Summary

Thank you for joining us in this module on debugging DAGS in Airflow.

1. We have learned some valuable techniques that will help you identify and resolve issues that may arise when working with Airflow.
2. Firstly, we looked at the basic checks that we can perform on our Airflow settings such as checking the wait intervals and the scaling parameters for our DAG.
3. We then learned how to validate that our DAGs and modules are present at the correct location and our DAG has been instantiated properly and that any dependencies are resolved correctly.
4. Next, we explored how to check the health and logs of our scheduler to make sure our DAGs have been parsed and are being scheduled properly.
5. We also learned how to verify the integrity of external systems that our DAGs may interact with, such as databases, and discussed the necessity to verify whether these systems allow connections to be established.
6. Finally, we also looked at how to resolve dependency conflicts between various DAGs by using operators such as the KubernetesPodOperator, the ExternalPythonOperator, or the PythonVirtualEnvironment operator. These three operators allow us to isolate our DAG and any of its dependencies from our main Airflow to avoid any conflicts that may arise.

With these techniques, we can ensure that our DAGs run smoothly and that any issues that may arise can be

quickly identified and resolved.

Thank you for tuning in and happy debugging.