



I'm not robot



Continue

Sql server interview questions pdf download

By ExtremeTech staff on December 20, 2001 at 12:28 pm This site can earn affiliate commissions from links on this site. Terms of use. SQL Server is an online journal for database administrators. Its articles discuss issues including database administration, service pack improvements, .NET services, and security. You will also find an informed editorial and SQL library code extracted from the tutorial. Some of the pieces are available for free online, others require a paid subscription. As an employer and examiner, it can be difficult to sort out good candidates than less qualified ones. When conducting interviews, make sure you ask the right questions, so that the candidate you choose is not only professional and career-oriented, but also has goals and healthy interests outside the office. One of the first questions you need to ask as an employer should lead you to learn more about who this person is. Ask the candidate to tell you about himself, his educational choices, his or her background and legacy. Every person has a different story, so ask to hear his. Ask the candidate why she chose this specific career or industry. For example, if a candidate is speaking for the position of legal secretary, inquiring about her interest in the right and interest in the post. From her answers, you can easily determine whether a candidate is enforcing the law because it is a passion or simply interviewing to get a job for money. Ask about the candidate's life goals. Objectives may include work or career objectives, as well as personal objectives. If the candidate's goal is to work effectively as part of a legal team, you may have a good candidate. If, on the other hand, a candidate's goals include working from home or staying at home, the candidate may not be the one you're looking for. While some employers want their employees to have healthy lifestyles and hobbies outside of work, others don't care as long as the work is done. Candidates enjoy talking about themselves, so ask about their hobbies and interests outside of work. Use the answers to get to know the candidate better. Ask a question concerning candidate selection and education level. For example, if a candidate is interviewed for a position as a secretary but has a degree in English literature, ask her how her learned education and skills will help her perform in this position. Two issues common during interviews relate to the strengths and weaknesses of candidates. While a candidate can easily identify their strengths, weaknesses can be more challenging, as the candidate does not want the weaknesses to take over and become the reason why they do not receive a job offer. Two more questions you should ask a candidate who deals with previous experiences in the business. Ask the candidate about responsibilities or tasks in previous jobs. Then ask her about personal enjoyment Job. While the candidate may have been good at her job, her answers will show if she enjoyed the job. This can be harmful, especially if the candidate works directly with clients. The last question you should ask the candidate is why you should hire him. It's an interview point of sale, because the candidate has to explain why he thinks he's qualified for the job. I like to collaborate with my readers and find out what their concerns are when it comes to a technical interview. In this article, I will go through the reader question and 3 real SQL questions that were asked during technical reviews from real companies. Readers' most common question: How should I prepare for SQL interviews? As for questions about the practice, you can check out my second article on some of my favorite exercise platforms. I would do as many medium and difficult problems as you have available. I would seriously do as many of them as possible to make writing queries second nature. I want the technical part to be an easier part of the interview process, so I can focus and spend my energy communicating and crafting my approach and solutions with the examiner. In my experience on technical interviews, the biggest focus they test on is understanding what code does, how you think about different scenarios/fringe cases and the implications of results. The SQL part is pretty simple compared to the long, complex queries you'd probably write at work. You will either get 1 or 2 tables and are asked to create an SQL query that requires association or self-association, or you will receive an SQL code and are prompted to correct it. As long as you understand JOINS and slightly advanced functions like COALESCE, you'll be fine. What makes it difficult to interview is to link the way the code is written with the result you'll get if you're running the code. Here are 3 real SQL questions asked during technical interviews: You got 1 table containing requests for user friend, acceptance and their dates all on 1 table. How do I write an inquiry so you get a % friend acceptance over time? This query is easy to create (tip: use self-join), but you will be further tested on compromises on how you write a query. How do you deal with accepting a friend that happened a few days later? What day do you count acceptance? Do you count on the day your friend sent the request, or do you count it on the day the request is accepted? Why? There is no right or wrong answer here, but you need to identify that there are two different ways to write queries, and then talk about compromises between the two choices. 2. Example from a real interview: If you started an AB experiment and saw a 2x increase in friend acceptance, with p < 0.05, due to a new feature running in pilot regions, do you implement it in production and launch it to everyone? People would say yes, but in an interview, the most obvious answer is probably not correct. The correct one in this case is – it depends. Then you should talk about why it depends and what additional information you need to make a decision. These are all examples of the types of considerations that you will mention as expected to formulate a complete answer. 3. Example from a real interview: You have a master table that contains a user ID and their most recent sign-in date, and you have another table that contains all the users who signed up for the day (there may be multiple logins from the same user that day). Write a query that will update the master table with the user ID and their latest sign-in date. Simple, but you have to go through an exercise in understanding all the different scenarios. In this case, you have a scenario of a new user who just logged in that day, as well as a scenario where you have multiple logins from the same user for the day. How do you deal with these cases? More tips I'm sure you think these questions are simple as you read this article. But what makes it difficult is that you don't expect these questions during the interview, but are asked to cut them down and solve them on the fly in front of the interviewer. You are asked to code something and think about all the different scenarios and implications of the results. It can be a stressful situation and most people freeze to death. This is why your technical skills need a different nature. Your SQL skills must be as good as your English writing skills. My main advice to you in preparing for an SQL interview is to understand why and how to write code to solve a particular problem. Be prepared to communicate why you write specific lines of code, what logic you add to solve specific fringe cases and scenarios, and what output it will give. Your explanation is just as important (if not more important) than the code itself. Also, be sure to talk to the examiner as you build a solution and keep them up to date on your thought process as much as possible. SQL interviews are designed to test your technical skills. Zero in on the skills that will be included in your interview and be sure to do as many practice questions as possible. Join Hacker Noon Create your free account to unlock your custom reading experience. 365 Data Science is an online platform for an educational career. SQL is one of the most popular coding languages today and its domain is relational database management systems. And with the extremely rapid growth of data in the world today, it is no secret that companies from all over the world want to hire the best experts in this field. So imagine being interviewed for your ideal job and advanced professionals sitting in front of you, interested in how Do. Such a meeting will be crucial for both sides. However, there's no reason to go crazy! To reduce stress, here are our top tips for answering 10 frequent encounters with SQL interview questions. What is SQL? SQL is an acronym for structured query language. It is a programming language specifically designed to work with databases. Of course, some may argue and say that this is not exactly a programming language because it was not created with the idea of using procedural language features such as conditional statements or for loops. These people will insist on calling SQL an encoding language because it is only about executing query commands, creating, inserting, updating, and deleting data into a database. However, it is more important to know which domain is SQL. But don't rush to tell the interrogators, because that could be your next question! And in our exemplary excerpt with SQL interview questions, that's exactly the case! What is a database? What is DBMS? The database, which includes an electronic database, the data is stored on the computer and organized in a way that facilitates access and manipulation. A software tool that allows a user to interact with data stored in a database is called a database management system – DBMS. You can end these two questions by saying that there are two types of database management systems – relational and non-relational. SQL is a language, designed only to work with the DBMSs. It is normal for examiners to start with two fundamental questions with which you feel comfortable. So you can relax and prepare for the continuation of some of the more challenging ones. You can read more about SQL language and database management systems in our Guide Why You Should Learn SQL. What is the difference between DDL, DML, DCL and TCL? First of all, what do these acronyms mean? L means Language in all of them. And it has to help you remember that these are the four categories into which SQL statements are separated. DDL stands for data definition language and includes commands that allow you to create, DROP, ALTER, and TRUNCATE data structures. DML, instead, includes commands to manipulate information. This actually means Data Manipulation Language and refers to the ability to select, insert, UPDATE, and DELETE data. If you use SQL in the sphere of data science or business intelligence, it's this part of the language you'll use most at work. DCL, the language of a data control, consists of commands commonly used by database administrators. This category allows the developer to approve and revoke rights about how much control you can have over the information in the database. Similarly, TCL, which is the language of transaction control, also contains commands applied by database administrators. They ensure that transactions that occur within the database will occur in a way that minimises the risk of data loss. What is the purpose of using a foreign key restriction? Once you've gone through the underlying questions of the SQL interview, you're likely to be asked something more specific. Therefore, your next task will not be to explain what SQL restrictions and keys mean in general, although you need to be very familiar with the concept. You'd rather get a chance to demonstrate your ability to work out a certain type of SQL limit - foreign key limit. The limitation of foreign key factors includes a set of rules or restrictions that will ensure that values in tables for children and parents are matched. Technically, this means that the external key limitation will maintain the reference integrity within the database. If you want to dig deeper into this topic, here we explain in more detail the primary, foreign and unique keys Define and provide an example of using an inner compound. It's not all about theory. Using a convenient approach to handling realistic tasks is often more important. That's why you'll also have to deal with the practical questions of SQL interviews. Obviously, you need to be aware that associations are one of the most commonly used tools in SQL, regardless of your work role. Especially if you work in the sphere of business intelligence, your work will focus on understanding SQL joins in depth. So SQL Association is a tool that allows you to build a relationship between objects in your database. Consequently, the join displays a result set that contains fields derived from two or more tables. For example, suppose you have personal user pot and sales-related fields that the customer has made in one table, and in another, you have information about the personal user message and their private information, such as first and last name and e-mail address. Therefore, the inner join allows you to obtain an output that contains information from both tables only for customer ID documents that are in two tables that match. Provided that you place the user ID field on the corresponding column, of course. Use the previous example to explain how to use the left port. SQL joins such an important topic that this could lead to a follow-up question. It's good to give a sharp answer in this case. You can say: Unlike internal association, the left port will ensure that we extract information from both tables for all customer IDs we see in the left table. User IDs that match between two tables may also contain data from the right table, while personal ones found only in the left table will display null values in the right table. To expand your knowledge on this topic, see this article This is the difference between MySQL and PostgreSQL? How between PL/SQL and This is tricky. Basically, the reason to encounter a question of SQL interviews like this is that the examiner wants to understand to what extent you are familiar with the fact that SQL has several versions, each of which carries specific characteristics. You could say that MySQL and PostgreSQL are only two versions of the structured query language. Since you've just been asked about joins, you can mention that PostgreSQL supports external associations, while MySQL doesn't - you'll need to use UNION or UNION ALL to mimic an external association with MySQL. And so, perhaps you could impress interviewers with additional knowledge in this subject. PL/SQL not a version of SQL, though, and that's a tricky part of the question. PL/SQL is a complete procedural programming language and its scope of application is different. This is not strictly related to relational databases. What is this inquiry about? CHOOSE emp_no, AVG(salary) FROM THE PAY GROUP EMP_NO AVG(salary) > 120000 RED BY emp_no; The version of SQL in which this query is written is MySQL, but you won't really have to mention it. Even if you don't recognize the version, then common sense, the keywords you see, and the names of the fields should convince you that this inquiry to extract the average salary that employees receive is only when the salary value is more than \$120,000. And do not be surprised if after you give your answer, the examiner asks: And the database will not throw an error?. Read the query carefully before responding. It is much better to double-check and make sure that everything is right in this situation. More about the differences between WHERE or HAVING can be found in this guide This next two tables are part of the database you are working with. Write an inquiry that will show the salaries received by the last contract of a particular employee as a result. Limit the number of records received to 1,000. SELECT s1.emp_no, s1.from_date, s1.salary FROM s1 WHERE s.from_date = (SELECT MAX(s2.from_date) FROM salary s2 WHERE s2.emp_no = s1.emp_no GROUP BY emp_no) LIMIT 1000; In fact, this is the question about using SQL sub queries - a subset of SELECT statements whose output sets the conditions to which the data for the main query will be filtered. However, you may not be given this hint, so it is up to you to remember that in such a situation the subspecies is exactly what you need. And this is a pretty complex inquiry, to be honest. However, by asking you to create one, questionnaires can check your command over SQL syntax, as well as how you approach troubleshooting. So if you fail to come up with the right answer, you will probably get time to think and you can definitely get their attention by trying to solve the problem. It's interesting to know more using SQL subqueries? Then go to this tutorial. What is SQL view? To conclude the interview, your potential future employers may prefer to give a toned interview question with QL. So we could ask you something that's unrelated and go back to asking the general question. A view is a virtual table whose contents are obtained from an existing table or table, called a master table. Retrieving occurs through an SQL statement, embedded in the view. So, you can think of the object of the view as a view of the base table. The view itself does not contain actual data; data is stored electronically in the base table. The view simply displays the information contained in the base table. If you are interested in learning more about this tool, check out our Introduction to SQL Views. General Tips guide. While you may have answers to all the SQL interview questions you've been asked, there are many other components that will determine whether you get the job. The company you are applying for can have very strict requirements regarding work ethic, employee background and so on. And it all counts, believe me. So if you want to be fully prepared to make a great first impression, check out the most comprehensive article: Starting a Career in Data Science: The Ultimate Guide. However, nothing else will matter if you're not a good professional, will it? That's why you need to stay focused on SQL and learn as much as you can about it. If that's what you're eager to do next, check out the tutorials we've provided above, or feel free to find more SQL content on our blog. Good luck! References //365datascience.com/sql-relational-databases/ <8>/techniques-for-processing-traditional-and-big-data/ published on Hacker Noon Create your free account to unlock a custom reading experience. Experience.