

1.Tell Me About Your Self.

I'm Mohan Jinkala, a recent Computer Science graduate from SVU College of Engineering with a CGPA of 8.5. During my academic journey, I focused on building strong foundations in Data Structures & Algorithms, Object-Oriented Design, and Machine Learning.

I completed a hands-on training course at ISRO's Indian Institute of Remote Sensing, where I worked on a weather prediction project using machine learning techniques. I applied data preprocessing, feature engineering, and model tuning—achieving over 97% accuracy with Decision Trees. I also built a customer churn prediction system using Support Vector Machines, where I handled imbalanced data using oversampling and tuned hyperparameters to achieve 93% accuracy.

On the system design side, I implemented a scalable Parking Lot System using OOP, SOLID principles, and design patterns like Factory, Singleton, and Strategy.

I've solved over 350 DSA problems on LeetCode and actively participate in coding contests and secured a global rank of 1490 in tcs codevita. I'm passionate about building efficient, scalable solutions—whether it's through core software engineering or intelligent ML models—and I'm now excited to bring that passion into a full-time software development or data science role."

2.Why You Not Attended Round 2 Of CodeVita?

Yes, I was shortlisted for the second round after securing a global rank of 1490 in TCS CodeVita Season 12. Unfortunately, I couldn't attend the next round due to a health issue at the time, and the exam center was quite far, which made it difficult toTravel. It was a tough call, but I made sure to prioritize recovery. That said, I gained a lot from the preparation and the experience itself, and it gave me confidence in competitive coding and problem-solving."

3.Why U choice Decision Tree instead of other.

I experimented with multiple models like knn, logistic regression , and svm but i found that Decision Tree gave the highest accuracy—around 97.3%—on my weather dataset.and other reasons or It's easy to understand, works well with both numerical and categorical data, and doesn't need much data preprocessing. It also helps in understanding how different features affect the prediction."

4.What do you think made Decision Tree work better than other models?"

"I think Decision Tree worked better because it can easily handle non-linear relationships and decision boundaries, which are common in weather data. It also captures feature-based splits well—like if temperature or humidity crosses a certain value—making it a good fit for such rule-based patterns. Plus, it doesn't need much data scaling or transformation, so it performed well out of the box."

5.What is AI?

Artificial Intelligence (AI) means making computers or machines smart enough to do things that usually need human thinking.

What is Generative AI?

Generative AI is a type of Artificial Intelligence that can create new content — like text, images, music, or code — by learning patterns from existing data.

6.Machine Learning (ML)?

ML is a part of Artificial Intelligence where we teach computers to learn from data and make decisions without being directly programmed.

Types: 1.supervised(Learns from labeled data), 2.unsupervised(Learns from Unlabeled data), 3.reinforcement(Learns by trial and error).

6.Deep Learning

DL is a special type of Machine Learning that teaches computers to learn like the human brain — using something called neural networks.

7.What is an Operating System?

An Operating System (OS) is the main software that runs your computer or mobile phone.

It connects you (the user) with the hardware (like CPU, memory, keyboard, etc.) and helps all other programs run.

7.Dbms?

It's software that allows you to create, manage, and interact with databases.

RDBMS stands for Relational Database Management System.

It's a type of DBMS that stores data in tables (also called relations) made up of rows and columns.

Key features of RDBMS:

- **Data organized in tables:** Each table has rows (records) and columns (attributes).
- **Relationships:** Tables can be linked using keys (primary key, foreign key) to represent relationships.
- **Use of SQL:** Structured Query Language (SQL) is used to create, read, update, and delete data.
- **Data Integrity:** Enforces rules like primary keys (unique identifiers) and foreign keys (referential integrity).

SQL is the language you use to interact with RDBMS.

8.Data Analytics?

DA is the process of examining, cleaning, transforming, and modeling data to discover useful information, draw conclusions, and support decision-making.

Key steps in Data Analytics:

1. **Data Collection:** Gathering raw data from various sources.
2. **Data Cleaning:** Removing errors, duplicates, and inconsistencies.
3. **Data Exploration:** Summarizing and visualizing data to understand it.
4. **Data Analysis:** Applying statistical and computational techniques to find patterns.
5. **Interpretation & Reporting:** Translating results into actionable insights.

Types of Data Analytics:

- **Descriptive Analytics:** What happened? (e.g., sales reports)
- **Diagnostic Analytics:** Why did it happen? (e.g., root cause analysis)
- **Predictive Analytics:** What will happen? (e.g., forecasting)
- **Prescriptive Analytics:** What should we do? (e.g., recommendations)

9.Core concepts of OOP:

1. Class

A blueprint or template for creating objects. Defines attributes (data) and methods (functions).

2. Object

An instance of a class. It has specific values for attributes and can perform methods.

3. Encapsulation

Bundling data and methods inside a class, restricting direct access from outside (data hiding).

4. Inheritance

A class can inherit properties and behaviors from another class (promotes code reuse).

5. Abstraction

Hiding complex implementation details and showing only the necessary features.

6. Polymorphism

allows the same method name to do different things based on the object that's using it.

10. What are Design Principles?

Design Principles are guidelines that help developers write clean, maintainable, scalable, and flexible code.

11. What are Design Patterns?

Design patterns are proven, reusable solutions to common problems in software design. They are not code you copy and paste, but general templates or blueprints you can adapt to your own code.

12. What is Concurrency?

Concurrency means doing multiple tasks at the same time (or appearing to). It doesn't always mean tasks run in *parallel*.

13. What is Multithreading?

Multithreading is one way to achieve concurrency. It means dividing a program into multiple threads (lightweight units of execution) that run independently but share the same memory.

14.What is UML (Unified Modeling Language)?

UML is a visual language used in software engineering to design and document software systems. It helps developers and designers understand how different parts of a system work together.

15.What is Pandas?

Pandas is a powerful Python library used for data manipulation, analysis, and cleaning.

16.What is Seaborn?

Seaborn is a Python data visualization library built on top of Matplotlib.

17.What is Matplotlib?

Matplotlib is a popular Python library used for creating visualizations like charts, plots, and graphs. It gives you full control over how your plots look — from basic line charts to advanced custom visualizations.

Matplotlib vs Seaborn:

Feature	Matplotlib	Seaborn
Level	Low-level (more control)	High-level (easier, better visuals)
Customization	Very customizable	Built-in styles
Usage	Detailed and manual	Simplified for stats & DataFrames

18.What is Scikit-learn (sklearn)?

Scikit-learn (or **sklearn**) is a powerful machine learning library in Python.we use it in various places in machine learning process.

Ex:--

1. Import a model

```
from sklearn.linear_model import LogisticRegression
```

2.Prepare your data

(Usually using Pandas or NumPy)

3.Split your data

```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X,
y, test_size=0.2)
```

4.Train the model

```
model = LogisticRegression()

model.fit(X_train, y_train)
```

5.Make predictions

```
predictions = model.predict(X_test)
```

6.Evaluate the model

```
from sklearn.metrics import accuracy_score

print(accuracy_score(y_test, predictions))
```

19.What is Git?

Git is a version control system used to track changes in your code or files over time. It helps developers collaborate, manage code history, and revert back to previous versions if needed.

20.GitHub?

- GitHub is a popular web-based platform that hosts Git repositories online.
- It makes it easy to share code, collaborate, review, and manage projects.

21.What is Exception Handling in Java?

exception Handling is a mechanism in Java to handle runtime and compile time exceptions.by using try,catch,final(optional).

Types of Exceptions in Java

Java exceptions are mainly divided into two categories:

1. Checked Exceptions

- These are exceptions that must be either caught or declared in the method using **throws**.
- They are checked by the compiler at compile-time.

- Examples: `IOException`, `SQLException`, `ClassNotFoundException`

2. Unchecked Exceptions (Runtime Exceptions)

- These are exceptions that do not require explicit handling (no need to catch or declare).
- They are not checked by the compiler.
- Typically caused by programming errors.
- Examples: `NullPointerException`, `ArithmeticException`, `ArrayIndexOutOfBoundsException`

22.Error (Not Exception)

- Errors represent serious problems that applications should not try to catch.
- Examples: `OutOfMemoryError`, `StackOverflowError`

23.JDK , JVM ,JRE?

1. JVM (Java Virtual Machine)

JVM is a virtual machine that runs Java bytecode. It's part of the Java Runtime Environment (JRE) and plays a crucial role in Java's "Write Once, Run Anywhere" capability.

Key Responsibilities:

1. Loads `.class` files (compiled Java code)
2. Verifies bytecode for security
3. Executes the code using its execution engine
4. Manages Memory (Heap, Stack, Garbage Collection)

Important JVM Components:

- **Class Loader:** Loads classes at runtime
- **Bytecode Verifier:** Validates bytecode before execution
- **Runtime Data Areas:**
 - **Heap:** Stores objects
 - **Stack:** Stores method call frames
 - **Method Area:** Stores class structure (like methods & fields)
 - **PC Register:** Stores the address of the current instruction
- **Execution Engine:** Executes bytecode instructions
 - **Interpreter:** Interprets bytecode line-by-line

- JIT Compiler: Converts frequently used bytecode into native code for performance
- Garbage Collector: Automatically frees unused memory

2. JRE (Java Runtime Environment)

JRE = JVM + libraries + other files needed to run Java programs.

- Doesn't contain development tools (like compilers).
- Used by end-users to run Java applications.

3. JDK (Java Development Kit)

JDK = JRE + development tools (compiler, debugger, etc.)

- Used by developers to write, compile, and run Java programs.
- Includes:
 - **javac** (Java compiler)
 - **java** (launcher for JVM)
 - Development tools and libraries