Consumer Lending - Prescreening -Onboarding Agent

Install Packages

```
In [ ]:  !pip install autogen
         !pip install gradio
```

Set API Keys

```
In [ ]:  from google.colab import userdata
         import os
         os.environ['OPENAI_API_KEY'] = userdata.get("OPENAI_API_KEY")
         os.environ['GOOGLE_API_KEY'] = userdata.get("GOOGLE_API_KEY")
```

```
In [ ]:  from autogen import ConversableAgent, LLMConfig

         # Define the LLM configuration (can be OpenAI or another provider)
         llm_config_openai = LLMConfig(api_type="openai", model="gpt-4o-mini")
```

Agent-1: Collects the customer details

```
In [ ]:  profile_info_collection_agent = ConversableAgent(
             name="profile_info_collection_agent",
             system_message='''You are a helpful customer onboarding agent,
             you are here to help new customers get started with our product/service.
             Your job is to gather customer's name and mobile number.
             Do not ask for other information. Return 'TERMINATE'
             when you have gathered all the information.''',
             llm_config=llm_config_openai,
             human_input_mode="NEVER",
         )
```

```
In [ ]:  profile_info_collection_agent.generate_reply([{"content": "Who are you?",
                               "role": "user"}] )
```

Agent-2: Collects the interested products

```
In [ ]:  preference_scan_agent = ConversableAgent(
             name="preference_scan_agent",
             system_message='''You are a helpful customer preference finding agent,
             you are here to help new customers get started with our product.
```

```
    Your job is to gather customer's preferences on
    -Looking for a Home Loan
    -Looking for a Personal Loan
    -Looking for a Education Loan
    -Looking for a Gold Loan
    ''',
    llm_config=llm_config_openai,
    human_input_mode="NEVER",
)
```

In [ ]:
```
preference_scan_agent.generate_reply([{"content": "Who are you?",
                        "role": "user"}] )
```

Agent-3: Loan Documents Agent

In [ ]:
```
loan_documents_agent = ConversableAgent(
    name="loan_documents_agent",
    system_message='''You are a helpful customer service agent.
    You will provide the information on loan processing checklist based on the

    Check List for a Home Loan
    Proof of Identity — Aadhaar Card / PAN Card / Passport
    Proof of Income — Salary slips / IT returns / Bank statements
    Property Documents — Sale deed / Agreement to Sell / Title papers

    Check List for a Personal Loan
    KYC Documents — PAN + Aadhaar / Voter ID
    Income Proof — Last 3 months salary slips / IT returns
    Bank Statement — Last 6 months

    Check List for an Education Loan
    Admission Letter — From recognized college/university
    Fee Structure — Official document from the institution
    Income Proof of Co-applicant — Salary slips / ITR of parent/guardian

    Check List a Gold Loan
    Gold Articles — Jewelry or coins to be pledged
    KYC Documents — Aadhaar and PAN
    Recent Photograph — Passport size photo

    Return 'TERMINATE' when you are done.''',
    llm_config=llm_config_openai,
    human_input_mode="NEVER",
)
```

In [ ]:
```
loan_documents_agent.generate_reply([{"content": "Who are you?",
                        "role": "user"}] )
```

Agent4: Customer Proxy Agent

```python
#This is not a llm agent. It just takes the input from the user
customer_proxy_agent = ConversableAgent(
    name="customer_proxy_agent",
    llm_config=False,
    human_input_mode="ALWAYS",
    is_termination_msg=lambda msg: "terminate" in msg.get("content").lower(),
)
```

```python
customer_proxy_agent.generate_reply([{"content": "Who are you?",
                                       "role": "user"}] )
```

Creating Tasks

```python
chats = [
    {
        "sender": profile_info_collection_agent,
        "recipient": customer_proxy_agent,
        "message":
            "Hello, I'm here to help you get started with loan approval proces
            "Could you tell me your name and mobile number",
        "summary_method": "reflection_with_llm",
        "summary_args": {
            "summary_prompt" : "Return the customer information "
                               "into as JSON object only: "
                               "{'name': '', 'Mobile Number': ''}",
        },
        "max_turns": 2,
        "clear_history" : True
    },
    {
        "sender": preference_scan_agent,
        "recipient": customer_proxy_agent,
        "message":
                "Great! Could you tell me kind of loan you are interested in?"
                "Is it Home Loan or Personal Loan or Education Loan or Gold lc
        "summary_method": "reflection_with_llm",
        "summary_args": {
            "summary_prompt" : "Return the customer information "
                               "into as JSON object only: "
                               "{'loan_type': ''}",
        },
        "max_turns": 2,
        "clear_history" : False
    },
    {
        "sender": customer_proxy_agent,
        "recipient": loan_documents_agent,
        "message": "Give me the checklist for the loan mentioned",
```

```
        "max_turns": 1,
        "summary_method": "reflection_with_llm",
    },
]
```

Initiate chats

```
In [ ]:  from autogen import initiate_chats

         chat_results = initiate_chats(chats)
```

```
In [ ]:  #Print out the summary
         for chat_result in chat_results:
             print(chat_result.summary)
```

```
In [ ]:  for chat_result in chat_results:
             print(chat_result.cost)
             print("\n")
```

```
In [ ]:
```