Machine learning

1. What is the difference between supervised, unsupervised, and reinforcement learning?
2. Explain overfitting and underfitting. How can they be avoided?
3. What is the bias-variance tradeoff?
4. Define precision, recall, F1 score, and accuracy.
5. What is the difference between parametric and non-parametric models?
6. Explain the difference between a generative and a discriminative model.
7. What are the main steps in a machine learning project workflow?
8. Explain feature scaling and normalization.
9. How does a decision tree work?
10. What is the difference between bagging and boosting?
11. Explain the working of the k-nearest neighbors (KNN) algorithm.
12. How does linear regression work?
13. Explain logistic regression and its assumptions.
14. How does a support vector machine (SVM) classify data?
15. What is PCA, and when is it used?
16. Explain how K-means clustering works.
17. What is a random forest, and how does it prevent overfitting?
18. Compare gradient boosting machines and XGBoost.

Model evaluation:

1. How do you evaluate the performance of a classification model?
2. What are confusion matrices, and how do you interpret them?
3. How do you handle an imbalanced dataset?
4. What is cross-validation, and why is it important?
5. When would you prefer ROC-AUC over accuracy?

Feature Engineering:

1. What is feature selection, and why is it important?
2. How do you handle missing data?
3. Explain one-hot encoding and label encoding.
4. What is the difference between L1 and L2 regularization?
5. How do you deal with categorical features in a dataset?

Optimization and Loss Functions

1. What is gradient descent?
2. Explain the difference between stochastic gradient descent and batch gradient descent.
3. What are common loss functions for regression and classification tasks?
4. What is the role of a learning rate in training a model?
5. Explain the concept of backpropagation.

Neural Networks and Deep Learning:

1. What is a perceptron?
2. Explain the architecture of a neural network.
3. What is the difference between CNNs and RNNs?
4. What are activation functions, and why are they needed?
5. Explain dropout and its purpose.
6. What is transfer learning?
7. What is batch normalization?
8. Explain vanishing and exploding gradients.

Natural language processing (NLP):

1. What are word embeddings? Explain Word2Vec, GloVe, and FastText.
2. How does TF-IDF work?
3. What is the difference between stemming and lemmatization?
4. What is a language model?
5. How do transformer-based models like BERT and GPT work?
6. What are the challenges of working with text data?
7. What is NLP, and how is it different from text mining?
8. What are stop words, and why are they important in text preprocessing?
9. Explain tokenization and its types.
10. How do you handle out-of-vocabulary (OOV) words?
11. What is the difference between a unigram, bigram, and trigram model?
12. What are the challenges of working with text data?
13. Explain the difference between Bag-of-Words (BoW) and Word2Vec.
14. How does the Skip-gram model work in Word2Vec?
15. What are contextual word embeddings? Compare BERT embeddings with Word2Vec.
16. What is the difference between static and dynamic word embeddings?
17. Explain the concept of cosine similarity and its use in NLP.
18. What is named entity recognition (NER), and how does it work?
19. How do you perform sentiment analysis?
20. What is part-of-speech (POS) tagging?

21. Explain dependency parsing in NLP.
22. What is semantic similarity, and how is it calculated?
23. What are embeddings for sentences and documents (e.g., Sentence-BERT, Doc2Vec)?
24. What are pre-trained models in NLP, and why are they important?
25. What metrics would you use to evaluate a classification task in NLP?
26. How do you evaluate the performance of a chatbot or conversational agent?
27. How does topic modeling work? Compare LDA with NMF.
28. Explain the architecture of a chatbot system.
29. What is text summarization? Compare extractive and abstractive summarization.
30. How does question answering work in NLP?
31. What is intent classification and how is it used in virtual assistants?
32. How do you handle biases in language models?
33. What are the privacy and security concerns when working with NLP systems?

Kore.ai

1. What is the Kore.ai platform, and how does it used for building chatbots?
2. What are the core components of Kore.ai Bot Builder?
1. What are the key advantages of using Kore.ai over other chatbot platforms like Dialogflow or Rasa?
2. How do you design dialog tasks in Kore.ai for complex user journeys?
3. Explain the concept of Natural Language Understanding (NLU) in Kore.ai. How does it work?
4. How do you train the NLU model in Kore.ai to improve intent recognition?
5. What are webhook services in Kore.ai? How did you use them in your project?
6. How did you handle account-related queries (e.g., balance inquiry, account statement)?
7. How does Smart Assist work in Kore.ai, and how have you used it in your projects?
8. Explain how you implemented context switching in Kore.ai.
9. What analytics features does Kore.ai provide?

10. How did you measure the success of your chatbot using Kore.ai analytics?
11. What KPIs did you track for your banking/insurance chatbot?
12. How did you use session insights to optimize bot performance?
13. Explain how you handle fallback intents and train the bot based on user utterances.
14. What were some unique use cases in your insurance or banking project, and how did you implement them?
15. Can you describe the Kore.ai project you worked on for the banking/insurance client?
16. How did you design the conversational flows for the bot?
17. What use cases or intents did the bot support for the banking/insurance client?
18. How did you implement role-based authentication or user verification in the bot?
19. How did the bot handle customer KYC (Know Your Customer) processes?
20. How did the bot assist customers with policy purchase, renewal, or claims?
21. What is Smart Assist in Kore.ai, and how does it enhance chatbot functionality?
22. How did you configure Smart Assist for your project?

Python:

1. String- top 30 problem
2. List- top 30 problem
3. Dict- top 20 problem
4. Tupple- top 10 problem
5. Basic python – function , class code samples
6. Pandas- merge data frame, create data frame, joining,concat,filter, group by aggregation ,apply map function,read csv file

SQL: