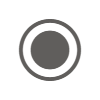
**Maneesh Gorantala \_ L1 Discussion \_ Data Engineer \_Google US Project-20250321\_093218-Meeting Recording**

March 21, 2025, 1:32PM

38m 26s

 **Raheem Khan** started transcription

 **maneesh** 0:06  
Yeah.  
Well, I have 10 years of experience in IT industry as a data engineer, I participated in all phases of data warehousing projects in requirements gathering like preparing technical specification, documents, architecture.  
Like then designing and then like migrating code.  
So like I I worked on like with respect to big data like data modeling Snowflake, Azure, AWS and GCP I can say like where I've been involved in terms of building endtoend like pipelines working with like where where we need to collect the data from multiple sources.  
Systems like then load the data into like a stagging table.  
Table like Mark Table and then find it like find it to the like target tables by applying some transformation and also mappings. Apart from this like apart from this sort of experience like with respect to working on with like migrating the like data from on premise systems to.  
The cloud I I can say in some projects like I migrated data.  
From Oracle database like Microsoft SQL Server into AWS, in other projects I've also migrated into like Azure platforms like and. Also I can select apart from this app experience with respect to working on some part of basic machine learning algorithms like in terms of working with respect to.  
Regression, classification. Doing statistical analysis like.  
Hypothesis testing and also like.  
Some some part of text mining I like I can say like coming to my professional experience. I have been associated with Kobe marketing as a senior data engineer. So like predominantly here I have been involved with respect to working with GCP where we need to create an end.  
To end data integration like data integration like pipeline using data proc big query.  
Like data flow pops up and also.  
Other services I can say.  
And apart from this I've used Apache Airflow to orchestrate ETL workflows on like DCP data PROPE also maintain the like data pro cluster where we need to use like spark for real time data processing and PY spark for. I mean like advanced data transformations like transformation with respect.  
To business requirements.  
Aggregations aggregation joins like sorting transformations or any other like transformation based on the business requirements. I can say like. Apart from that I also optimize some of ETL workflows using Spark with respect to big query SQL where we need to like automate the data like across multiple cloud.  
Like with comments, I can say.

 **Ramakrishna Medisetti** 3:25  
OK.  
That's great, Manish, right?  
You have over 10 years and.

 **maneesh** 3:29  
Yeah.

 **Ramakrishna Medisetti** 3:29  
Computer relevant into data engineering, right?

 **maneesh** 3:32  
Yes, yes.

 **Ramakrishna Medisetti** 3:33  
You worked on GCP as you also AWS.

 **maneesh** 3:38  
Sorry.

 **Ramakrishna Medisetti** 3:39  
So you worked on.  
Gcp.

 **maneesh** 3:43  
Yes, yes, yes.

 **Ramakrishna Medisetti** 3:44  
Azure also you work now.  
Have you worked on OK?

 **maneesh** 3:46  
Yes.

 **Ramakrishna Medisetti** 3:47  
You work on multi cloud.

 **maneesh** 3:48  
Yes.

 **Ramakrishna Medisetti** 3:49  
That's really great, right?

 **maneesh** 3:51  
Yeah, with like I with respect to GCP, all the like all the three I worked in different different projects. But predominantly I work with respect to GCP.

 **Ramakrishna Medisetti** 4:02  
OK, OK.  
That's great, yeah.  
Yeah, we'll start with the the technical evaluation, right.  
Yeah. How do you no design and ETL pipeline that processes large amounts of data in batches?

 **maneesh** 4:22  
Large amount of yeah, so.  
I like. Yeah. I mean to say.  
Like like you mean to say? Is it like on like on premise systems or like GCP talking about?

 **Ramakrishna Medisetti** 4:38  
Cloud cloud. Cloud cloud, yeah.

 **maneesh** 4:41  
Cloud. OK.  
Yeah. So like so. Like, let let me start like like with respect to like visit part like because like I worked with respect to GCP.  
So in order to design an ETL pipeline so like first we are like first thing is we should data mine the data systems like.  
In our projects like we had like databases like APIs and also.  
Like some part of like clouds storage, I can say and after that like we were using the like data extraction where we used like where we used to use this cloud functions to extract the data from various sources including like where we need to like consider the data.  
Flow for like a unified data stream and also batch processing solutions and also like like we need to use like this data flow for applying some.  
Transformation like where you can write your like like like a transformations in Python.  
Alternatively, you can also like it automatically in other project I have used like cloud data processing solutions and also like like like we need to use the like dataflow for applying some transformation.  
Like like we have loaded the transform like data into like destination like using a big query I can say.  
And apart from that, like in order to create a workflow orchestration like where we were using a cloud composer to schedule and monitor the ETL process.  
Where do we need to set up that like trigger mechanism to automatically run the ETL jobs based on the events or like or like events or some particular schedules like because in every projects we have like different schedules like in our project like in in our current project.  
We run the schedule every day like 12 AMS like EST. Where we where we need to run this ETL job and like after that.  
In order to monitor and logging we use like cloud monitoring and also like cloud logging to monitor our ETL jobs where we can.  
Identify issues during extraction like transformation and loading process. I can say and also apart from this like we also like do testing and validation like also like optimization like.  
Performance for like batch particularly.  
About batch loading all like and also we have done it in like real time streaming. Also using GCP with respect to like batch loading where we have done like once a transformation is done like we are going to load the data.  
So generally we use like the big query to lower the jobs to to load the data directly from like GCS and like the data transform it into.  
Big query and for scheduling our orchestration.  
Like I mentioned, we used like we use cloud composer in some project like so we used in.  
Airflow also in some projects I can say like I can give you one small example like with respect to batch ETL workflow where we have like what we have done is.

 **Ramakrishna Medisetti** 8:04  
OK.  
Oh yeah, one minute, Manish. OK. Like, let's say you are handling with the Hadoop data, right?  
Hadoop data you know migration right from on premises to the GCP.

 **maneesh** 8:12  
Yeah.

 **Ramakrishna Medisetti** 8:14  
Like what are the GCP services you use it to do?

 **maneesh** 8:15  
Yes.

 **Ramakrishna Medisetti** 8:21  
This data, you know initial and transmissions and moving it to the consumption zone.  
Like what is the?  
Tickle pot. You know services. Can you give me some?

 **maneesh** 8:29  
Yeah.  
Yeah.

 **Ramakrishna Medisetti** 8:35  
The scenario based explanation on this.

 **maneesh** 8:38  
Yeah, got it. Got it.  
Like sure.  
So in order to migrate the data from like like from Hadoop or like DCP Services, we can use Google cloud storage like which is going to act as a cloud storage repository for our data before our like after processing then we can also use big query like so.  
That it's going to use us like fully managed data warehouse for.  
Analytical like queries on reporting.  
And then we can also use Spark and Hadoop service which which like which is like Google Cloud data PROPE which is used for running out of jobs directly near GCP and then we can use transferring services for an on premise data like which is.

 **Ramakrishna Medisetti** 9:22  
One minute.  
One minute Manish.  
Sorry to interrupt you like what are the file formats right?

 **maneesh** 9:25  
Yes, yes.

 **Ramakrishna Medisetti** 9:27  
So let's say from source you handle like so in R2 you don't just the data, right?  
For data engine like in the source and target like what is the you know different file formats or file storage systems you use.

 **maneesh** 9:43  
Yeah.

 **Ramakrishna Medisetti** 9:43  
We use CSV or any other.  
So yeah.

 **maneesh** 9:48  
Yeah, like.  
Did did you come to the question? Yes.

 **Ramakrishna Medisetti** 9:52  
No. Yeah, I completely like what is the file format review is like storage storage. This kind of, yeah.

 **maneesh** 9:54  
Yes, yes. Yeah. OK.  
So throwing some projects I've used with respect to like like a CSV file format.  
Also, in some projects I've used parquet file format in. In one project I used AVERRO file format as well like in in like in. In one project I have used CSV like in so I used like.  
Like packet packet file format in another projects like when it comes to the target file system like we used file system like majority of them.

 **Ramakrishna Medisetti** 10:29  
OK, OK.  
OK.  
That's. And what about streaming? Like if you want to process the real time streaming data.  
Can you give me an example like how the end to end flow right from the on premise system right?  
To the GCP environment, how do you handle data ingestion process?

 **maneesh** 10:53  
So like.

 **Ramakrishna Medisetti** 10:54  
No text stack use and yeah.

 **maneesh** 10:58  
Yeah.  
Got got it. Got it.  
So in order to handling the streaming like dataflow like first thing, we need to determine the data source like the for example logs or like like or like log services like log devices data or like application data. We need to understand that then we need to use pop.  
Up for real time ingestion, so like.  
In in your pops up.  
It is a messaging service for like real time event. It can be used for ingestion and delivery where we can create a topic like in in pops up like where the data will be published and then we can use like like published message.  
Where you can use a like a client library or a connector like for your data source like published like streaming data to the Pops up topics like for example.  
I'm just telling you like.  
Then we can like then we can stream the like like processing data with the data flow by like by like where you can follow this process like data flow process to analyze the data in real time.  
Then like then we need to create a data like data flow pipeline where we can build a like stream processing pipeline.  
Let that needs that. That like that reads the data from like from pops up topics and transforming the data like where like like is we are going to apply any transformations like filtering or enrichment like to the standard streaming data as well.  
And after this like we need to.  
Write it as sync after processing like direct the output to the destination like such as big query or cloud storage like whatever you're going to use like that. Need that needs to be written like written to a sync so like then after this we need to store the.  
Processing data like for suppose if you decide to load the data like load the transformation data into big query for analytics.  
We need to set up a big query table like to receive the streaming data.  
So like we can use appropriate schema based on the structure of like incoming data.  
So alternatively, if you want to process the data.  
Like that needs to be in in like in the RK format. So you can also go for parquet like or like Jason file format or even file format like all they are they can also be considered so like this is in like Endtoend pipeline. Also if you want.  
To monitor an error handling.  
We can use Google cloud monitoring and logging services to track the health of our like streaming pipelines I can.  
I mean, I can say this that I have done in my project and this is like, AW, whatever the services.

 **Ramakrishna Medisetti** 13:56  
OK.  
That's fine. Yeah. Great, great.  
And when it comes to the orchestration, right?  
How do you use?  
Cloud composer or any other A+ services right for orchestrating the ETL pipelines and how do you handle the dependencies dependencies right between these multiple tasks, right which you.

 **maneesh** 14:18  
Yeah.

 **Ramakrishna Medisetti** 14:22  
Create like doing the pipeline.

 **maneesh** 14:25  
Yes. Yeah, so.

 **Ramakrishna Medisetti** 14:27  
2nd.

 **maneesh** 14:29  
Yes, yes, yeah.  
So like between multiple tasks. So like generally like when we are defining a orchestration job, right?  
I mean to say like I need to run my ETL jobs at a particular point of time.  
So like we can go to like using our cloud like I mean using our cloud composer where we can like where we can go to the composer section where we can click an envelopment like where we can specify the name.  
Location like machine type and and then we can also create a DAG that is going to represent like workflow in our like in your cloud like. I mean in your cloud compose and write a like.  
And and like write a like like for example like my DAG dot py file in the Dags folder. You can create it as a like composer bucket then we can.  
Said like set the dependencies like like in the above example.  
Like for example, I'm like I'm I'm telling you like like consider that you're using.  
This midway shift operators like double like, double greater than symbol or double like less than symbol so that that is going to start like start task must complete before processing task starts and then processing task must complete before like before the in task like like suppose if you.  
Are having a task one and task two like we can specify that.  
Like start task processing task and also you can define more complex dependencies.  
Such as like branching and parallel tasks like paralleling tasks for example, if you want to run 2 task in parallel after start task, you can modify that task like that to include that in your like Python code.  
I can say that and and also we can use some sensors for like dependency management, but it's not.  
In all ways, like in in all project.  
Like not in all projects.  
Like for for example, if you want to like wait for a specific file to be available in the cloud storage like or for a particular task to succeed.  
In those cases, you can use a sensor for like dependency management like I have done it in one project actually like as a part of my PS organization.  
Oh, I can say like and also we can trigger the workflows and handling failure choosing like X comp like cross communication to share the data between tasks if necessary.  
Like task and push and pull the data and like and from X comps like where we can configure some e-mail alerts notifications for failures or like retrace near DAG by modifying the default apps.

 **Ramakrishna Medisetti** 17:24  
OK.  
That's great, yeah.  
And how do you integrate Apache Kafka into ETL pipeline for real time data engine?

 **maneesh** 17:32  
Yeah. So like for, yeah.  
So for like you need to integrate like.  
Like Kafka for like realtime ingestion.  
So like we can do it like for example consider that you are getting a data from like Kafka system where you have awa S or suppose if you are working with respect to Kafka.  
Clusters I can use a Kafka producer so that we can write or configure our producer to send the data to the Kafka topics.  
So there can be like so there could be an application that generates the data in real time like for example log data like sensor data or web log or log generators then so then we need to use.  
Their Google cloud pops up if you are working with respect to awls or other services.  
Like you can use it like I mean so we can have a pops up which is going to act as a bridge between your Kafka and GCP services.  
And then we can integrate the Kafka Connect framework to send the data.  
From Kafka to Google Cloud pops up then like the Kafka will allow integrating Kafka with like external systems like Pubsub easily.  
So that we can use pops up connector. There is something called a sync like connector to publish a cafka message to the cloud pops up.  
So this is how we can do it.  
Like also we can assume like consume the like message from Pops up using dataflow.  
For that, again, we need to use a good cloud data flow to process messages from Pops up for like further analysis and also store like Store part I can see.

 **Ramakrishna Medisetti** 19:30  
OK, OK.  
Like, how do you handle schema evaluation in data pipeline?  
Especially like when dealing with the changing the data sources like adding new fields to a database.

 **maneesh** 19:46  
Yeah. So like when handling the.  
Like so, like in the in data pipeline with respect to GCP. If you want to handle the schema evaluation, we can use a schema like aware like data format for example like ARROW file format or like file format or like packet file format or Jason so like this.  
We can.  
Leverage big query for schema management like so because if you like, if you want to add a column right like you can easily add new column to your big query tables like without affecting existing data. Or suppose like if you want to add a new columns like that.  
Them as a like nullable like so that you can ensure that existing row do not conflict with any new schema like. So you can use some like SQL DDL like commands to modify your table.  
So like for that we can maintain a schema registry like so implementing a schema registry can help us to track the version of your like data schema over time and like.  
And the changes more effectively and also.  
And also like schema register will maintain the repository like for schemas with versioning capabilities. I can I can say.  
And apart from that, we can also use conflict schema registry like if you're using some like third party tools or solutions.  
Suppose if you're using Kafka like or Google Cloud store plug. For smaller setups you can like use this also like we can we can have an option called as backward and forward like compatibility suppose.  
If you have a new schema like, I mean if you have a new schema version.  
That should be able to read your data from like from produce with others like. I mean older schemas. You need to add a new field to ensure it is nullable.  
Like or like provides a default value or if you can use a forward compatibility. So older versions of the data will still work with the new schema like like for instance when you remove an optional field. If you'll handle it in your processing logic.

 **Ramakrishna Medisetti** 22:04  
OK.

 **maneesh** 22:10  
Like in like like that you can handle your schema registry.

 **Ramakrishna Medisetti** 22:14  
That's fine, that's fine.  
Yeah, like, how do you design your data pipeline for like scalability?  
Like for handling the millions of records per day with minimal performance.

 **maneesh** 22:29  
With minimal performance.

 **Ramakrishna Medisetti** 22:31  
Degradation like yeah.

 **maneesh** 22:33  
Yeah, yeah.

 **Ramakrishna Medisetti** 22:34  
Yeah.

 **maneesh** 22:35  
Legal. Yeah, so, right.

 **Ramakrishna Medisetti** 22:36  
Yeah.

 **maneesh** 22:37  
Yeah. Like caught it.  
I mean like so in order to design A pipeline with scalability, we can choose a like a particular right services like like you can use data engine like Google Cloud Pub for realtime messaging and data ingestion and for like data processing you can use.  
Google cloud dataflow.  
Because it's going to process the data in like a serverless manner like and for data storage you can use big query or like analytical workloads or like Google Cloud storage for like both unstructured data as well like for structured and unstructured data as well. So like this is.  
Going to scale effectively and also we can implement like parallel processing like wherever it's possible. So like that you can use data flow like you can leverage like parallel.  
And through the use of your like transform such as like a call peak or like caller like peak caller like to distribute the P collection to like distribute the workloads across like multiple nodes and also for batch processing like consider breaking down the data into smaller chunks that.  
Can be processed independently and we can also use the.  
Like data partitioning.  
Like because in big query to optimize.  
Like the query performance partitioning of your tables based on the data or like other criterias you can use it and also we can use auto like auto loggers, auto scaling features like like data flow will automatically adjust the like number of worker nodes based on the like work.  
Which helps maintain.  
Which helps maintain and like performance during spikes.  
You can also use Google Kubernetes engine for using like horizontal port or like autos killing to adjust the resources based on the workloads and and also if you haven't like opportunity you can also implement like catching resources like based on the workloads and like and also like you.  
Can implement the like catching resources like strategies where you can.  
Use Google cloud.  
Memories or to like to readies or catching latest in your application to reduce the load and speed up the access.  
So these are some things we can do and also we can implement asynchronous processing to decouple the components of your pipeline that is going to ensure the failure of like one component that not cascade through pipeline to the pipeline so.  
So like these are the some things we can do.

 **Ramakrishna Medisetti** 25:29  
OK.  
That's great.  
OK.  
Let's say you have a data pipeline that process daily.

 **maneesh** 25:34  
OK.

 **Ramakrishna Medisetti** 25:36  
No transactional data and loads into the relational database.  
The transaction value means increasing rapidly and database is starting to, you know, experience in the performance degradation issues.  
What changes do you make to the pipeline for improving the scalability issues?

 **maneesh** 25:56  
Yeah. So in this case.  
So in this case, like you might, I mean, you might say just like the data volume increases like how we can optimize it, right?

 **Ramakrishna Medisetti** 26:08  
Right, right. Yes.

 **maneesh** 26:10  
Yeah. For performance. Yeah. OK.  
So in this case.  
Like, I mean like we can.  
We can use batch processing like by like. We can batch the like transaction like instead of processing each transaction like like individually we can aggregate them into batches like. This is going to reduce the overhead like of processing calls and allows for more efficient data handling.  
And like if it's real time, I mean, if it's real time processing, it's not critical, right?  
I mean like we can process the data.  
During off peak hours to minimize content creation or like like like minimize like and for like contention like resources we can employ like parallel processing means like that means we can split your data into smaller manageable chunks and that can be processed in parallel and also we can.  
Choose the like right storage format like for example you can use a columnar formats like.  
Packet or like that are optimized for analytical like use like we can use table partitioning in databases to improve the query performance and also we can tune the resource allocations like.  
Scaling the resources like auto scaling features like auto scaling feature services like Google Cloud dataflow and we can also implement like intermediate catching.  
Too frequently, access like our intermediate results so that like we can use Google cloud memory like I mean so we can use Google Cloud memory store to help reduce load on database and speed up the like data retrieval and also we can implement like asynchronous processing.

 **Ramakrishna Medisetti** 28:06  
OK.  
That's fine. Yep. Let's say.  
Your user is experiencing the significant delays when running complex SQL query which involves multiple subqueries and aggregations or large datasets. Particularly, how do you not troubleshoot the these kind of issues?  
To optimize and resolve them.

 **maneesh** 28:32  
Yeah.  
Like.  
Two looks. First thing is like we should identify like where my query is taking time like probably we can check for like like for understanding whether my like whether it's an like like expensive operations that consume the most of the risk risk the like the resources such as.  
Full tables like such as full tables can or like large short functions.  
We can also check for.

 **Ramakrishna Medisetti** 29:09  
How do you how do you use any kind of feature in SQL now?

 **maneesh** 29:14  
Sorry.

 **Ramakrishna Medisetti** 29:15  
How do you check that I the full table scan are insufficient giants?  
How do you check that?  
Do you use any kind of feature in SQL?  
I mean in the sky.  
Kind of explain plan or something. Have you ever used that?

 **maneesh** 29:27  
Yeah, yeah, you can use like explain plan.  
Yes, like something called as unless command is there, right?  
Like explain plan, yeah.

 **Ramakrishna Medisetti** 29:36  
OK, OK.  
Yeah, like there are ways, right?  
So definitely like.

 **maneesh** 29:43  
Yeah, data.  
Yes, yes, definitely.

 **Ramakrishna Medisetti** 29:47  
Like like you're replacing them with the joints and you CTS for better reliability, right, consider indexing right.

 **maneesh** 29:54  
Yes.  
Yes, we can also use like appropriate data types like. We can refund your queries.  
We can set the limit results like we can use query catching like we can optimize join operations.

 **Ramakrishna Medisetti** 30:04  
No.  
Right.  
OK.

 **maneesh** 30:17  
Yes.

 **Ramakrishna Medisetti** 30:18  
Right, right, right.  
OK. And pythons, how good are you?  
Like, have you ever done the?  
All the libraries.

 **maneesh** 30:31  
Yeah, I'm pretty comfortable in Python.  
I can say.

 **Ramakrishna Medisetti** 30:37  
How do you handle missing data in dataframe using pandas?

 **maneesh** 30:44  
Uh.  
Use it.  
Can you can?  
Can you?  
Can you repeat the question?

 **Ramakrishna Medisetti** 30:49  
Yeah. How do you handle missing data in the data plane? Yeah.

 **maneesh** 30:52  
Yeah, OK, handling handling missing data.

 **Ramakrishna Medisetti** 30:54  
I need to see data data from you want us.

 **maneesh** 30:54  
Yeah, yeah.  
Raj, yeah.  
So we can use missing values like you can like you can drop in like in a function or we can also like impute those missing values by using like a misdrop function or like we by using like imputation technique.  
So where you can impute your missing values by.  
Like imputation like by median imputation techniques or like mode of imputation techniques.  
So where we are you like we we can use data frame like I mean we can use data frame is null, null, null function to check the to check the entire data frame whether it is having like missing values or not.

 **Ramakrishna Medisetti** 31:44  
OK. Can you give me some like the what is the particular function you call it like you do you use DF dot something like right?  
What is that functional function? You call it here?

 **maneesh** 31:55  
Yeah, like.  
Yeah, they have taught like df null, df.

 **Ramakrishna Medisetti** 32:04  
OK. Do you?

 **maneesh** 32:04  
Like dear dot, dot is null, null.

 **Ramakrishna Medisetti** 32:08  
OK, like there are multiple right DUP.

 **maneesh** 32:13  
Yeah.

 **Ramakrishna Medisetti** 32:15  
Yeah.

 **maneesh** 32:15  
Dear dot is like.

 **Ramakrishna Medisetti** 32:17  
OK. OK. OK.  
Fine. Yeah. Yeah, that's it.

 **maneesh** 32:19  
It is to check DF dot E is null to check whether you can like whether your data consists of any missing values.

 **Ramakrishna Medisetti** 32:28  
OK, Ben.  
How do you no mage two data frames in pandas and handle potential conflicts in the column names?

 **maneesh** 32:38  
Yeah. Yes you can.  
Like you can use merge function you can.  
Like you can merge or else you can also use like concatenate function to merge two data frames.

 **Ramakrishna Medisetti** 32:54  
OK.  
OK, how do you write an ETL script in Python that process large data set efficiently?  
We're considering both memory and time constraints.

 **maneesh** 33:09  
Yeah, in Python.  
Yeah, like we can.  
Like we can do, we can.  
We can write in a like like we can write in a sense like we like.  
We need to use a request library as library and also like Python pandas library and then we can use some particular functions especially with respect to like with with respect to using your SQL.  
Like 3 we can install that particular library like then we can.  
First step to extract by using your PD like like dot, read, under score, CSV or anything with respect to that particular file format like file path.  
And then we can transform the data like we can create a user defined function or like for transferring the data like for example you. If if you want to remove the rows with missing values. If you want to remove the duplicate values or if you want to like.  
Filter rows.  
You can use some filter function or we can also.  
Like use some load function where we can create a user defined load data into your.  
A particular table where we can create an SQL like connection.  
Like like by using SQL like 3 dot Connect database frame and then we can load the data like I mean load the data frame into SQL like tables then we can implement your ETL like process functions like and finally load the data like extract the data and also.  
Transform the data like that is what we can do.  
The and like you're like, OK.

 **Ramakrishna Medisetti** 34:58  
OK, OK.  
Fine, that's fine.  
Yeah, like, how do you, you know, handle like debugging in Python script?  
So which process large datasets and running out of memory?

 **maneesh** 35:10  
Yeah. So for.  
Like so for debugging in Python with respect to memory, right?  
So like generally we can go for like using like garbage collection and then we can also check it out.  
Then we can also identify the memory usage by using your SY like get size of function that is going to give us.  
A.  
Provide that.  
The size of a specific objects in bytes and then we can also drag the memory usage and like with trace malloc function in order to check the memory allocations like for your program. Then we can use like we can also.  
Use memory under score profiler which is a library for tracing the like memory usage in Python scripts.  
Then we can also like.  
Common like I mean memory. You can like memory just just for a large data set for considering using generators instead.  
And where?  
Like where memory leaks or last memory objects can be checked like like instead of at least to save memory efficient data like in data structure limit scope of variable garbage collections, all those stuff we can do it.

 **Ramakrishna Medisetti** 36:30  
OK, OK.  
That's great.  
Yeah. OK.  
I'm good, manisha.  
Like, do you have any questions for me?

 **maneesh** 36:40  
Yes.  
Yes, yes, I would like to know like what should be the roles and responsibilities of this particular position.  
Also, I would like to know what would be the next steps next steps in terms of interview.

 **Ramakrishna Medisetti** 36:54  
Yeah. Like these opportunities to work with.  
Prestigious client, right?  
So where we'll be playing, you know, neutral role like from the end to end handling the.

 **maneesh** 37:00  
Yeah.

 **Ramakrishna Medisetti** 37:08  
You know the complete.  
Data engine transmission right? And the conception creating the final reporting tables, right?  
Yeah, on the.

 **maneesh** 37:20  
Yeah.

 **Ramakrishna Medisetti** 37:23  
Cloud system.

 **maneesh** 37:25  
Yeah.

 **Ramakrishna Medisetti** 37:26  
And the process.  
Will be, you know, taken. Take it to paragr for the next levels of evaluations.

 **maneesh** 37:35  
OK.  
OK.

 **Ramakrishna Medisetti** 37:38  
What?  
Yeah. Any other questions?

 **maneesh** 37:43  
Yeah.  
That's it.  
I'm. I'm good. So like.

 **Ramakrishna Medisetti** 37:47  
OK.

 **maneesh** 37:49  
I'll you'll like the person will let the next steps, OK.

 **Ramakrishna Medisetti** 37:54  
Right, right.  
Yeah, like in the water.  
What about the the the farther rounds and everything?  
Yeah, the HR talent team.  
Now we'll get back to you part.

 **maneesh** 38:00  
Yeah, yeah.

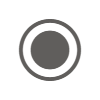
 **Ramakrishna Medisetti** 38:02  
No need to worry.  
Yeah. So it was great.  
Nice talking to you, Manish.

 **maneesh** 38:07  
Thank you.  
Thank you. Am a Vishnu.  
It was great talking to you.  
I had a good chat.

 **Ramakrishna Medisetti** 38:10  
Yeah.

 **maneesh** 38:13  
Thank you.

 **Raheem Khan** 38:24  
Hello.  
My name is.

 **maneesh** stopped transcription