Task 1

1. A brief explanation about what the selected application is and what it does

A brief explanation about the selected application is a program to calculate the training fees for athletes at North Sussex Judo. It is a small local judo training company. Its purpose is to calculate training expenses for athletes who are registered in their program automatically.

The application will display the information about the North Sussex Judo, rules, training plan with fees based on the levels like beginner, intermediate, and elite. Not only that, it allows the users to input the athlete information whether admin, coaches, or athletes such as name, training plan, weight information, competition details, and optional private coaching hours.

After the information has been collected the program or application will check the validation based on the input of users and whether it is acceptable or not if the inputs are all validate the program will continue to the next step which will compute each athlete’s total training and competition cost based on the input. After that, the program will show the result comparison between the athlete’s current weight and competition weight category and response with the feedbacks such as the users are within the middleweights or lightweight.

So, to conclude, the application called North Sussex Judo simplifies the management of athlete registrations, fees, and an overview of training expenses and weight category comparisons so it can assist in the management of their financial resources and judo training program.

1. Explain briefly Procedural, Object Oriented and Event driven Paradigms

Programming paradigms refer to the categorization, approach, or style of programming. There are a lot of paradigms but in these days, most popular and widely used paradigms are declarative, procedural, object-oriented, functional, and imperative paradigms.

To explain briefly about the Procedural, Object Oriented and Event driven paradigms, I can say that Procedural Programming means the program is organized into different procedures or routines which are the sequences of computational steps. It carries out the tasks step by step while following predetermined set of instructions. Procedural programming languages include C, Pascal, and BASIC.

Secondly, Object Oriented Programming which is also known as OOP in programming is a method where the programs are arranged around the objects and represent the actual entities in the real world and contain data and activity. And, OOP includes concepts like inheritance, polymorphism, and encapsulation. Object-oriented programming languages include Java, C++, Python, and Ruby.

And Lastly, Event driven programming is the method that the program flow is determined by external events such as user actions, sensor outputs or communications from other programs. And these programs response to events by making callbacks or triggering event handlers. Event-driven programming is often associated with web development languages like JavaScript.

1. What programming paradigms you use for this project? Give example of programming language which will be used to implement the given scenario

After thorough analysis, the Object-Oriented Programming (OOP) paradigm is determined to be the best fit for this project. Object-oriented programming (OOP) provides an organized method for developing software by structuring code around objects, which are collections of data and action. This encapsulation precisely matches the needs of the North Sussex Judo software solution, enabling the modeling of real-world things in a natural and straightforward manner.

The programming language used for this project is Java because is suitable for implementing an OOP-based solution into practice. Java is well known for its stability, cross-platform compatibility, and broad support of object-oriented programming (OOP). Because of its extensive feature set, which includes classes, inheritance, polymorphism, and encapsulation, developers are able to produce software that is scalable, modular, and maintainable.

With Java's OOP features, we can create an extendable and adaptable system that accurately represents the complex interactions and relationships seen in the North Sussex Judo area. Most importantly, this method encourages code reuse and modularity in addition to improving code structure and readability, so it will surely result in more effective development and simpler maintenance.

So, in conclusion, the Java programming language and the OOP paradigm offer a strong basis on which to develop the North Sussex Judo software solution.

Task 2

1. Briefly explain 5 features which are available in IntelliJ or Eclipse, which you use to build this application.

To build this application, we need to use the 5 essential features that are available in IntelliJ or Eclipse such as code completion, syntax highlighting, debugging tools, version control integration and refactoring capabilities.

Because, Code completion is a tool that increases the efficiency of developers by generating suggestions for variables, methods, and code snippets as they type. Plus, this function helps eliminate syntax errors by offering contextually appropriate ideas, which not only saves time by eliminating the need to manually write out lengthy or complex code constructions.

And, when writing the code, syntax highlighting makes it easier to read and understand because it is a visual feature that helps the developers able to read and comprehend code by assigning distinct colors or styles to different components such keywords, comments, strings, and identifiers. Plus, with this, developers can rapidly spot the any syntax problems, keywords, and other significant aspects by highlighting various areas of the code using separate colors.

Also, with the help of debugging tools like IntelliJ or Eclipse, developers may effectively find and correct mistakes in their code and they can also watch the execution flow, examine the state of variables, identify the source of defects, and more. Moreover, features like breakpoints, step-through execution, variable inspection, and stack trace analysis are the best debugging tools among them.

Version control integration promotes collaborative development by managing code changes and improving team cooperation. So, with integrated version control systems like Git, SVN, or Mercurial, developers can effortlessly handle conflicts within the IDE, monitor changes, manage branches, and merge code changes.

Lastly, refactoring capabilities assist to improve maintainability by reorganizing code without altering its exterior behavior. Not only that, refactoring tools assist developers in removing code duplication, simplifying code, and guaranteeing code consistency by automating repeated refactoring and enforcing best practices for coding. Thus, this can result in software solutions that are very easier to maintain.

1. Explain briefly how using an IDE is better than not using an IDE.

Using an Integrated Development Environment (IDE), such as IntelliJ or Eclipse, provides various benefits over not using one because debuggers, build automation tools, code editors, and other tools specific to software development and 5 features shown in above are all included in integrated development environments (IDEs). Furthermore, IDEs usually support a wide range of programming languages and frameworks, and this is giving the developers to flexibility. In software development projects, productivity, code quality, and teamwork are all improved by utilizing an IDE.

These are the some of the benefit of using IDE.

1. **Unified Development Environment:** IDEs offer an integrated environment where developers can collaborate on several projects at once, including version control management, testing, debugging, and coding, all from inside one interface.
2. **Integrated Debugging Tools:** Debuggers are pre-installed in IntelliJ Eclipses (IDEs) and enabling developers to efficiently examine and debug their code.
3. **Build automation support:** Software application compilation, testing, and deployment are made easier with the help of integrated development environments (IDEs) and their built-in build automation tools.

In conclusion, there are several advantages to using an IDE, including an integrated development environment, advanced code editing abilities, and smooth version control and debugging integration. It could, however, also have disadvantages including high resource requirements, a challenging learning curve, and needless complexity. But, in this case, using the IDE can lead the North Sussex Judo software to the success.