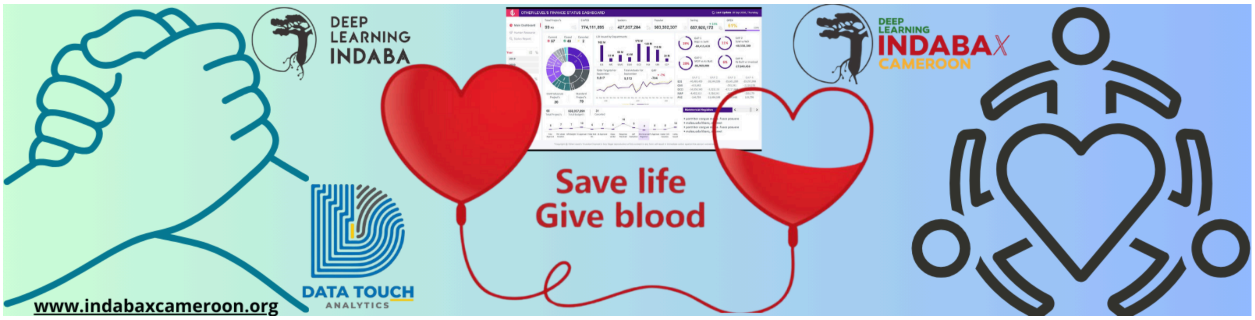


Blood Donation Campaign Dashboard

Objective:

We are looking to build a comprehensive dashboard, fully implemented in Python, for the visualization and analysis of blood donation campaign data. The dashboard should showcase the richness of the dataset and provide valuable insights for optimizing blood donation campaigns.

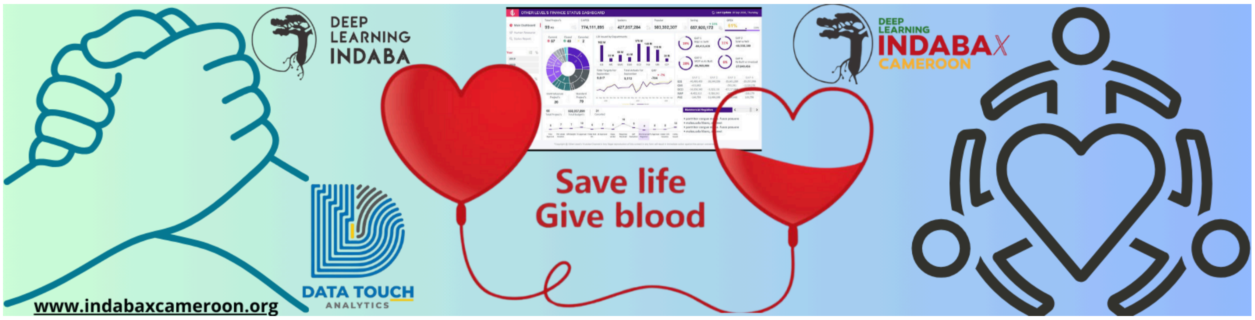
Participants will need to create a **dashboard in Python** that addresses key questions for campaign organizers, helping them make data-driven decisions to improve the success of future blood donation campaigns.

Application link: https://forms.gle/vvW7JDH3GTDHYBL56

Key Features to be Implemented:

The dashboard should provide answers to the following questions, leveraging the provided dataset:

1. **Map Donor Distribution:**
   - Visualize the geographical distribution of blood donors based on their residential area (i.e., "Arrondissement de résidence" and "Quartier de Résidence").
   - Use a map to plot the locations of donors, highlighting regions with high or low participation.
2. **Health Conditions & Eligibility:**
   - Visualize the impact of health conditions (e.g., hypertension, HIV, asthma, diabetes) on blood donation eligibility.
   - Create charts or graphs that display the number of eligible vs. non-eligible donors based on these conditions.
3. **Profiling Ideal Donors:**
   - Use clustering techniques to group donors into similar profiles based on demographic and health-related features (e.g., age, gender, profession, health conditions).
   - Generate insights into the characteristics of the ideal blood donor.



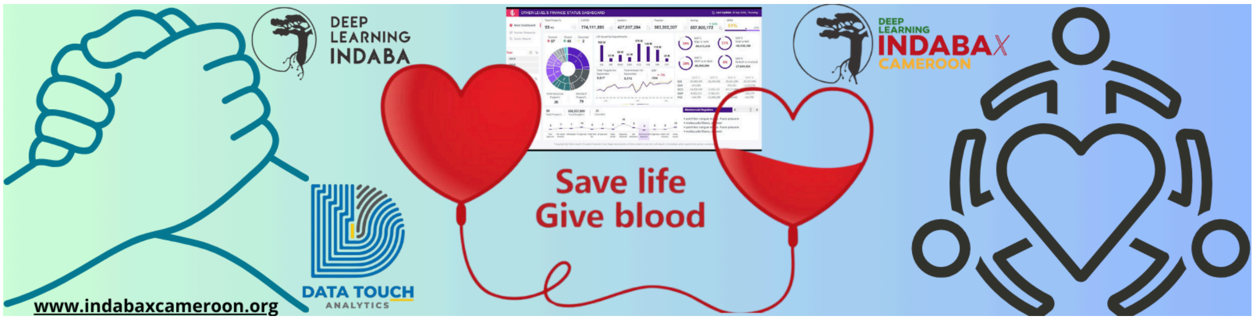
4. **Campaign Effectiveness:**
    - Analyze past campaigns by examining the donation date and other demographic factors.
    - Visualize trends, such as the time of year when blood donations are highest or which demographics contribute more to campaigns.
    - Identify patterns in donor behavior over time.
5. **Donor Retention:**
    - Investigate donor retention by analyzing how often individuals return to donate blood.
    - Use demographic data to determine which factors (e.g., age, profession, region) correlate with repeat donations.
6. **Survey/Feedback Sentiment Analysis:**
    - If feedback text data is available (e.g., in the "Si autres raison préciser" column), perform sentiment analysis on the textual feedback provided by donors.
    - Classify the feedback into positive, negative, or neutral categories and visualize sentiment trends over time or by demographic group.
7. **Blood Donation Eligibility Prediction Model (API):**
    - As an additional challenge, participants are asked to build a machine learning model that predicts the eligibility of new donors based on demographic and health data.
    - This model should be wrapped in an API to allow easy integration into the dashboard for real-time predictions.

🚀 This project aims to harness the power of data to improve the management of blood donation campaigns and enhance the impact of collection initiatives. Thank you to all participants for your commitment! 🙌

Expected Deliverables:

A. **A Fully Functional Python Dashboard:**
    - The dashboard should be interactive, displaying all the visualizations and insights mentioned above.
    - Candidates should use Python libraries such as Dash, Streamlit, or Plotly to create the dashboard interface.
    - The dashboard should support interactivity, allowing users to filter and drill down into the data (e.g., by region, age, health condition, etc.).

B. **Codebase:**



  - Provide all code for the dashboard, including data cleaning, visualization, and modeling.
  - The code should be well-commented and modular.

C. **Machine Learning Model:**
  - If candidates choose to implement the bonus feature, the machine learning model for predicting blood donation eligibility should be packaged as an API.
  - The API should accept inputs (e.g., age, health condition, profession) and return a prediction of whether the individual is eligible to donate blood.

D. **Documentation:**
  - A comprehensive README file explaining the functionality of the dashboard, the tools used, and any assumptions made during development.
  - Clear instructions on how to run the dashboard and interact with the visualizations.
  - If applicable, explain how to use the prediction model API.

Suggested Tools and Libraries:

Here are some recommended tools and libraries that participants can use to build the dashboard and implement the required features:

Python Libraries for Data Analysis and Visualization:

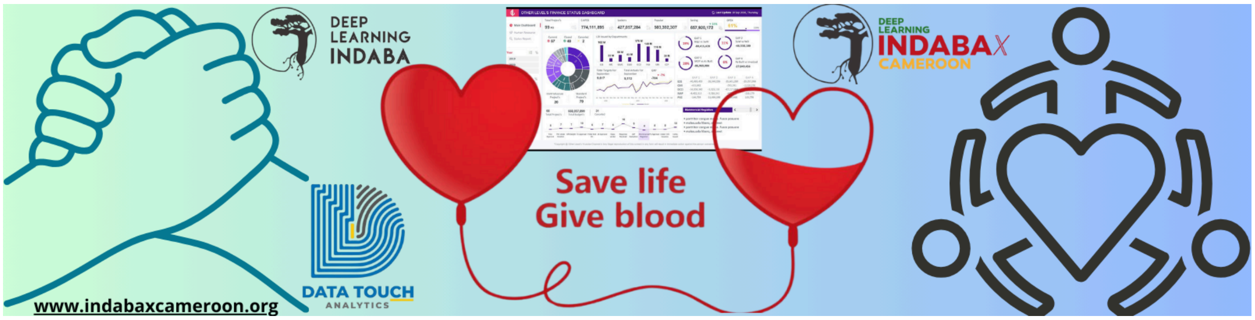
- Pandas: For data manipulation and cleaning.
- Matplotlib / Seaborn: For creating static visualizations (e.g., bar charts, line graphs).
- Plotly: For interactive visualizations, including maps, bar charts, and scatter plots.
- Dash / Streamlit: For building interactive web applications in Python.
- Geopandas: For geospatial analysis (mapping donor locations).
- NLTK / TextBlob: For sentiment analysis of textual feedback.

Machine Learning:

- Scikit-learn: For building machine learning models, such as Logistic Regression, Random Forest, or SVM.
- Flask / FastAPI: For building a REST API around the machine learning model.

Map Visualization:

- Folium: For creating interactive maps based on geospatial data.
- Plotly: Also supports geographical plotting with choropleth maps.



Other Tools:

- Jupyter Notebooks: For exploratory data analysis and prototyping.
- Heroku or PythonAnywhere: For deploying the dashboard and API (if needed).

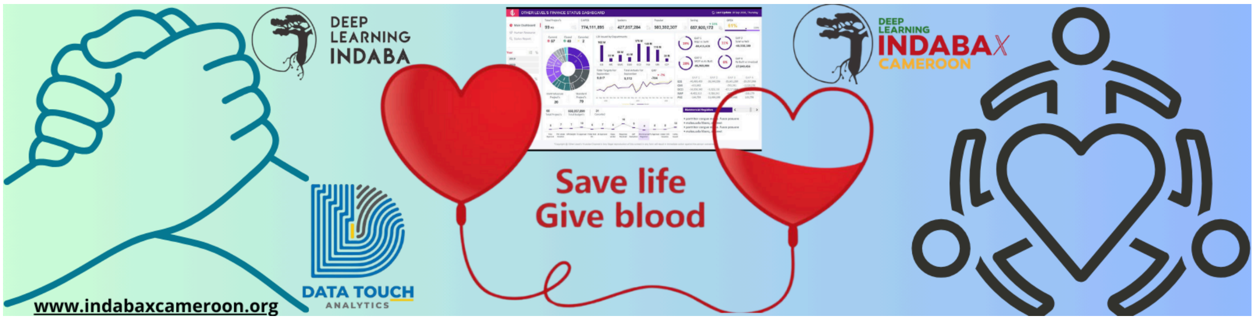
Judging Criteria:

The submissions will be evaluated based on the following criteria:

1. **Functionality**:
   - Does the dashboard meet all the required features and functionality as outlined in the competition description?
   - Are the visualizations interactive and easy to understand?
2. **Usability**:
   - Is the dashboard user-friendly and intuitive? Can users easily filter data and navigate through the insights?
   - Is the user interface clean and professional?
3. **Data Insights**:
   - How well does the dashboard showcase the richness of the dataset? Are the insights meaningful and actionable for blood donation campaigns?
   - Are the visualizations clear and easy to interpret?
4. **Innovation**:
   - Does the submission go beyond the basic requirements, adding creative features or advanced visualizations?
   - If the bonus feature is implemented, how well does the machine learning model integrate with the dashboard?
5. **Code Quality**:
   - Is the code well-organized, modular, and properly commented?
   - Is the code reusable and maintainable?

Key Notes for Participants:

- ★ **Data Privacy**: Discuss how you ensured that any sensitive or personal information in the dataset is anonymized or treated according to the appropriate guidelines.



- **Data Imbalance**: There might be cases where certain classes or categories in the dataset are imbalanced (e.g., eligibility classes, health conditions). Consider this, as well as missen data while building models or interpreting results.
- **Complexity**: While building the dashboard, ensure that the focus is on creating clear, actionable insights. Avoid overcomplicating the design or features.
- **Real-time Prediction**: For the bonus challenge, ensure the eligibility prediction model is fast and can handle real-time input from the user.
- **Dataset details** can be found here

Conclusion

This competition is a unique opportunity to explore the power of data visualization and machine learning in a real-world context. By building a dashboard that allows campaign organizers to make informed decisions, participants will play an integral role in optimizing future blood donation drives.
💡The challenge combines data analysis, interactive visualization, and machine learning, providing a rich environment for learning and growth.

🚀 Good luck to all participants! 🙌