**Tasks:**

1. **Task 1:** Split similar functions in different .py file and make them run from main.py file.
2. **Task 2:** I have created a dataset which has the following features.

**Features**:

* 1. Date\_of\_cr
  2. Customer\_id
  3. First\_name
  4. Last\_name
  5. Gender
  6. Email
  7. Phone
  8. Age
  9. Occupation
  10. Income
  11. Marital\_status
  12. Principle\_Amount
  13. Interest\_rate
  14. Processing\_fee
  15. Tenure
  16. EMI
  17. First\_duedate
  18. Tenure\_rem
  19. Principle\_amt\_rem
  20. Last\_duedate
  21. Next\_duedate
  22. Loan\_status
  23. Loan\_purpose

Challenge here is to create a payment history for all customers or just create two more features which will be “Last\_payment\_received” & “Late\_Charge”. If the customers last payment date is after the due date, there will be a specific amount of late charge or else will show zero.

**Note:** The original dataset of a bank is a running dataset which keeps on getting updated each hour or day. But since there is no way to get a running data and since we have a manually created dataset which is not getting updated each hour or day, the conversation with the bot will be time bounded

1. **Task 3:** Create functions, which when called should ask a set of questions and capture all answers. Which will be further shared as input for a specific model or function.
2. **Task 4:** Create functions for authorization check, new loan application, top-up loan application, foreclosure. These functions will take data as input from task 3 function and do the need full. Some functions will also update the result in a new database on mongodb.
3. **Task 5:** Create ML model for ‘eligibility check’, ‘recommendation of other banking product’ which will be based on customers previous banking relations and ranking/score.
4. **Task 6:** Create DL model for ‘Sentiment Analysis’ to check if the customer is satisfied with the resolution provided. {We can also achieve this task of checking satisfaction with a simple ‘Yes/No’ answer as well, but that will make the bot more rule based, hence letting the user answer the way he/she wants and then lets analyze the out come in positive or negative sentiment)
5. **Task 7:** Create small talks data, and create a function or program which lets the bot answer any small talks initiated by user, up-to two questions and right after each answer it redirects the conversation to previous context, by asking the same question which the user did not answer and took the conversation to small talks.

(**Example**:

Bot: May I know your Date of Birth ?

User: How is the weather out there?

Bot: Its fine out here. May I know your Date of Birth ? Please.

User: How are you doing?

Bot: I’m good. However, I would need your Date of Birth to further proceed with the application?

User: Where are you located?

Bot: Thank you for asking. However, my motive is to first solve your product related query and since you need a loan, I would need a your Date of Birth to further proceed.

User: Are you machine?

Bot: I am sorry but I have to end this conversation now, if you need help you may speak or text me. Goodbye.

**After this message the bot will close, if user wants to chat, user will have to reopen the chatbot and start and new conversation.**

1. **Task 8:** Create a function which will alert before changing the context