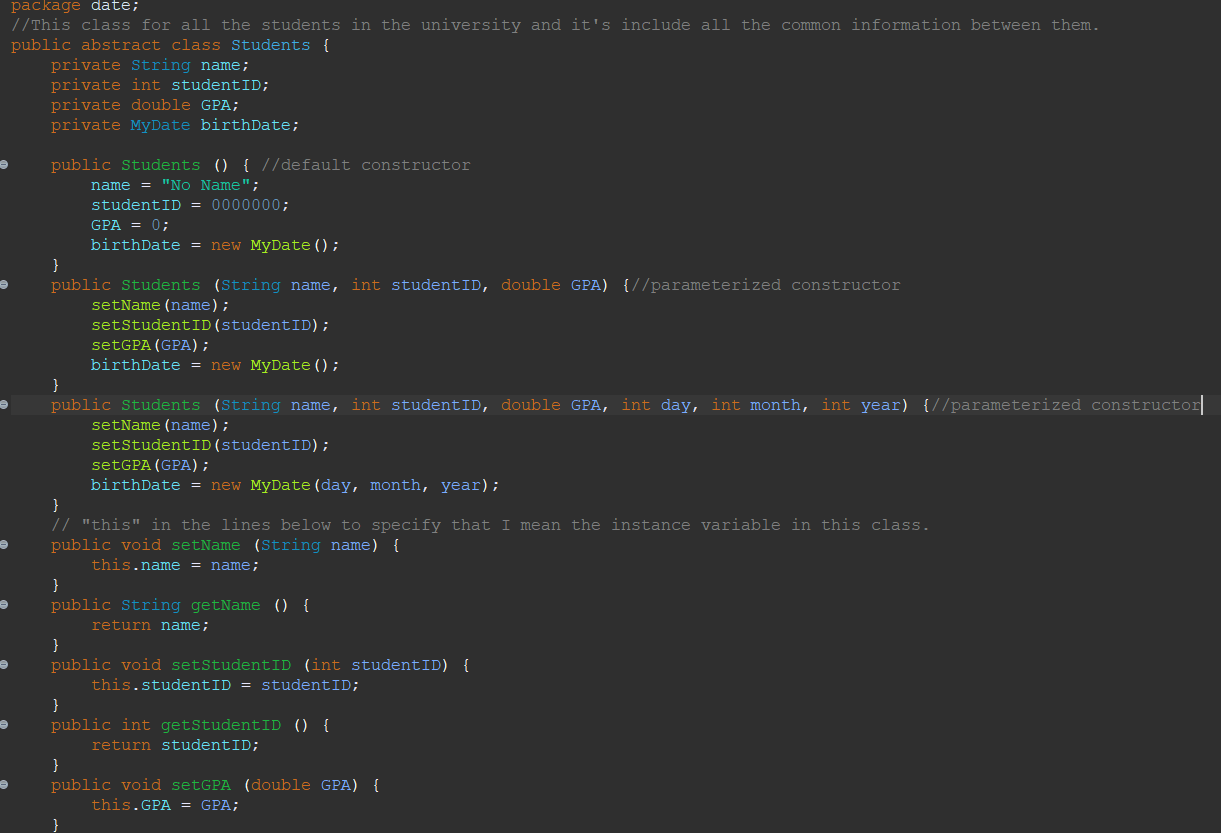
Above the main method I declared a variable for debugging and it’s a constant it will not change so I wrote before the type of the variable final. These pictures show the main method, and the main method includes 4 variables then inside the main method called the other two methods based on what the user enters when the code asks the user to enter graduate or undergraduate. Based on the user entries one of the graduate and undergraduate will be called and both methods will return Boolean value (true or false). Also, we can see in the picture that there are 2 other methods based on what the user will enter one of these methods will be called form the main method or if the user entered something wrongly, he/she will receive a massage telling invalid option. And there is a method named debug this method for debugging all we have to do to stop this method to change the value of the debug variable to false.

**Quick review of procedural programming diagram:**

The main method belongs to the same class as these two others (graduate and undergraduate) methods. Procedural code separates the code into separate parts within the same class in this way.

The procedural diagram employs a top-down strategy, as seen in the images. Also, the procedural is less safe since, as we can see, it is not permitted to create private methods or variables, making it impossible to hide the data. And changing the code is difficult. The primary distinction between procedural paradigms and other paradigms is the inability to implement real-world codes.

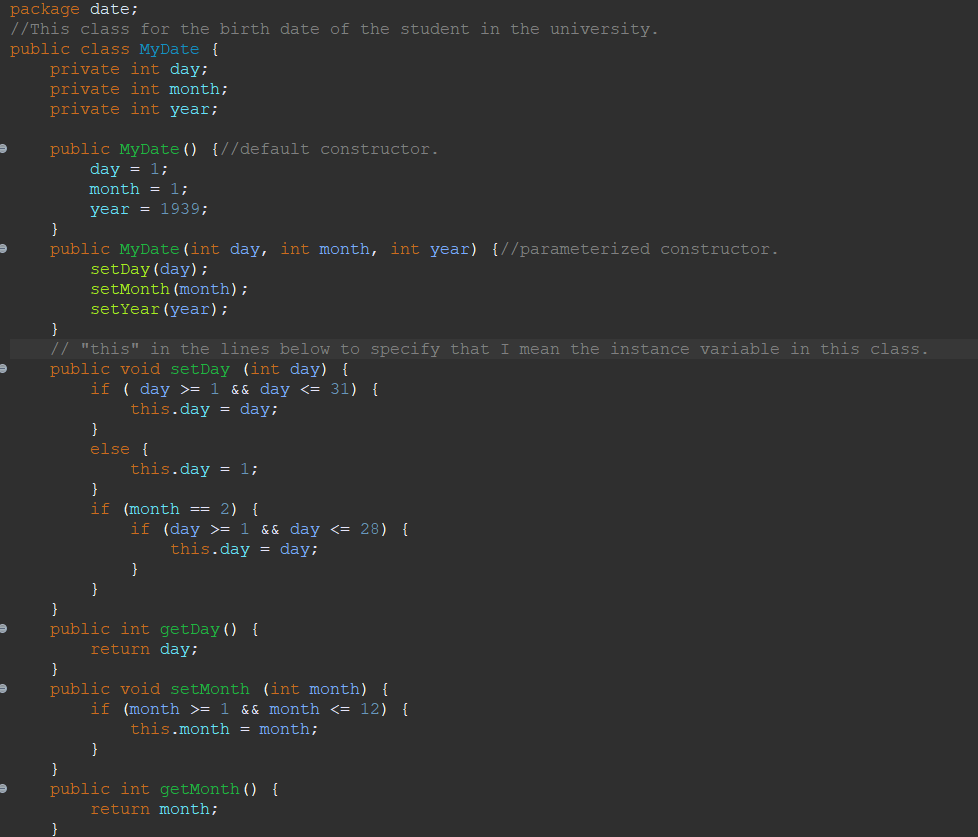
Second: the OOP diagram (object-oriented programming): the code in this type of diagrams we had to create many classes each class own their variables and methods (attributes and behaviors) and I used the encapsulation, composition, polymorphism, and polymorphism. So, it was easier than the procedural paradigms to implement the code that can work in real life because OOP is can secure the data, easier to add or delete data, or shaping the code on the way I want to make the ideas from real life in a code. And the OOP helps to design a huge complex code easier because of the concepts that the OOP provide them.

****

**Text

Description automatically generated**

As shown in the picture we have a class named Students and this is the super class for the other two sub classes. It has an abstract method to force all the classes that will be inherited from the Students class to have a Boolean method named isPassed and one of the private variables is typed from another class (this called composition). Also, I used the overloading features as we can see in lines 46 and 51 because I used the same method but with different signatures.



A screenshot of a computer

Description automatically generated with medium confidence

This class is MyDate class in which the student class created an object from MyDate class. So, there will be instantiated object in the Students class from MyDate class.

Text

Description automatically generated

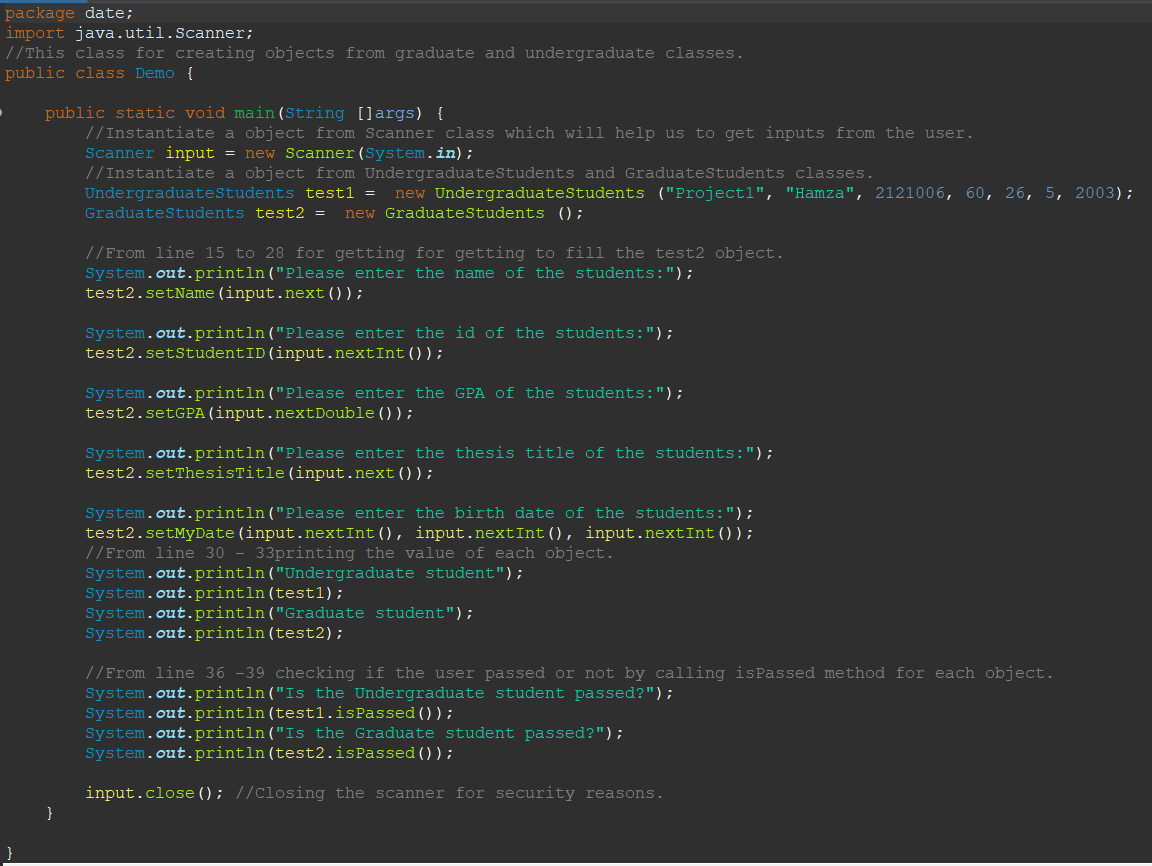
Text

Description automatically generated

In the last two pictures we can see two classes which are sub class for the super class (Students class) so these two-sub class inherited all the methods and variables from the super class.

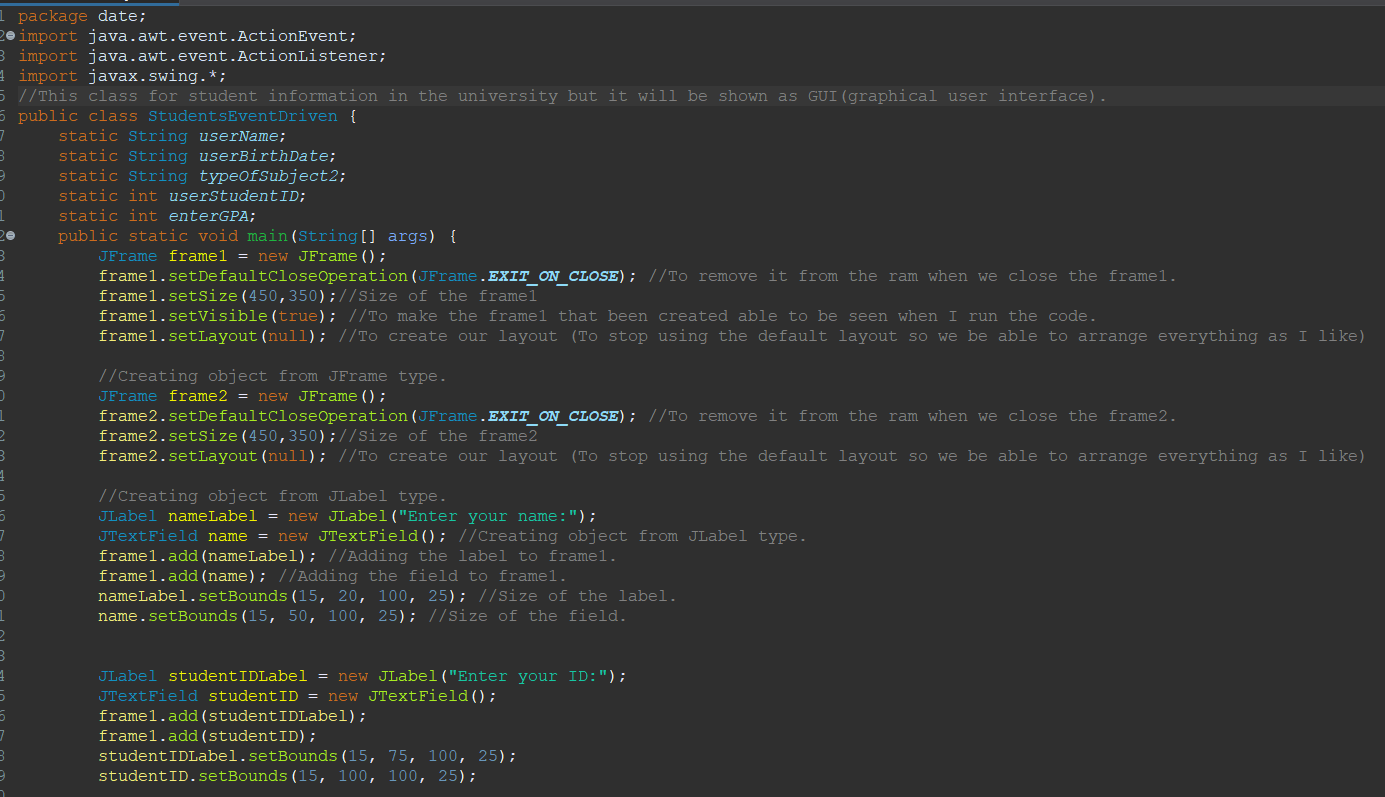
These two sub class each one of them has different methods and variables form the other but the common between them the methods and variables that they got from the super class but both of the two sub class have a method called isPassed because this method is a abstract method in the super class. And both of them have a method called toString method to override the method in the super class.

After we have seen the procedural paradigm and OOP, we will be able to find out the OOP many ways better because we were able to secure the data, implement a real life code, easier to troubleshoot the code to find error or to develop the code in the future, and flexibility through polymorphism helped to edit the code as we like.



In the last picture we can see the demo class for the OOP this class contain a main method and inside the main method I instantiate 3 objects each one from the Scanner class, UndergraudateStudent class, and the last one from the GraduateStudents.

Third: the event driven code we can say that the event driven code is an OOP code, but the shape of the output is different and event driven has something called action listener (action listener is a listener if something happened (event) then a specific code will start so if someone click on a button using the mouse so the specific code will start).



Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

Here we have all the events driven code as we can see this code using the OOP features but there is one different with OOP code that the event driven code has something called action listener in my code we have two action listener if the user clicked on box that will be shown when I run the code a specific code will react with this actin listener and start other code.

When I run the code there a frame will start working. This frame will ask you to enter your name, ID, birth date, and to write if you are graduate student or undergraduate student. Then you will click on a button so clicking on the button will react with this clicking to get you in another frame based on what you wrote if you are graduate or undergraduate. If you wrote undergraduate, you will go to another frame and this frame will ask you to enter your project title and your GPA then you will click on the button to check if you passed the project or not. But if you wrote graduate you will go to another frame and this frame will ask you to enter your thesis title and your GPA then you will click on the button to check if you passed the thesis or not.

## 2.4: Critically evaluate the code examples that you have developed:

* Procedural code: As can be seen from my code, it complies with all coding requirements and is clear enough for readers to understand. The procedural code is divided into three methods; thus I have to divide it into three smaller codes inside one code. And each of the three codes' methods serves a certain purpose. To achieve the result I desire, all of the strategies work together. Moreover, all of the code for a procedural code must be written in a single class, which is what I did to construct my procedural code. The code also functions in the way that procedural programming is supposed to—line by line, bottom to top. Because the procedural paradigm does not provide as many features as other paradigms, it was difficult for me to see the result.
* OOP code: I used all available capabilities since they made it easier for me to transfer what was in my head to the code base because of the features that the OOP paradigm supports. The private variables for each class in the OOP paradigm also allowed me to secure the data. Also, when I used inheritance, it enabled me to create only one class while still providing other classes with the same characteristics.
* Event-driven code: The code I created asked the user for a variety of information before taking them to a different frame when they clicked on a particular item with the cursor. This paradigm assisted me in writing code that can be presented to users—even those who are not programmers—so that they can utilize it. I tried my best in the situation when employing try and catch will boost security. Also, I did my best to arrange all the fields and labels so that they would make a pleasing visual for the user.

# **Part 3:**

## 3.1: Evidence on how IDE was used to manage the development process of my code implementations:

IDE (integrated development environment): is software that combines, in one place, all the tools needed for a software development project. It provides all the tools for programmers in one application, such as debugging, managing applications, code editors, source control systems, and more.

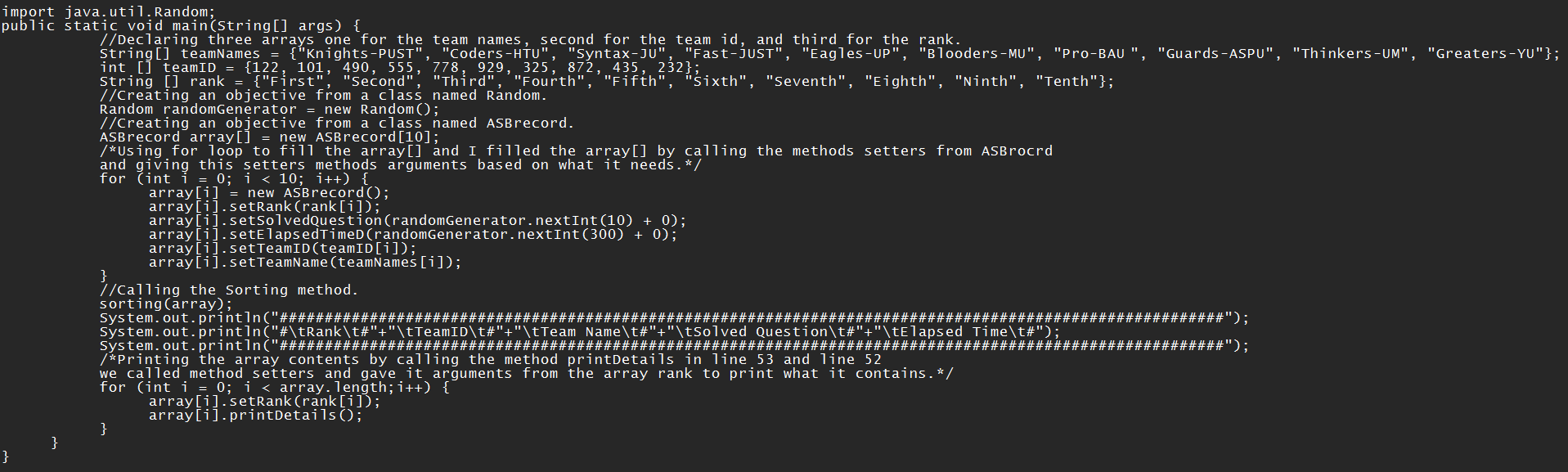
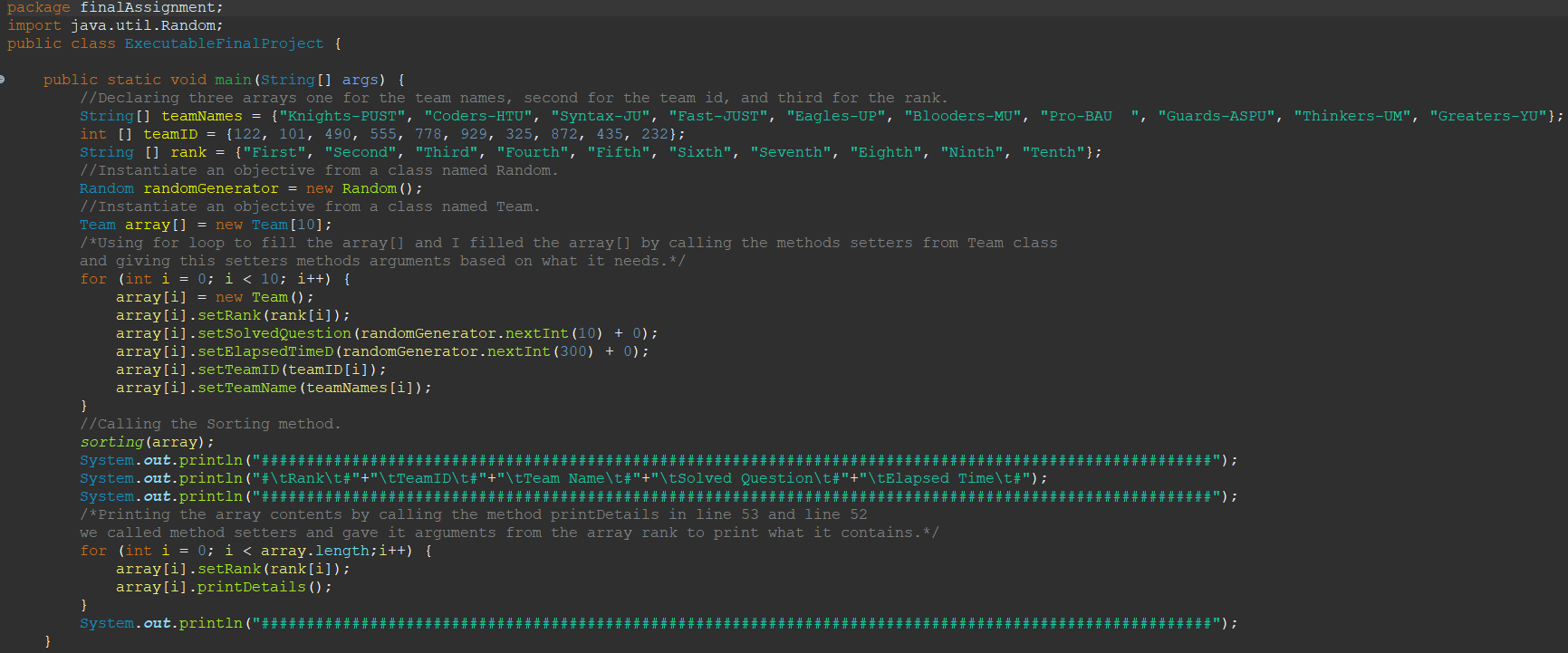
I have written, tested, and implemented the algorithm that I created for ASB system by using the IDE because the IDE gave a perfect environment to build the application I have been asked for. Evidence on how IDE was used to manage the development process of my code implementations:

* Sometimes I made syntax errors, such as forgetting to end the line of code with a semicolon. So, the IDE gave me a red alert which helped me notice the mistakes and correct them.
* It helped me import more than one library easily. Which helped me to have access to more classes, which I needed in implementing the application. It was easy to find all the classes I wanted, such as Random class to create an object from it.
* It helped me to debug the code with one click on a tool.
* It helped to use the OOP (Object-oriented programming) easier because IDE provides a tool for inspecting objects and object-oriented classes within the program.
* It helped me to test more than one scenario in the ASB system easier and faster than using cmd.
* When I was implementing the application code. The IDE was detecting the meaning of a symbol or term and then displaying it in the code in different colors or styles (such as bold or italic) to make it easier to read and understand.

## 3.2: Evaluation of using IDE for development of applications contrasted with not using an IDE:

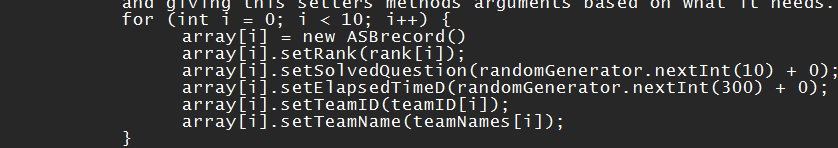
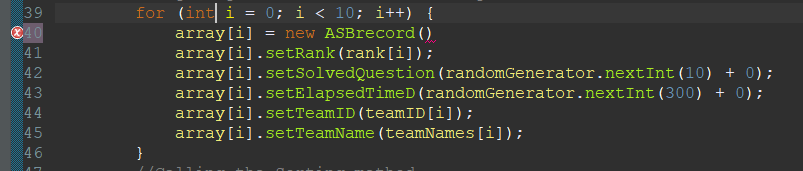
There are many things can shows us the difference in using IDE for development of applications contrasted with not using an IDE:

* Understandable code: while we are using the IDE its easier to understand the code and the steps are used because of the syntax highlighting. As we will see in the next two figures:

****

in the first figure we used the IDE which highlighted everything; so, everything is easy to understand and to know what is used for. Also, it is more comfortable for our eyes. But when we did not use the IDE, it was difficult to understand the process of the code and hard to read and focus.

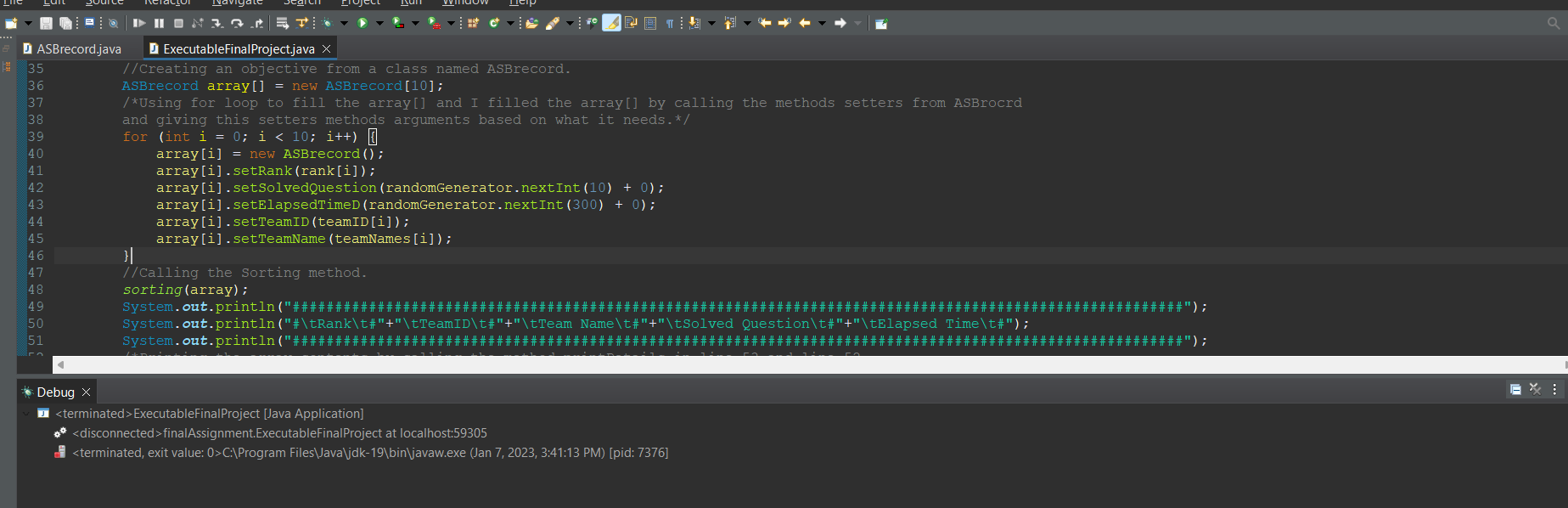
* Using IDE is easier to find the syntax mistakes:

****

When we used the IDE as we can see in the left figure, we had a syntax error, and the IDE gave us as alert that we have a syntax error in line 40. But in the right figure in gave us nothing which is hard to know that we had till we run the code and we will not be able to know in which line the error because the lines are not numbered as in IDE.

* There is a debugging tool in the IDE but when we are not using the IDE we wont have a debugging tool:

**A screenshot of a computer

Description automatically generated**

Debugging tool in the IDE and the result of it but when we are not using IDE, we will not have a debugging tool.

* Easier to run and test the code while you are using the IDE because with one click you run the code but when you are not using it you must go to cmd and write more than one command to run the code.
* A screenshot of a computer

  Description automatically generated with medium confidenceThere are many methods that explain why they are used for what in the IDE (especially the methods are used for event driven. So, it will help us and reduce the effort and time when we are implementing our application).
* All your classes and packages are saved in one place, and it is easy to find them when you are using the IDE. But while you are using the normal files in laptop you may lose the files or forget where they are.

Text

Description automatically generated

## 3.3: Debugging:

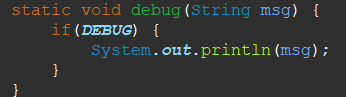
### 3.3.1: Debugging process and different error types that I have dealt with:

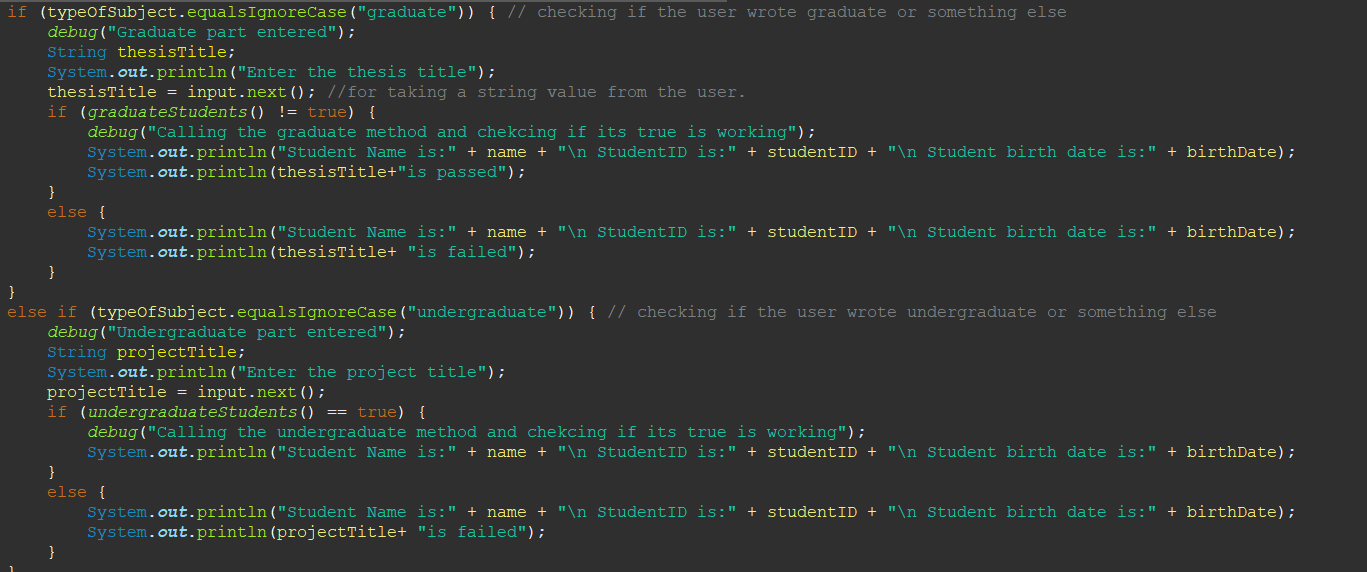
The debugging process that I have used to find out the logical errors that I have got:

* + Creating a debugging method void type and it take a arguments (parameter).
  + Declare a variable Boolean type and this variable is constant so you must write final when you declare the variable and the variable equal true.
  + Calling this method after all the if statements or loop statements to check if the code enters the statements.
  + Run your code if there are no logical errors the code will enter these statements then will call the debug method then print out a message.

Sample from my code on the debugging process that I have used:

* I wrote a function called debug that includes one if statement that prints the value of the argument (parameter) if it is true:



* The code was then run four times after four if statements to see whether there were any logical errors. If there were, the code would call the debug function and output a notice. If the code didn't enter the if statements, it means there was a mistake prior to those statements, in which case the code would have called the debug function.
* Now that the code has been executed, I can see if it calls the debug function. If it does, then the code has entered the if statement, which implies I have no issues with this section of the code.
* As we can see, even though the code was built correctly, something went wrong when I ran the program. Line 27's if statement has a logical error since the code did not enter it; otherwise, it would have called the debug function and printed a notice, as was the case with the other if statements.
* Text

  Description automatically generatedSo, when I went to that if statement, I found out that I wrote the condition wrongly. After I fix the logical error, they must call the debug method after that if statement.

Text

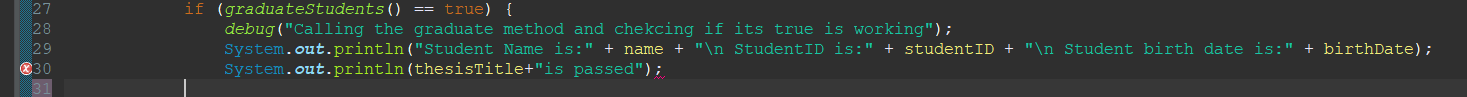
Description automatically generated

* As we can see, thanks to the debugging process, the code now executes without logical issues.

### 3.3.2: different error types that you have dealt with and how you and how solved them:

* Syntax error: is an error made in the syntax of a coding or programming language. The compiler will detect syntax error. The programmer must correct them before the program is built and launched.

Examples of the syntax error that I have dealt with:



Here I forgot to close the curly bracket and the IDE give me an alert that I forgot to close the curly bracket.



Here I forgot to end the line with simi column, and the IDE gave me an alert that I did a syntax error.

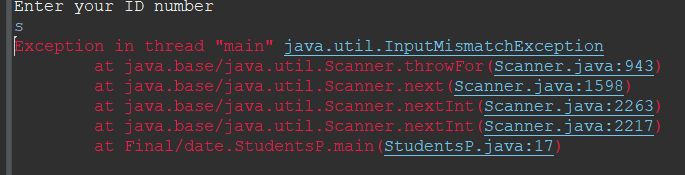
* Text

  Description automatically generatedLogical error: When I execute the code, everything goes as it should, but the result is not what I had in mind.

In this picture we can see that the code runs normally but in the last line the code must give us “t1 is passed” but it did the opposite because I did a logical error in the code.

* Runtime error:

Runtime errors are errors that happen while a program is being executed. As there is nothing grammatically incorrect about these mistakes, they cannot be found during compilation.



As we can see here when the code asked for ID number the code was waiting for a integer but the user entered a character.

## 3.4: Evaluate how the debugging process can be used to help develop more secure, robust applications:

Debugging: In computer programming and engineering, debugging is a multi-step process that entails locating a problem, tracking down its cause, and then fixing the issue or finding a workaround.

When a developer can find and recreate a code fault in a computer program, the debugging process in software development gets started. The debugging process begins when the programmer start writing the code and it continues in phases as code is coupled with other programming units to create a software product.

Examining the code's logs and using a standalone debugger tool or the debug mode of an integrated development environment might be helpful for finding issues. One of the techniques is to set a "breakpoint" and run the program until it stops.

Debugging is a vital aspect of figuring out why an operating system, application, or software is acting strangely. Even if developers adhere to the same coding standards, it is still possible that a new software application will have defects. In many circumstances, the process of debugging a new software program might take longer than the program itself. Inevitably, the flaws in the most heavily used software components are discovered and addressed first.

One of the most crucial steps in developing and implementing applications is debugging. Because you will be able to lower the likelihood that you would create a code that has a vulnerability, making it simpler for hackers to hack the code. Also, debugging reduces human error while building the application. As a result, you will be able to create apps that are more robust and secure.

## 3.5: coding standards:

Code standards: A set of guidelines, and best practices for writing cleaner, easier-to-read, and more error-free code there are many standards we can follow but I will mention 5 of them that I have used while I was writing my code:

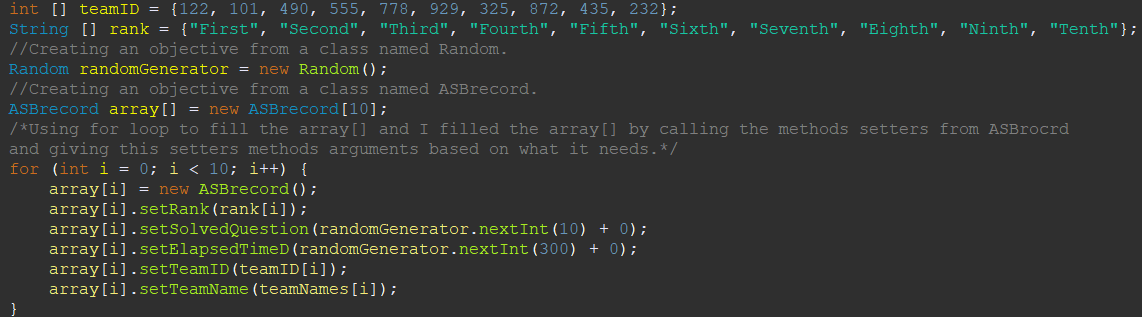
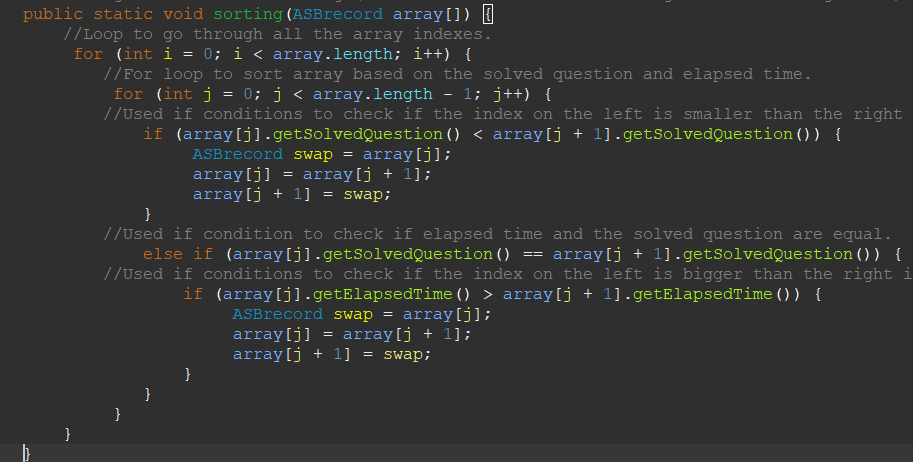
* Naming conventions:

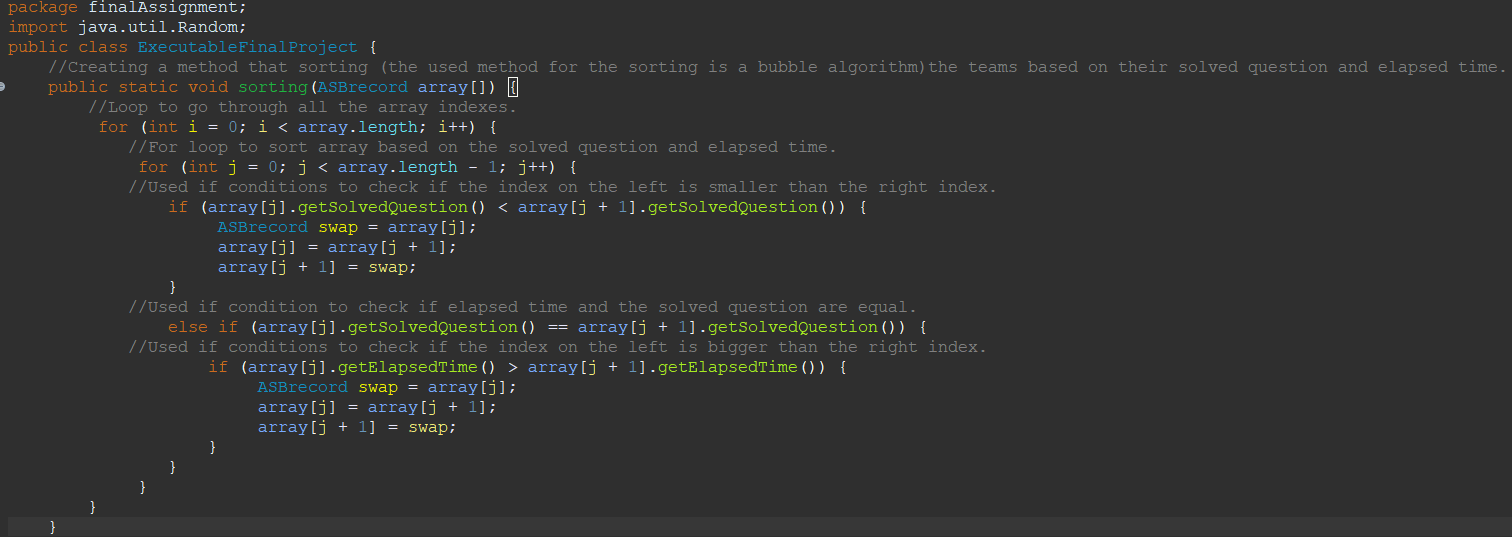
All classes must be named using PascalCase (UpperCamlCase). Additionally, the variables must be named using (camelCase).

* Indentation: provide a space at the start of a code line after opening the curly brackets. The usage of indentation helps to make the code more comprehensible, understandable, and trackable.

****

* ****Documentation the code by using the comments to make the code more understandable for the readers to know what is happening in the code.
* Stay away from lines. Humans find it simpler to read blocks of lines that are both short and lengthy on the horizontal axis.
* Text

  Description automatically generatedSteers clear of coding that is too difficult to decipher: Code needs to be simple to understand. Maintenance and bug fixing are difficult and expensive because of the sophisticated code.
* Every variable needs to have a meaningful and evocative name that explains why it is being used. If one identifier is used for several reasons, this is not possible, and the reader may become confused as a result. Furthermore, it makes future improvements more challenging.

Text

Description automatically generated

# References:

## Part 1 references:

Uri.edu. (2010). Computer Programming. [online] Uri.edu.com Available at: <https://homepage.cs.uri.edu/faculty/wolfe/book/Readings/Reading13.htm> [Accessed 5 January. 2023]

www.cseworldonline.com. (n.d.). Stages of program development process (Program development steps) || Online. [online] CseWorld.com Available at: <https://www.cseworldonline.com/articles/stages-of-program-development-process.php> [Accessed 5 January. 2023]

Downey, L. (2019). Algorithm Definition. [online] Investopedia.com Available at: <https://www.investopedia.com/terms/a/algorithm.asp> [Accessed 6 January. 2023]

Heusser.M. What is debugging? [online] TechTarget.com Available at: <https://www.techtarget.com/searchsoftwarequality/definition/debugging> [Accessed 6 January. 2023]

Engati. (n.d.). *Brute-force search*. [online] Engati.com Available at: <https://www.engati.com/glossary/brute-force-search> [Accessed 6 January. 2023]

## Part 2 references:

DeepSource. What is Procedural Programming? [online] DeepSource.com Available at: <https://deepsource.io/glossary/procedural-programming/> [Accessed 11 January. 2023].

Cocca, G. (2022). Programming Paradigms – Paradigm Examples for Beginners. [online] freeCodeCamp.org. Available at: [https://www.freecodecamp.org/news/an-introduction- to-programming-paradigms/](https://www.freecodecamp.org/news/an-introduction-%09to-programming-paradigms/) [Accessed 11 January. 2023]

GeeksForGeeks (2018). Introduction of Programming Paradigms - GeeksforGeeks. [online] GeeksforGeeks.com Available at: [https://www.geeksforgeeks.org/introduction- of-programming-paradigms/](https://www.geeksforgeeks.org/introduction-%09of-programming-paradigms/) [Accessed 11 January. 2023]

developer.mozilla.org. Object-oriented programming - Learn web development | MDN. [online] developer.mozilla.org Available at: [https://developer.mozilla.org/en- US/docs/Learn/JavaScript/Objects/Object-oriented\_programming](https://developer.mozilla.org/en-%09US/docs/Learn/JavaScript/Objects/Object-oriented_programming) [Accessed 11 January. 2023]

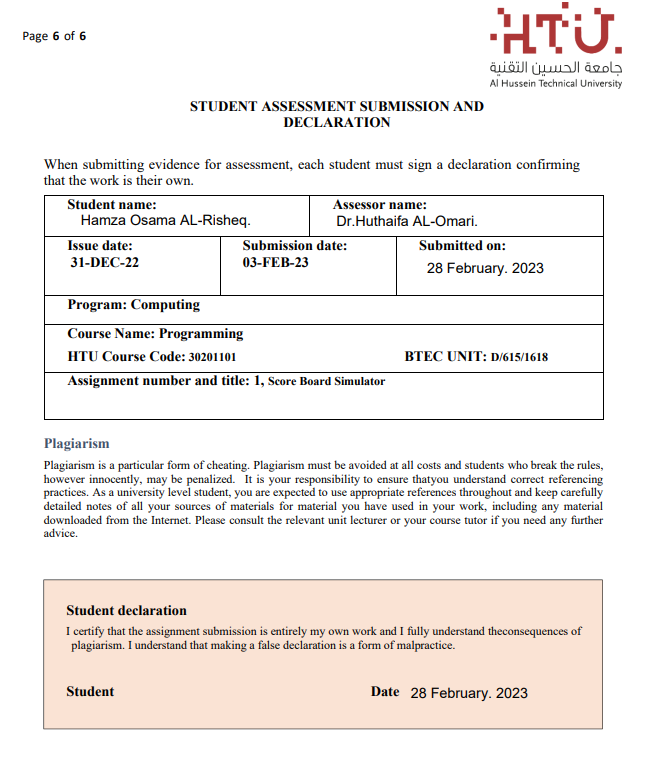
## Part 3references:

Martinez, A. (2020). 5 reasons why you need an IDE (and how it can save you so much time) | [online] ProstDev Blog. Available at: <https://www.prostdev.com/post/5-reasons-why-you-need-an-ide> [Accessed 7 January. 2023]

walker, A. (2023). What Is an IDE? How It Helps Developers Code Faster. [online] g2.com. Available at: <https://www.g2.com/articles/ide> [Accessed 7 Jan. 2023].

Heusser, M. (2022). *What is debugging?* [online] SearchSoftwareQuality. Available at: <https://www.techtarget.com/searchsoftwarequality/definition/debugging> [Accessed 21 Feb. 2023].

A picture containing sitting, plant

Description automatically generatedDeclaration form: