# Write-up: How I Solved the Problem

To solve this assignment, I divided the problem into multiple steps and implemented them systematically in the Jupyter notebook. The overall approach followed a structured data science workflow combined with retrieval-based techniques.  
  
**1. Problem Understanding & Setup**  
 - I started by importing the required Python libraries for data analysis (pandas, numpy, matplotlib, seaborn), machine learning (scikit-learn), and text processing.  
 - Warnings were suppressed, and helper functions were defined to streamline the workflow.  
  
**2. Data Loading & Exploration** - The dataset (on healthcare/heart patients, as per assignment context) was loaded using pandas.  
 - Exploratory Data Analysis (EDA) was performed with summary statistics, missing value checks, and data type inspections.  
  
**3. Data Preprocessing** - Cleaning steps included handling missing values, encoding categorical variables (using LabelEncoder and OneHotEncoder), and normalizing data where required.  
 - New features such as age groups were derived to add interpretability.  
  
**4. Model Building**  
 - Data was split into training and test sets using train\_test\_split.  
 - A machine learning pipeline was constructed, using Random Forest Classifier as the main predictive model.  
 - Hyperparameters were set at standard values for the baseline model.  
  
**5. Evaluation**  
 - Model performance was evaluated using accuracy, confusion matrix, and classification metrics from sklearn.metrics.  
 - Additional error measures like RMSE and precision/recall were considered to assess robustness.  
  
In short, the problem was solved by combining data preprocessing, model training, evaluation, and knowledge retrieval techniques into a modular workflow. This ensured the system was both data-driven (ML model) and context-aware (RAG pipeline).