# Java – AAD – SQLDB – Connect PoC

Reference: <https://docs.microsoft.com/en-us/sql/connect/jdbc/connecting-using-azure-active-directory-authentication?view=sql-server-2017>

Client Prerequisites:

* Java 7 or above
* Download Microsoft JDBC Driver 6.0 (or higher) for SQL Server

https://www.microsoft.com/en-ca/download/details.aspx?id=11774

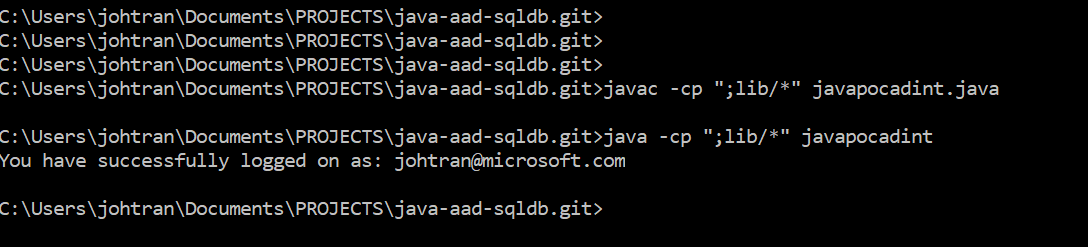
* azure-activedirectory-library-for-java and its dependencies.

Clone the github repo and build the project:

<https://github.com/AzureAD/azure-activedirectory-library-for-java>

Note on dependencies: <https://docs.microsoft.com/en-us/sql/connect/jdbc/feature-dependencies-of-microsoft-jdbc-driver-for-sql-server?view=sql-server-2017>

Sample Output: sample will connect to DB, select username and print to screen.



# ActiveDirectoryIntegrated Authentication

1. Create a new SQL server and Database or use an existing DB. Create a AD user on the Database and make sure that they have Connect permissions.
2. On your client machine set your Kerberos Ticket: <https://docs.microsoft.com/en-us/sql/connect/jdbc/connecting-using-azure-active-directory-authentication?view=sql-server-2017#set-kerberos-ticket-on-windows-linux-and-mac>
3. Within the sample code; replace the server/database name with your target server and DB.
4. Run the sample application
5. Expected output: You have successfully logged on as: <your domain user name>

Select fields from DB and display to console screen.

# ActiveDirectoryPassword Authentication

1. Create a new SQL server and Database or use an existing DB. Create a AD user on the Database and make sure that they have Connect permissions.
2. Within the sample code; replace the server/database name with your target server and DB.
3. Within the sample code; set your AD user and Password.
4. Run the sample application
5. Expected output: You have successfully logged on as: <your domain user name>

Select fields from DB and display to console screen.