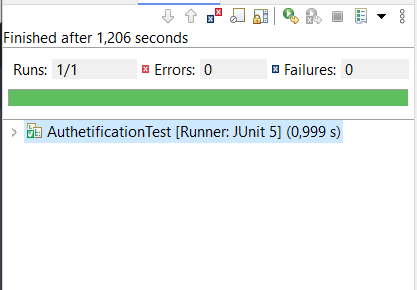
JUNIT Tests

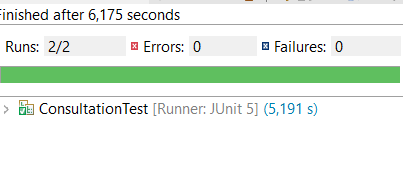
## Authetification Test:

This section describes a test for the Authentifier class using the JUnit 5 framework. The test ensures that the graphical interface is rendered correctly by setting the Nimbus look-and-feel for consistency and displaying the Authentifier window in an undecorated mode. This approach validates the proper initialization and presentation of the user interface, aligning with the expected design and usability standards.



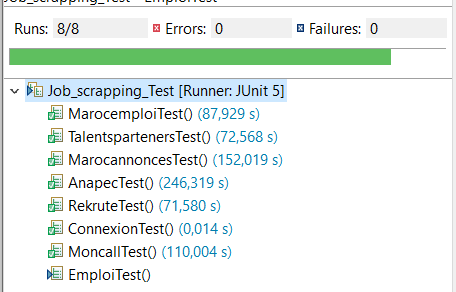
## ConsultationTest :

This section highlights the testing of the Consultation module to ensure the proper functionality of its graphical user interface components. The tests utilize the JUnit 5 framework and include initializing different consultation windows, Consult and Consult2, under the Nimbus look-and-feel to maintain visual consistency. Each test involves creating an instance of the respective consultation class with a given parameter, verifying that the interfaces are rendered and initialized correctly. These tests play a vital role in ensuring the robustness and usability of the application's consultation features.



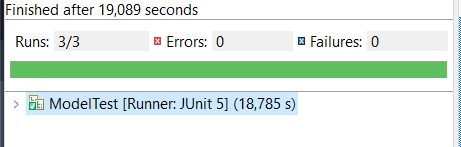
## Job scrapping Test:

This section focuses on the testing process for the Job\_scrapping module, which is responsible for web scraping job listings from various platforms. Using the JUnit 5 framework, a series of unit tests were implemented to validate the functionality of individual scrapers and ensure their proper integration. Each test targets a specific scraper, such as Anapecscrapper, emploiScrapper, rekruteScrapper, and others, by invoking their respective scraping methods to extract data from designated websites. Additionally, the ConnexionTest verifies the successful establishment of a database connection, ensuring that scraped data can be stored effectively. These tests are critical for confirming the reliability and correctness of the scraping process in the context of automated data collection.



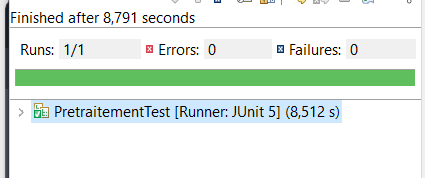
## ModelTest :

This section focuses on the testing of the Model class, which encapsulates core machine learning functionalities such as classification, regression, and clustering. Using the JUnit 5 framework, separate tests were implemented to validate each method (classifier, regresser, and clustering) in isolation. These tests ensure that the respective machine learning models are invoked correctly and perform as expected under various scenarios. By verifying the proper execution of these core functionalities, this testing process contributes to the reliability and accuracy of the system's predictive and analytical capabilities.



## Pretraitement Test

This section addresses the testing of the DataPreprocessor class, specifically its load\_And\_ProcessData method. The test validates the data preprocessing pipeline, ensuring that data loading, cleaning, and transformation are executed correctly and efficiently. By using the JUnit 5 framework, this test guarantees that the preprocessing steps meet the requirements for downstream tasks, such as machine learning or data analysis, while maintaining data integrity and consistency throughout the pipeline.



## Visualisation Test

This section focuses on testing the Visualisation module, which is responsible for generating graphical representations of data. Using the JUnit 5 framework, the test validates the creation of various charts, including SalaryByContractTypeChart, SalaryBySectorChart, and SalaryExperienceLineChart. This ensures that the visualization components are initialized correctly and capable of rendering accurate and meaningful insights. Verifying these charts is essential to confirm the reliability of the data visualization process, which plays a crucial role in enhancing the interpretability and presentation of analytical results.

