**DAILY ASSESSMENT FORMAT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | **11-July-2020** | **Name:** | **Raziya Banu** |
| **Course:** | **Coursera** | **USN:** | **4AL16EC058** |
| **Topic:** | **Subqueries** | **Semester & Section:** | **8th sem & ‘B’ section** |
| **Github Repository:** |  |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
| **Image of session** |
| **Report –** In my first session today I have studied about -Subqueries Things are starting to really heat up for us now with SQL because we're finally getting to a point where we're going to be able to use multiple tables together. Up until this point we've just been working with a single table, and we've learned how to do some cool things. We can get lots of data returned. We can perform some calculations on it, but we're still really limited in terms of we can do because we're only using one table. And where we get a lot of value in relational databases is when we combine different facts, dimensions and sources together to obtain new meaning. One of the easiest ways to do that is with something called subqueries, which is what we're going to focus on in this lesson. After this video, you should be able to define subqueries, discuss advantages and disadvantages to using subqueries.  Explain how using subqueries can help us merge data from two or more tables together and write efficient subqueries to make your data analysis smoother. So to begin let's define subqueries. In essence, subqueries are queries embedded into other queries. It's literally just a query inside another query. So far we've only worked with a single select statement and a single table, but when you're working with a relational database, since data is stored in multiple tables, you may need to merge these together or  get information from multiple sources. Data scientists often use subqueries to select specific records or  columns and then use that criteria as a filtering criteria for the next thing they want to select. Not only are subqueries helpful when it comes to getting information from multiple tables, but they're often used for adding additional criteria like filtering criteria that's not in your current table from another table into your query.  Let's look at another example of this. Let's say we want to know the region each customer is from,  who has had an order with freight over 100? Maybe we're wanting to look at a shipping cost and say,  who's really had heavy freight and where are these coming from? If we did this with the normal method we have right now, since this information is in two different tables, the customer table and the orders table, we would have to first retrieve all of the customer IDs for orders with freight over 100. But then we wouldn't have what region the customers from because that's in a different table. The orders tables in this case.  Then I would have to go and retrieve the customer information to get the regions from that separate table and then I would need to combine that so now I can select the IDs for the ones that had freight over 100.  Clearly this is quite a few steps and we always want to limit these and make things as simple as possible in SQL. In this example again you can see we pulled our customer IDs WHERE Freight > 100.  To do another query, to get my CompanyName from my Customers region And then now, combine the list of those customer IDs into my other query which had my customer information. This is also taxing right now because I just had a few customer IDs. So I could just say where a customer ID is this. Because I've already prequalified this as being over a hundred in freight. If that was a large list, this would be very taxing, when either trying to find a way to manually copy and paste in, or also computationally for it go through each one and match it. So to avoid this missing SQL, we write what are called subqueries. We can take the three queries we had to write to do the job. And instead combine all three queries into one larger query in order to make things more efficient. What I usually do when writing a subquery is start with the innermost query.  How we're using this query is to filter for customers, that we're getting back. My filtering criteria in this example is to find records where the freight is over 100. I'm first going to take that first query I wrote,  which is to select the customer IDs from orders where freight is over 100. And that's going to be my main base. I'm going to filter down to that and then I'm going to go and expand and add to it. This is your innermost query. From there, I'm going to write the query to select my customer id's. My company name, my region, to get my region information. I'm going to get that from inside the customer table. Where now I can connect the subquery and my customer information query together is using the WHERE clause.  When I say is where the customer ID is in the SELECT statement, remember we're using this kind as a filtering criteria. It's saying only to return the region CompanyName and CustomerID for those CustomerID's that are in this other SELECT statement. A good way to think about this is that the query statement is always going to go perform the innermost SELECT portion first. It's going to say okay, I've got the customer IDs where the freight's over 100. And then it's going to sit there with them. Then it's going to go and go through to bring in the CustomerID, CompanyName and Region. But it's going tomatch to the customer IDs up against the list it pulled earlier to see if the customer IDs are actually in there.  Then it's only going to return those customer IDs that were actually in that list of the select statement.  You can see this helps reduce the number of queries we have to write, and it also helps keep everything concise in one query. Two things to remember about this in the innermost query is it's always what is performed first. If I'm looking at somebody's query and trying to troubleshoot it I always start with the innermost query to see what's going on and then slowly from there build and work backwards. In this example, remember the database is really performing two operations.  One, it's getting our customer IDs for the criteria freight over 100. Then it's going and pulling the other customer information and matching it up against the customer IDs we've already pre-selected. Okay, this is a good place to stop for now. You know all of the basics with subqueries. What they are? When they're advantageous? How they help us when combining data from multiple tables and how to write them effectively. Keep all of this in mind because we're going to be using subqueries a whole lot more from here on out. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Date:** | **11-July-2020** | **Name:** | **Raziya Banu** | |
| **Course:** | **Cisco** | **USN:** | **4AL16EC058** | |
| **Topic:** | [**Educational and Business Opportunities**](https://static-course-assets.s3.amazonaws.com/I2IoT20/en/index.html#6) | **Semester & Section:** | **8th sem & ‘B’ section** | |
| **AFTERNOON SESSION DETAILS** | | | |
| **Image of session** | | | |
| **Additional Learning Resources**  Certifications can show an employer that an individual has the appropriate skills for a job. Community college or university degrees can show that a person has gained a broad understanding in a field of study. This broad understanding creates a solid foundation for emerging career opportunities in the IoT. A combination of industry certifications and university degrees provides a student with the best background, experience, and education to pursue a career with greater opportunities and higher salary.  When looking for a degree to pursue at a community college or university that will pertain to the skillsets needed for a career in the IoT, watch for some of the following degrees:   * Business Intelligence * Computer Information Systems * Computer Programming * Computer Science * Database Administration * Electromechanical Automation * Electronics Engineering * Linux Networking * Machining * Network Administration * System Analysis * Web Server Administrator   This is not an exhaustive list. Even traditional degree programs such as supply chain management, business, and project management are helpful for careers in IoT. Computer-Aided Design (CAD), drafting, math, and physics are applicable and show a diverse education, which is perfect for an IoT career. **Lab – IoT Related Job and Learning Opportunities** As an individual or group, you will search for and document IoT educational and job opportunities that are available to you. You will then share these results with your colleagues.  This chapter began by discussing the challenges and opportunities in the digital world. Digitization continues to provide new opportunities for professionals who are trained to develop and support the technology that is used to deliver the IoT.  In IT, opportunities may be specific to fog computing, developing new processes, or a specialization in a discipline that has not yet been realized. These jobs reflect skills spanning multiple disciplines that include computer science, computer engineering (a blend of computer science and electrical engineering), and software engineering. There are broad categories that summarize the job opportunities that exist in the evolving digitized world:  • **Enablers** – These jobs develop and implement the underlying technology.  • **Engagers** –These jobs design, create, integrate, and deliver IoT services to customers.  • **Enhancers** – These jobs devise their own value-added services, on top of the services provided by Engagers, which are unique to the Internet of Things.  An entrepreneurial workforce is needed that specializes in both information science and software or computer engineering. The Cisco Networking Academy Program has trained more than five million students to date. Many graduates have gone on to successful IT careers in a variety of industries, while others have harnessed the entrepreneurial spirit and knowledge they acquired to start their own businesses and create new jobs. Go to [www.netacad.com](http://www.netacad.com/) to see the variety of courses offered.  There are two basic types of certification available: vendor-specific and vendor-neutral. Vendor-specific certifications are tailored to technologies offered by a company to prove that an individual is qualified to deploy and manage that technology. Vendor-neutral certifications are offered by many different organizations. Certifications can show an employer that an individual has the appropriate skills for a job. Community college or university degrees can show that a person has gained a broad understanding in a field of study. This broad understanding creates a solid foundation for emerging career opportunities in the IoT. | | | |