



Project Discussion Question Sets

1. Project Overview & Ownership Questions

Question	Model Answer
“Can you explain your project in short?”	“Sure. I developed a <i>Student Attendance Automation System</i> using <i>Python (Flask)</i> and <i>SQLite</i> . My role was backend development — I built QR-based attendance capture and data validation features. It reduced manual entry errors by 95% in testing.”
“Was it an individual or team project?”	“It was a 3-member team project. I handled the backend logic and integration. My teammates worked on UI and testing.”
“Why did you choose this project?”	“We wanted to solve a manual attendance problem common in colleges. It was practical, and it helped us apply Python, DBMS, and OOP concepts in a real use case.”
“What was your exact contribution?”	“I designed the database schema, implemented login validation, and created APIs for attendance entry.”

2. Technical Stack & Language Choice

Question	Model Answer
“Why did you use Python (or Java) for this project?”	“Because it’s simple to write and integrates easily with Flask for web development. It also has good library support for QR code generation and DB handling.”
“Can you explain your backend-frontend flow?”	“The user scans a QR code from the web interface → request goes to Flask → Flask validates data and stores it in SQLite DB → response sent back to frontend as confirmation.”
“What are the libraries you used?”	“Flask for routing, SQLite3 for database, and qrcode library for generating QR codes.”
“If you had to switch to Java, what changes would you make?”	“I’d use Spring Boot for backend APIs and MySQL for persistence. The flow would remain same but the libraries and syntax differ.”



3. Code & Logic-Based Questions

Question	Model Answer
“How do you check login credentials?”	“We query the users table using the entered username and password. If a match is found, we return ‘Login Successful’, else invalid. We use hashing for security.”
“Can you explain any function or class from your code?”	“Sure. I have a function <code>mark_attendance()</code> that takes student ID and timestamp, checks if record already exists, and inserts it if not.”
“What type of loop or conditional statements did you use?”	“Mainly for-loops for iterating data and if-else for validation checks.”
“Did you handle any exceptions?”	“Yes. For DB operations, I used try-except blocks to handle connection and integrity errors.”

4. Database & SQL-Focused Questions

Question	Model Answer
“What tables did you have in your database?”	“Three: students, attendance, and users. attendance has foreign keys referencing students.”
“Explain your primary key and foreign key.”	“ <code>student_id</code> is the primary key in <code>students</code> , and the same is a foreign key in <code>attendance</code> .”
“Write a query to count attendance per student.”	<code>SELECT student_id, COUNT(*) AS total_days FROM attendance GROUP BY student_id;</code>
“What is the use of GROUP BY here?”	“It groups all attendance records by student, so we can count total presence per student.”
“Did you use JOINs?”	“Yes, I joined <code>students</code> and <code>attendance</code> to display names with attendance count.”



5. Challenges Faced & Debugging

Question	Model Answer
“What was the biggest challenge you faced?”	“Initially, attendance records were duplicating. I solved it by applying a UNIQUE constraint on (student_id, date) and adding validation in code.”
“How did you debug that issue?”	“I printed logs at each step and checked the DB query execution in console to trace duplicates.”
“What did you learn from that bug?”	“That validating data at both application and DB level prevents inconsistency.”
“What would you improve if you had more time?”	“I’d add role-based access for admin vs students and implement a dashboard for analytics.”

6. Tooling, Testing, and Deployment

Question	Model Answer
“What IDE did you use?”	“VS Code for coding and DB Browser for SQLite for queries.”
“Did you test your project?”	“Yes, I tested each function manually by running sample inputs and verifying database changes.”
“How did you deploy or run it?”	“Locally on my laptop using Flask’s inbuilt server. I’d host it on Render or PythonAnywhere in the future.”

7. Project Reflection / Learning

Question	Model Answer
“What new skill did you learn from this project?”	“Integrating backend and database. I learned how APIs connect front-end and DB.”
“If you were to rebuild this, what tech would you choose?”	“Probably Node.js with MongoDB for scalability — just to explore a NoSQL option.”
“How did you divide the work in the team?”	“We divided by modules — I handled backend, one teammate handled UI, one testing. We synced code weekly.”



8. Hobby & Project Connection (Bonus HR–Tech Blend)

Question	Model Answer
“How did your hobby help you in this project?”	“I enjoy UI design, so it helped me make the layout user-friendly and simple for others to test.”
“Do you plan to build more projects?”	“Yes, I’m working on a version with login analytics and mobile UI.”