# PREREQUISITES

**To complete this project:**

- IBM account
- Watson Assistant

**Prior Knowledge:**

We should know the following Concepts:-

- **What is a chatbot?**

  A chatbot is a computer program that simulates human conversation through text. Chatbot, short for chatterbot, is an artificial intelligence (AI) feature that can be embedded and used through any major messaging application.

- **Watson Assistant:- Intents, Entities, Dialogs**

  **Watson Assistant** lets you build conversational interfaces into any application, device, or channel. Add a natural language interface to your application to automate interactions with your end users. Common applications include virtual agents and chatbots that can integrate and communicate on any channel or device.

  **Intent:** An intent represents the purpose of a user's input. You define an intent for each type of user request you want your application to support.

  **Entity:** An entity represents a term or object that is relevant to your intents and that provides a specific context for an intent. You list the possible values for each entity and synonyms that users might enter.

  **Dialogue:** Dialogue is a branching conversation flow that defines responses to the defined intents and entities. You use the dialogue builder in the tool to create conversations with users to respond.

- **What is Natural Language Processing?**

  Natural language processing (NLP) refers to the branch of computer science—and more specifically, the branch of artificial intelligence or AI—concerned with giving computers the ability to understand the text and spoken words in much the same way human beings can.

- **What is Deep Learning?**

  Deep learning is a subset of machine learning, which is essentially a neural network with three or more layers. These neural networks attempt to simulate the behavior of the human brain—albeit far from matching its ability—allowing it to "learn" from large amounts of data.

- **What is Python - Flask?**

  Flask is a web framework that provides libraries to build lightweight web applications in python. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework-related tools. It is based on the WSGI toolkit and the jinja2 template engine.