

[Intro music] Hi everyone. In this lecture, we will talk about the Oracle Database Architecture. [No audio] In this lecture, you will learn how to perform the following. List and describe the major components of the database instance. List and describe the major Oracle database files. It is very important to understand the basic Oracle Database Architecture. You will need the knowledge that you receive in this lecture as long as you work with Oracle databases. It represents the foundation for the rest of the course lectures. So, please sit back, pay attention, and let's dive in.

[No audio] As with any running application, when an Oracle database is up and running in a system, it has two major groups of components. One group resides in the memory and one group saved in the storage. The group that resides in the memory is called the Database Instance, and the one that resides in the storage is called Database. I know this term may make you confused because we understand that those two groups together represents the database, so why one of them is called database. I really don't know, but this is the term that Oracle documentation uses to describe the database files. Anyway, it's not that big problem. All what we need to know that that group of components in the memory are collectively called Database Instance, and the one in the storage is called Database.

Database Instance is a group of memory areas and a group of processes. Each memory area is used to save specific data, and each process is responsible for a specific functionality. We will discuss about the basic areas and the processes as we go on with the lecture, and as we go on with the course. At the storage level, we have two types of files, data files, which contain the database data and system files, which are needed for the database operation. We will go deeper into those files later in this lecture. [No

audio] When a client process wants to connect to the database, it must first communicate with an external Oracle Process called the Listener. The Listener is not part of the Oracle database of processes, it runs by its own. However, for any client who wants to connect to the database via the network, it must communicate with the Listener first. The Listener initiates the connection between the client and the database. Once the connection is established, Oracle Database spawns a new process called Server Process.

The process represents the client in the database site. It executes whatever the client wants to achieve. For example, it reads the data from the memory for the client, or it writes the data into the memory for the client. [No audio] This means that the more clients connecting to the database, the more server processes will be spawned in the database to serve the client processes. There is another database configuration that makes the database behaved differently, but we don't need to worry about it now. We will cover it in

a separate lecture. As we just mentioned, the Database Instance constitutes from memory areas and Background Processes.

The memory part is divided into two major areas. They are named System Global Area or SGA, and Program Global Area or PGA. Again the memory is divided into two areas SGA and PGA. PGA is used to save memory areas required by the server processes. Each process has its own private area in the PGA. That's why the more client processes connecting to the database, the more PGA memory database needs to allocate for the clients. While the PGA is dedicated to the clients, the SGA is dedicated to the database. It contains the memory areas that Oracle database needs to continue its operation. SGA itself is divided into a few memory areas. We will cover them soon in this lecture. But you need to understand for now that SGA areas can be shared by multiple server processes, whereas the memory areas allocated to server processes in the PGA are private. Each server process has its own session information saved in the PGA.

Server processes can read from the SGA or write into it, depending on the operations that the client wants to perform.

[No audio] This is a summary of what we discussed in the previous slide. [No audio] Let's bring the magnifying glass closer to the SGA and see what's inside it. The SGA contains the following basic memory areas, Shared Pool area, Database Buffer Cache, Redo Log Buffer, Large Pool, and the fixed SGA. There could be more memory areas configured in the SGA. But those are the basic ones. In the figure in the slide, the areas with solid lines are mandatory areas. The database doesn't work without having those areas in the SGA. The other areas are optional. Let's see what each memory area is used for.

Shared Pool is used to save the SQL statements and PL/SQL code executed by the clients in the sub area called Library Cache. It also contains the following other sub areas, Data Dictionary Cache, Server Result Cache, and Reserved pool. The Data Dictionary Cache holds information about the accessed database objects. It doesn't contain data saved in the object, it contains data about the object. This information is called Data Dictionary. This is needed by Oracle database when executing the statements. Server Result Cache holds the result sets. It contains the SQL Query Result Cache, and the PL/SQL Function Result Cache.

By default this area is not configured. It could be configured by the developers for enhancing the performance of a specific SQL statements, or PL/SQL code. Another newer area in SGA is the Database Buffer Cache. This area is used to save the copies of the data blocks. When the clients submit the queries to the database to retrieve some data, by default, the data blocks that contain the required data are not read from the data files in

the storage, straight to the client process. They are saved in the buffer cache first to optimize the I/O operation. If later the same data is required, the server process doesn't need to read it again from the storage, it reads the required data from the memory, and sends it back to the requester. Obviously, this is much faster than reading the data from the storage. Another memory area in the SGA is the Large Pool. Although it is technically optional, but it is configured in most production databases, because usually it doesn't take much memory. Large Pool memory is used for buffers during backup and recovery operations, buffers for different inserts, message buffers used in parallel executions, and some others. The next memory area is the Redo Log Buffer. This memory is used to save information about the changes made on the data by DML and DDL statements. Internally, the changes made to the data is saved in records called redo entries.

Redo entries have the information to reconstruct or to redo the changes made to the data. They are not copies of the changed data itself. They contain information enough to redo the changes made on the data. [No audio] This is a summary of what we discussed in the previous slide. [No audio] In Oracle databases, user data is saved in a small storage units called Data Blocks. When the rows are retrieved by the clients, the data blocks containing the required rows are read from the storage and saved in the data buffer cache. For the sake of simplicity, data buffer cache memory units could be called just buffer. The buffer could take one of the following states, Unused, Clean, or Dirty.

Unused buffer is a buffer that has been allocated but never been used or accessed. This is the state of many buffers when the database instance is started up. The buffer take the clean state if it contains data blocks that are not changed or changed and committed. When a block is read from the data file and saved into an unused buffer, the buffer states turns into clean. If later or more is modified in the data block, the data block state turns into dirty. It remains on this state till the modified block is written to the data file in the disk. This state has given this negative term because it represents an overhead on the database engine to manage the modified or dirty blocks. Understanding these states is useful for understanding some other concepts later in this course. It also helps when optimizing data buffer cache size.

[No audio] We finished from discussing the memory areas. Now it's time to cover the background processes. There are a very long list of processes that could run when an Oracle Database Instance is in operation. We will cover in this lecture the basic processes, the processes that run with every Oracle database. [No audio] We will start with the DB Writer. The major responsibility of the DB Writer is to write the modified blocks, or in other

words, the dirty blocks from the buffer cache to the data files in the storage. The process doesn't do that immediately after the blocks are modified. It writes the dirty blocks in a lazy manner. It does take some time to write the dirty blocks now and then. There are a few specific situations where the DB Writer is forced to write the dirty blocks to the storage. But it doesn't really matter to know them now. All what you need to understand is that the task of the DB Writer is to write the dirty blocks to the storage. In most systems, having a single DB Writer is more than enough.

However, in some under high stress databases, we may need to configure more than one DB Writer process. You might be wondering, what will happen to the modified blocks if the Database Instance crashes for any reason. How the data will recover the modified blocks that haven't been written to the disk yet. Later in the course, we will dedicate a lecture to discuss this question in more details. [No audio] The next process to discuss is the Log Writer process. This process is responsible for writing redo entries from the Redo Log Buffer to the Redo Log Files. Remember, the redo entries contain the information to reproduce the changes made to the data. The Log Writer writes the redo entries very frequently. Specifically, the process writes the redo entries in the following circumstances. A user commits a transaction, with each commit, the Log Writer writes the redo log entry from the buffer to the online redo log files. The second situation is when an online redo log switch occurs. The third situation is when three seconds have passed since the Log Writer last wrote. The next situation is when the Redo Log Buffer is 1/3 full, or it contains 1 megabyte of buffer data, and finally, when the DB Writer must write the modified buffers to the disk.

Because the Log Writer writes the redo log entries in all those situations, we can see that it would be very active in any database. For this reason, the redo log buffer size is normally very small in its size, compared to the other memory area sizes. The database keeps catching up with a redo generated in the redo log buffer. The redo log entries generated by the database are used for recovery purposes. They are used during the instance recovery after its crash and for backup and recovery and for creating Data Guard configuration. After we understand about the DB Writer and the Log Writer processes, we can reach to the following conclusion. During the normal database operation, some modified blocks are saved in the data files and some of them remain in the memory waiting for the DB Writer to save them in the storage. But the information about all the changes made by the committing transactions are saved in the redo log files.

[No audio] The next process to discuss is the Checkpoint Process, CKPT. Checkpoint Process performs two actions. One, it updates the control file and the data file headers with checkpoint information. Two, it signals that DB Writer to write the modified blocks to the disk. But what is the checkpoint information in first place? Checkpoint information is the information that tells the database engine up to what time the data changes have been saved in the storage. If the Database Instance crashed, and then it has started up, the database engine wants to know which data was saved in the data files and which modified data is missing because it was in the memory at the time of the instance crash. From the checkpoint information, the database engine could know that. From the checkpoint information, the database engine could know to which point the modified data was saved in the data files. Remember, the checkpoint information is saved in the control file and in the data file headers. In the data file headers, not in the data blocks. [No audio] Here is a summary of the Mandatory Background Processes in Oracle databases. We have already discussed some of them. Let's discuss about the ones that we have missed. System Monitor Process or SMON is responsible for performing some system-level clean duties, like instance recovery, recovering terminated transactions, and others. Listener Registration Process or LREG is responsible for registering the Database Instance in the Listener. This process is a new process. In Oracle 11g, it was the PMON responsibility to register the Database Instances in the Listener. Finally we have the PMON group of processes. They are responsible for the monitoring and cleanup of the other processes.

[No audio]

[Intro music] Hi everyone. In this lecture, we will talk about the Oracle Database Architecture. [No audio] In this lecture, you will learn how to perform the following. List and describe the major components of the database instance. List and describe the major Oracle database files. It is very important to understand the basic Oracle Database Architecture. You will need the knowledge that you receive in this lecture as long as you work with Oracle databases. It represents the foundation for the rest of the course lectures. So, please sit back, pay attention, and let's dive in. [No audio] As with any running application, when an Oracle database is up and running in a system, it has two major groups of components. One group resides in the memory and one group saved in the storage. The group that resides in the memory is called the Database Instance, and the one that resides in the storage is called Database. I know this term may make you confused because we understand that those two groups together represents the database, so why one of them is called database. I really don't know, but this is the term that Oracle documentation uses to describe the database files. Anyway, it's not that big problem. All what we need to know that that group of components in the memory are collectively called Database Instance, and the one in the storage is called Database. Database Instance is a group of memory areas and a group of processes. Each memory area is used to save specific data, and each process is responsible for a specific functionality. We will discuss about the basic areas and the processes as we go on with the lecture, and as we go on with the course. At the storage level, we have two types of files,

data files, which contain the database data and system files, which are needed for the database operation. We will go deeper into those files later in this lecture. [No audio] When a client process wants to connect to the database, it must first communicate with an external Oracle Process called the Listener. The Listener is not part of the Oracle database of processes, it runs by its own. However, for any client who wants to connect to the database via the network, it must communicate with the Listener first. The Listener initiates the connection between the client and the database. Once the connection is established, Oracle Database spawns a new process called Server Process. The process represents the client in the database site. It executes whatever the client wants to achieve. For example, it reads the data from the memory for the client, or it writes the data into the memory for the client. [No audio] This means that the more clients connecting to the database, the more server processes will be spawned in the database to serve the client processes. There is another database configuration that makes the database behaved differently, but we don't need to worry about it now. We will cover it in a separate lecture. As we just mentioned, the Database Instance constitutes from memory areas and Background Processes. The memory part is divided into two major areas. They are named System Global Area or SGA, and Program Global Area or PGA. Again the memory is divided into two areas SGA and PGA. PGA is used to save memory areas required by the server processes. Each process has its own private area in the PGA. That's why the more client processes connecting to the database, the more PGA memory database needs to allocate for the clients. While the PGA is dedicated to the clients, the SGA is dedicated to the database. It contains the memory areas that Oracle database needs to continue its operation. SGA itself is divided into a few memory areas. We will cover them soon in this lecture. But you need to understand for now that SGA areas can be shared by multiple server processes, whereas the memory areas allocated to server processes in the PGA are private. Each server process has its own session information saved in the PGA. Server processes can read from the SGA or write into it, depending on the operations that the client wants to perform. [No audio] This is a summary of what we discussed in the previous slide. [No audio] Let's bring the magnifying glass closer to the SGA and see what's inside it. The SGA contains the following basic memory areas, Shared Pool area, Database Buffer Cache, Redo Log Buffer, Large Pool, and the fixed SGA. There could be more memory areas configured in the SGA. But those are the basic ones. In the figure in the slide, the areas with solid lines are mandatory areas. The database doesn't work without having those areas in the SGA. The other areas are optional. Let's see what each memory area is used for. Shared Pool is used to save the SQL statements and PL/SQL code executed by the clients in the sub area called Library Cache. It also contains the following other sub areas, Data Dictionary Cache, Server Result Cache, and Reserved pool. The Data Dictionary Cache holds information about the accessed database objects. It doesn't contain data saved in the object, it contains data about the object. This information is called Data Dictionary. This is needed by Oracle database when executing the statements. Server Result Cache holds the result sets. It contains the SQL Query Result Cache, and the PL/SQL Function Result Cache. By default this area is not configured. It could be configured by the developers for enhancing the performance of a specific SQL statements, or PL/SQL code. Another newer area in SGA is the Database Buffer Cache. This area is used to save the copies of the data blocks. When the clients submit the queries to the database to retrieve some data, by default, the data blocks that contain the required data are not read from the data files in the storage, straight to the client process. They are saved in the buffer cache first to optimize the I/O operation. If later the same data is required, the server process doesn't need to read it again from the storage, it reads the required data from the memory, and sends it back to the requester. Obviously, this is much faster than reading the data from the storage. Another memory area in the SGA is the Large Pool. Although it is technically optional, but it

is configured in most production databases, because usually it doesn't take much memory. Large Pool memory is used for buffers during backup and recovery operations, buffers for different inserts, message buffers used in parallel executions, and some others. The next memory area is the Redo Log Buffer. This memory is used to save information about the changes made on the data by DML and DDL statements. Internally, the changes made to the data is saved in records called redo entries. Redo entries have the information to reconstruct or to redo the changes made to the data. They are not copies of the changed data itself. They contain information enough to redo the changes made on the data. [No audio] This is a summary of what we discussed in the previous slide. [No audio] In Oracle databases, user data is saved in a small storage units called Data Blocks. When the rows are retrieved by the clients, the data blocks containing the required rows are read from the storage and saved in the data buffer cache. For the sake of simplicity, data buffer cache memory units could be called just buffer. The buffer could take one of the following states, Unused, Clean, or Dirty. Unused buffer is a buffer that has been allocated but never been used or accessed. This is the state of many buffers when the database instance is started up. The buffer take the clean state if it contains data blocks that are not changed or changed and committed. When a block is read from the data file and saved into an unused buffer, the buffer states turns into clean. If later or more is modified in the data block, the data block state turns into dirty. It remains on this state till the modified block is written to the data file in the disk. This state has given this negative term because it represents an overhead on the database engine to manage the modified or dirty blocks. Understanding these states is useful for understanding some other concepts later in this course. It also helps when optimizing data buffer cache size. [No audio] We finished from discussing the memory areas. Now it's time to cover the background processes. There are a very long list of processes that could run when an Oracle Database Instance is in operation. We will cover in this lecture the basic processes, the processes that run with every Oracle database. [No audio] We will start with the DB Writer. The major responsibility of the DB Writer is to write the modified blocks, or in other words, the dirty blocks from the buffer cache to the data files in the storage. The process doesn't do that immediately after the blocks are modified. It writes the dirty blocks in a lazy manner. It does take some time to write the dirty blocks now and then. There are a few specific situations where the DB Writer is forced to write the dirty blocks to the storage. But it doesn't really matter to know them now. All what you need to understand is that the task of the DB Writer is to write the dirty blocks to the storage. In most systems, having a single DB Writer is more than enough. However, in some under high stress databases, we may need to configure more than one DB Writer process. You might be wondering, what will happen to the modified blocks if the Database Instance crashes for any reason. How the data will recover the modified blocks that haven't been written to the disk yet. Later in the course, we will dedicate a lecture to discuss this question in more details. [No audio] The next process to discuss is the Log Writer process. This process is responsible for writing redo entries from the Redo Log Buffer to the Redo Log Files. Remember, the redo entries contain the information to reproduce the changes made to the data. The Log Writer writes the redo entries very frequently. Specifically, the process writes the redo entries in the following circumstances. A user commits a transaction, with each commit, the Log Writer writes the redo log entry from the buffer to the online redo log files. The second situation is when an online redo log switch occurs. The third situation is when three seconds have passed since the Log Writer last wrote. The next situation is when the Redo Log Buffer is 1/3 full, or it contains 1 megabyte of buffer data, and finally, when the DB Writer must write the modified buffers to the disk. Because the Log Writer writes the redo log entries in all those situations, we can see that it would be very active in any database. For this reason, the redo log buffer size is normally very small in its size, compared to the other memory area sizes. The database

keeps catching up with a redo generated in the redo log buffer. The redo log entries generated by the database are used for recovery purposes. They are used during the instance recovery after its crash and for backup and recovery and for creating Data Guard configuration. After we understand about the DB Writer and the Log Writer processes, we can reach to the following conclusion. During the normal database operation, some modified blocks are saved in the data files and some of them remain in the memory waiting for the DB Writer to save them in the storage. But the information about all the changes made by the committing transactions are saved in the redo log files. [No audio] The next process to discuss is the Checkpoint Process, CKPT. Checkpoint Process performs two actions. One, it updates the control file and the data file headers with checkpoint information. Two, it signals that DB Writer to write the modified blocks to the disk. But what is the checkpoint information in first place? Checkpoint information is the information that tells the database engine up to what time the data changes have been saved in the storage. If the Database Instance crashed, and then it has started up, the database engine wants to know which data was saved in the data files and which modified data is missing because it was in the memory at the time of the instance crash. From the checkpoint information, the database engine could know that. From the checkpoint information, the database engine could know to which point the modified data was saved in the data files. Remember, the checkpoint information is saved in the control file and in the data file headers. In the data file headers, not in the data blocks. [No audio] Here is a summary of the Mandatory Background Processes in Oracle databases. We have already discussed some of them. Let's discuss about the ones that we have missed. System Monitor Process or SMON is responsible for performing some system-level clean duties, like instance recovery, recovering terminated transactions, and others. Listener Registration Process or LREG is responsible for registering the Database Instance in the Listener. This process is a new process. In Oracle 11g, it was the PMON responsibility to register the Database Instances in the Listener. Finally we have the PMON group of processes. They are responsible for the monitoring and cleanup of the other processes. [No audio] [Intro music] Hi everyone. In this lecture, we will talk about the Oracle Database Architecture. [No audio] In thi: Added to Selection. Press [CTRL + S] to save as a note : Added to Selection. Press [CTRL + S] to save as a note

[No audio] We went through the Oracle database instance memory areas and background processes. Let's switch to the storage size and go through the database files saved in the disk. The slide is demonstrating a list of the files that together constitute the database in the storage. The boxes with solid lines are files that are mandatory for database operations. The boxes with the broken borders are optional files that are configured based on the DBA demands. Let's go through the files and scratch the surface of each file type. You just need to understand the roles. As we go on with the course, you will learn more about managing each file type.

Control Files, Control Files are used to save the database physical structure. This includes the database file names and their locations, the backup files, and their database basic information like creation time and last checkpoint. Technically speaking, Oracle database instance needs only a single control file to operate. However, in most production databases, more than one control file are configured. No Oracle database can operate without control files, they must be there otherwise the instance goes down The

Parameter File. The Parameter File is a small file to save the parameter-value pairs.

The parameters saved in the file are called Initialization Parameters.

Initialization Parameters are configuration parameters that control the instance behavior.

As you go with the course, you will learn more and more initialization parameters.

Online Redo Log Files. Online Redo Log Files are two or more pre-allocated files that

store changes to the database as they occur. As we learned earlier, the redo log writer

writes the redo entries from the redo log buffer to the online redo log files. These

files typically used for instance recovery generating archived redo log files, replication,

and Data Guard. System Data Files.

System Data Files are data files which contain data used by Oracle internal operations.

You can think of them as the Windows folder in Windows platform. Windows

folder contains a set of files that are used solely by Windows operating system. The users

should not save their own files in the Windows folder. Going with this analog, as Windows

cannot operate if its Windows folder is gone, Oracle database cannot operate, if its system

data files are deleted. They are mandatory for normal database operation. The

Automatic Diagnostic Repository or ADR. ADR is simply a directory structure where the

database saves data such as trace files, the alert log file, DDL log files, and the

health monitor reports. We normally refer to the ADR for maintenance

and troubleshooting purposes. User Tablespace Data Files. User Tablespace Data Files are

the files that contain the user data. Those files contain the data that the entire RDBMS is

built for. They have the most valuable asset in any IT system. However, the

database instance can continue its normal operation, if a user data file is lost. It's true that

the end user cannot access the data in the lost data file. But the database instance can go

on with its operation without a crash. Backup Files. Backup Files are used to restore the

database in case of data losses. Archived Redo Log Files or Archive Logs. Archive Logs are

offline copies of the online redo log files when they are get filled by the redo entries.

They are not considered part of the database files, but they are important for data

recovery.

The Passwords File. A database instance may work with a single password file. A password

file is a file to store the passwords of the database users with the administrative

privileges. Of course the passwords are encrypted in the file. Keystore or Wallet file.

A keystore is a file to store the encryption keys used by the database. Block Change

Tracking or BCT files are used when the BCT is enabled. Flashback Logs are files that get

generated by the database when the flashback option is enabled. As I said, as you go

on with the course, you will learn the details about managing each file type.

[No audio] Here is a summary of what we explained in the previous slide. [No audio] What you have learned in this lecture are the basics of the Oracle internals. There are plenty of books and articles that explain about Oracle internals in further details. However, from my experience, you don't really need to know deep about Oracle internals when you start learning and using Oracle databases. As you go on with the course and as you keep using Oracle databases, you keep learning about new processes and new concepts on Oracle internals. For now, what was covered is fair enough. [No audio] And that's it for this lecture.

By the end of this lecture, you should be able to perform the following. List and describe the major components of the database instance and list and describe the major

Oracle database files. See you guys in the next lecture. Thank you very much. [No audio]

[No audio] We went through the Oracle database instance memory areas and background processes. Let's switch to the storage size and go through the database files saved in the disk.

The slide is demonstrating a list of the files that together constitute the database in the storage. The boxes with solid lines are files that are mandatory for database operations. The boxes with the broken borders are optional files that are configured based on the DBA demands. Let's go through the files and scratch the surface of each file type. You just need to understand the roles. As we go on with the course, you will learn more about managing each file type. Control Files, Control Files are used to save the database physical structure. This includes the database file names and their locations, the backup files, and their database basic information like creation time and last checkpoint. Technically speaking, Oracle database instance needs only a single control file to operate. However, in most production databases, more than one control file are configured. No Oracle database can operate without control files, they must be there otherwise the instance goes down The Parameter File. The Parameter File is a small file to save the parameter-value pairs. The parameters saved in the file are called Initialization Parameters. Initialization Parameters are configuration parameters that control the instance behavior. As you go with the course, you will learn more and more initialization parameters. Online Redo Log Files. Online Redo Log Files are two or more pre-allocated files that store changes to the database as they occur. As we learned earlier, the redo log writer writes the redo entries from the redo log buffer to the online redo log files. These files typically used for instance recovery generating archived redo log files, replication, and Data Guard. System Data Files. System Data Files are data files which contain data used by Oracle internal operations. You can think of them as the Windows folder in Windows platform. Windows folder contains a set of files that are used solely by Windows operating system. The users should not save their own files in the Windows folder. Going with this analog, as Windows cannot operate if its Windows folder is gone, Oracle database cannot operate, if its system data files are deleted. They are mandatory for normal database operation. The Automatic Diagnostic Repository or ADR. ADR is simply a directory structure where the database saves data such as trace files, the alert log file, DDL log files, and the health monitor reports. We normally refer to the ADR for maintenance and troubleshooting purposes. User Tablespace Data Files. User Tablespace Data Files are the files that contain the user data. Those files contain the data that the entire RDBMS is built for. They have the most valuable asset in any IT system. However, the database instance can continue its normal operation, if a user data file is lost. It's true that the end user cannot access the data in the lost data file. But the database instance can go on with its operation without a crash. Backup Files. Backup Files are used to restore the database in case of data

losses. Archived Redo Log Files or Archive Logs. Archive Logs are offline copies of the online redo log files when they are get filled by the redo entries. They are not considered part of the database files, but they are important for data recovery. The Passwords File. A database instance may work with a single password file. A password file is a file to store the passwords of the database users with the administrative privileges. Of course the passwords are encrypted in the file. Keystore or Wallet file. A keystore is a file to store the encryption keys used by the database. Block Change Tracking or BCT files are used when the BCT is enabled. Flashback Logs are files that get generated by the database when the flashback option is enabled. As I said, as you go on with the course, you will learn the details about managing each file type. [No audio] Here is a summary of what we explained in the previous slide. [No audio] What you have learned in this lecture are the basics of the Oracle internals. There are plenty of books and articles that explain about Oracle internals in further details. However, from my experience, you don't really need to know deep about Oracle internals when you start learning and using Oracle databases. As you go on with the course and as you keep using Oracle databases, you keep learning about new processes and new concepts on Oracle internals. For now, what was covered is fair enough. [No audio] And that's it for this lecture. By the end of this lecture, you should be able to perform the following. List and describe the major components of the database instance and list and describe the major Oracle database files. See you guys in the next lecture. Thank you very much. [No audio] [No audio] We went through the Oracle database instance memory areas and background processes. Let's switch to the storage size and go through the database files saved in the disk. The slide is demonstratin: Added to Selection. Press [CTRL + S] to save as a note : Added to Selection. Press [CTRL + S] to save as a note