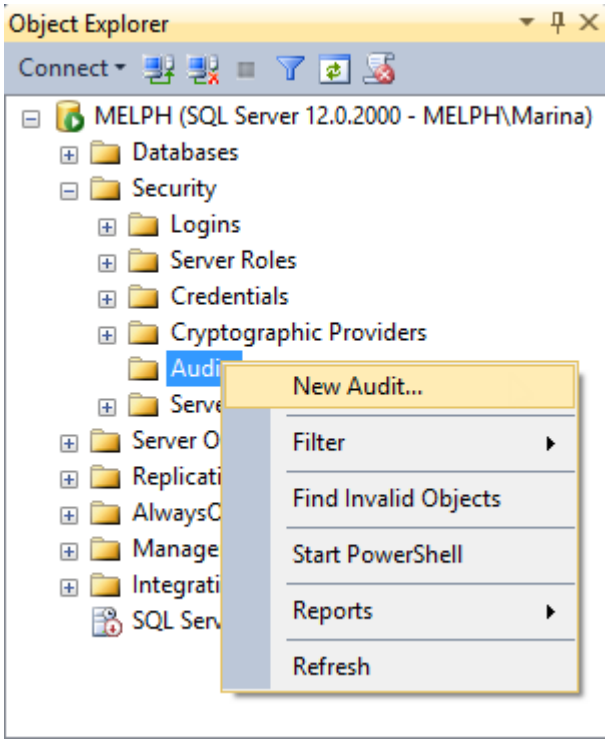
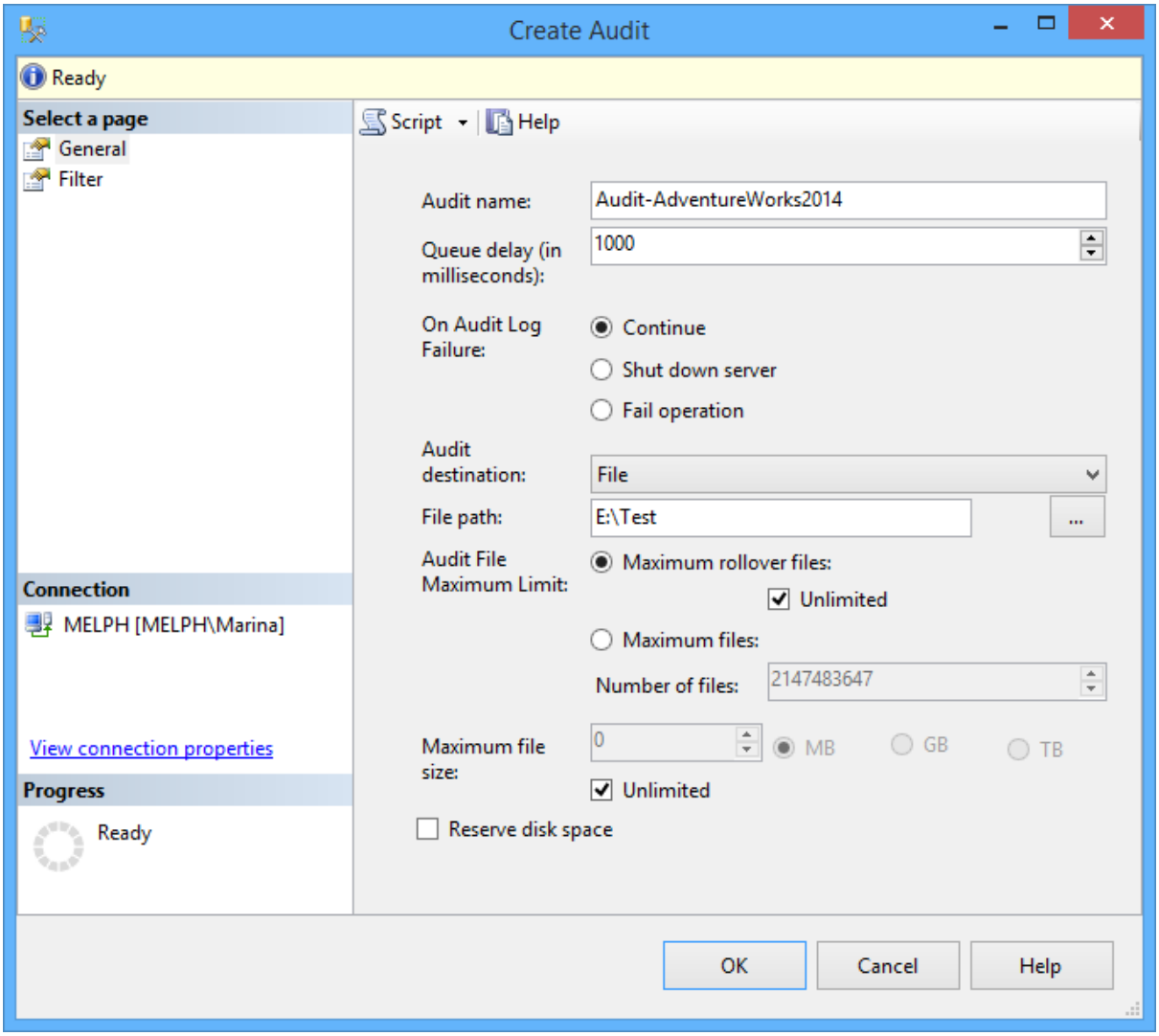
To create a new SQL Server Audit object:

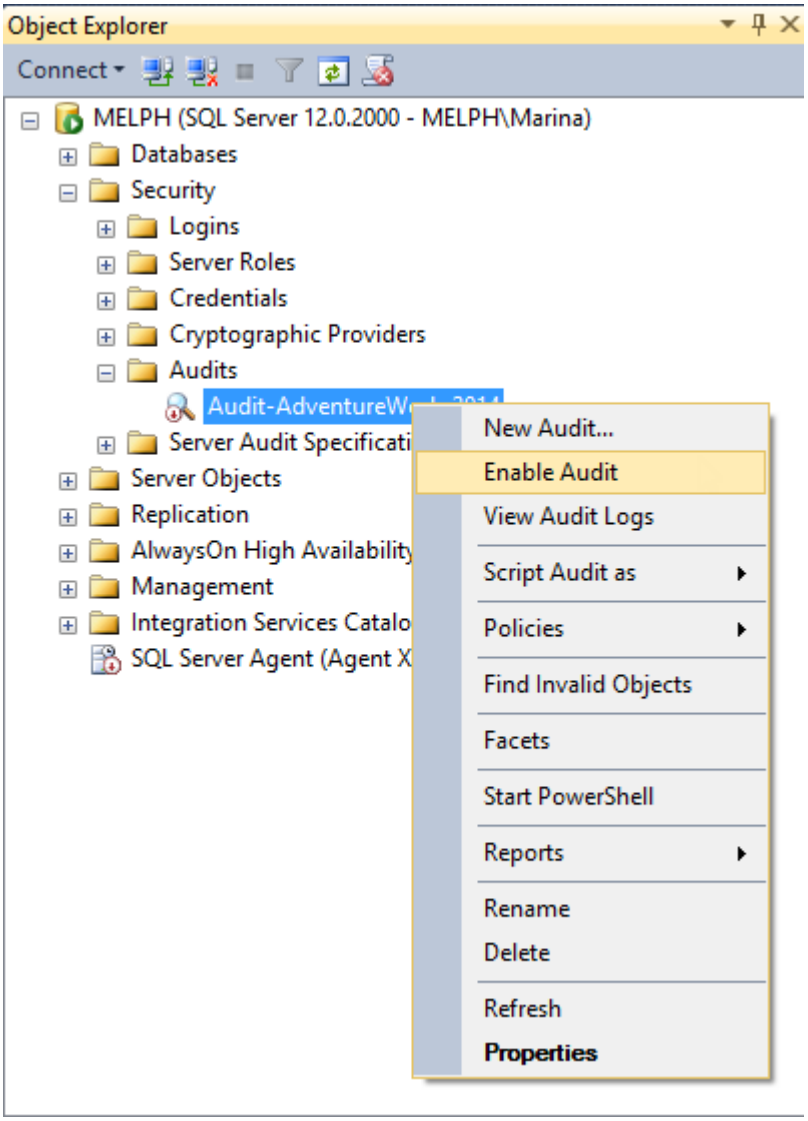
1. Expand Security and right-click Audits in SSMS
2. Select **New Audit**



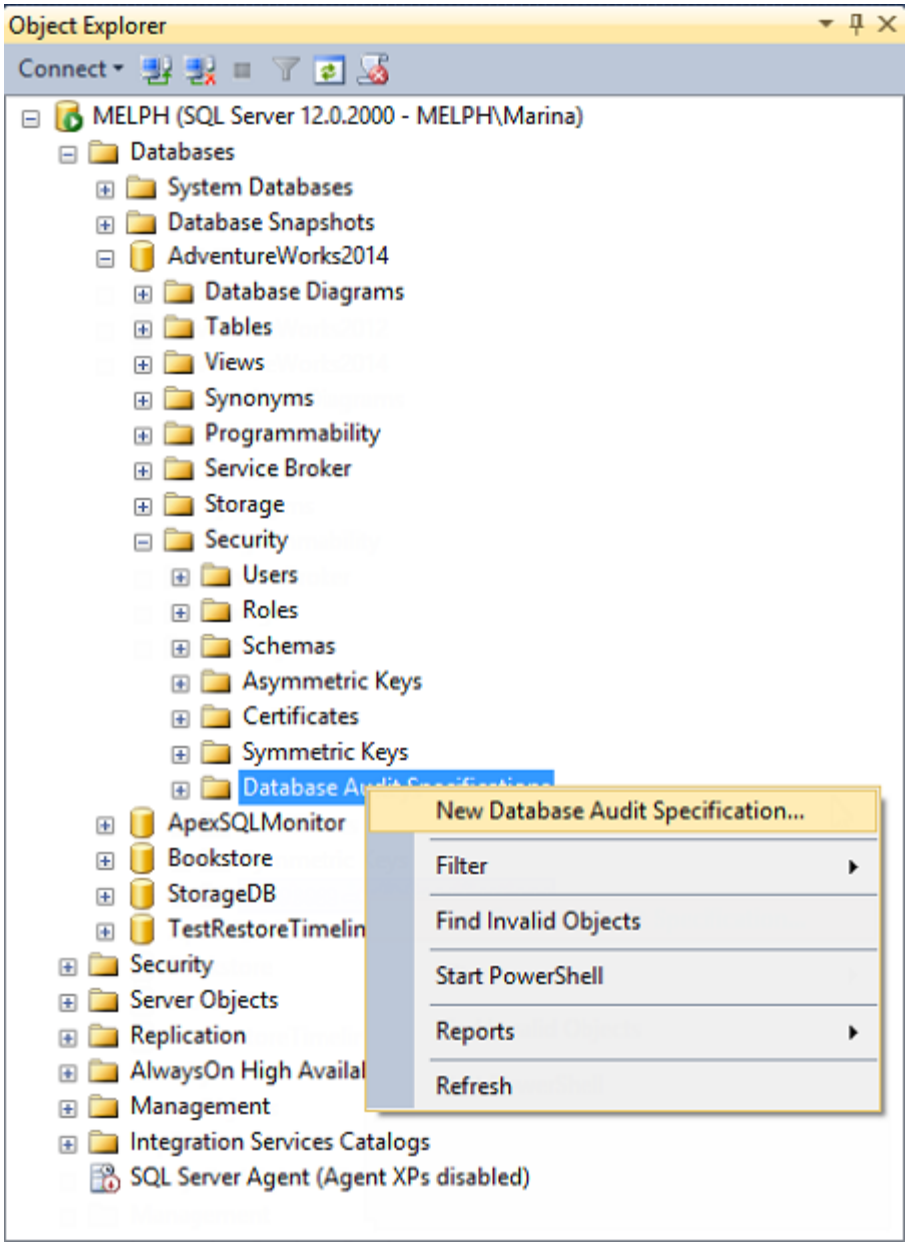
3. You will need to create a name for the audit, and then indicate whether to store the audit data in an application security event log, event log or a file. Finally, indicate a location for the audit file



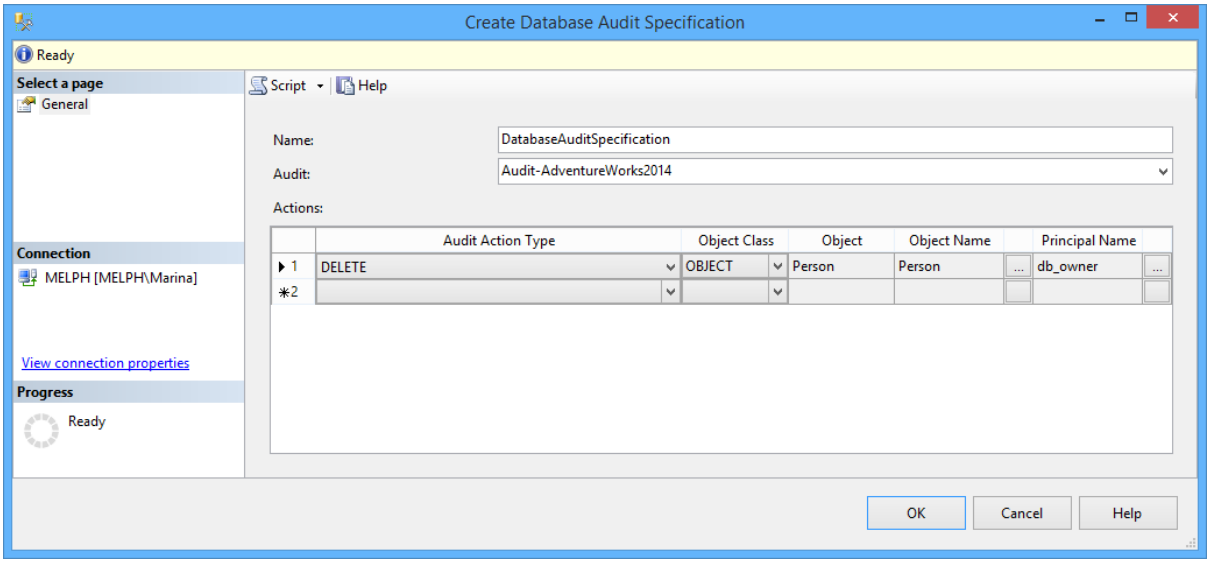
4. Click OK and your audit will appear in the Audits node of the Object Explorer
5. By default, it's disabled. The disabled status is indicated by a red arrow. Right-click and select **Enable Audit**, to enable



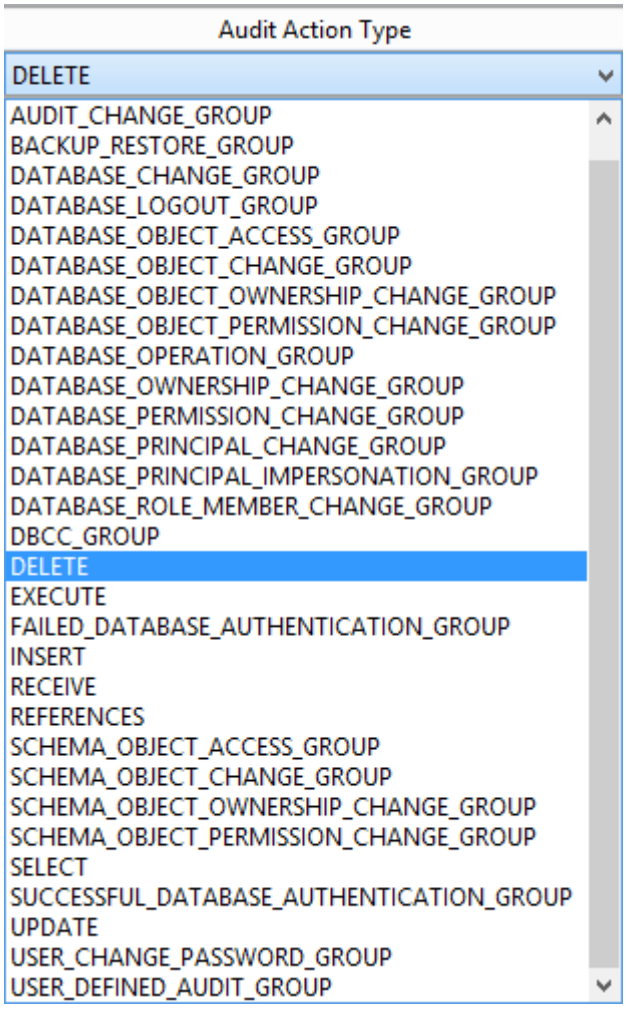
6. Depending on whether you want to audit the activity on an entire SQL Server instance or just a particular database you will choose between Server Audit Specification or Database Audit Specification
7. For a Database Audit Specification, expand the node of the database to audit, go to **Security**, right-click **Database Audit Specifications** and select **New Database Audit**



8. In the Create Database Audit Specification dialog, indicate the specification name, associate the specification with the audit object created in the previous step, specify the activity to audit in the Audit Action Type. For auditing a particular database, indicate the database, object, or schema as an Object Class, the name of the audited object, and the audited login

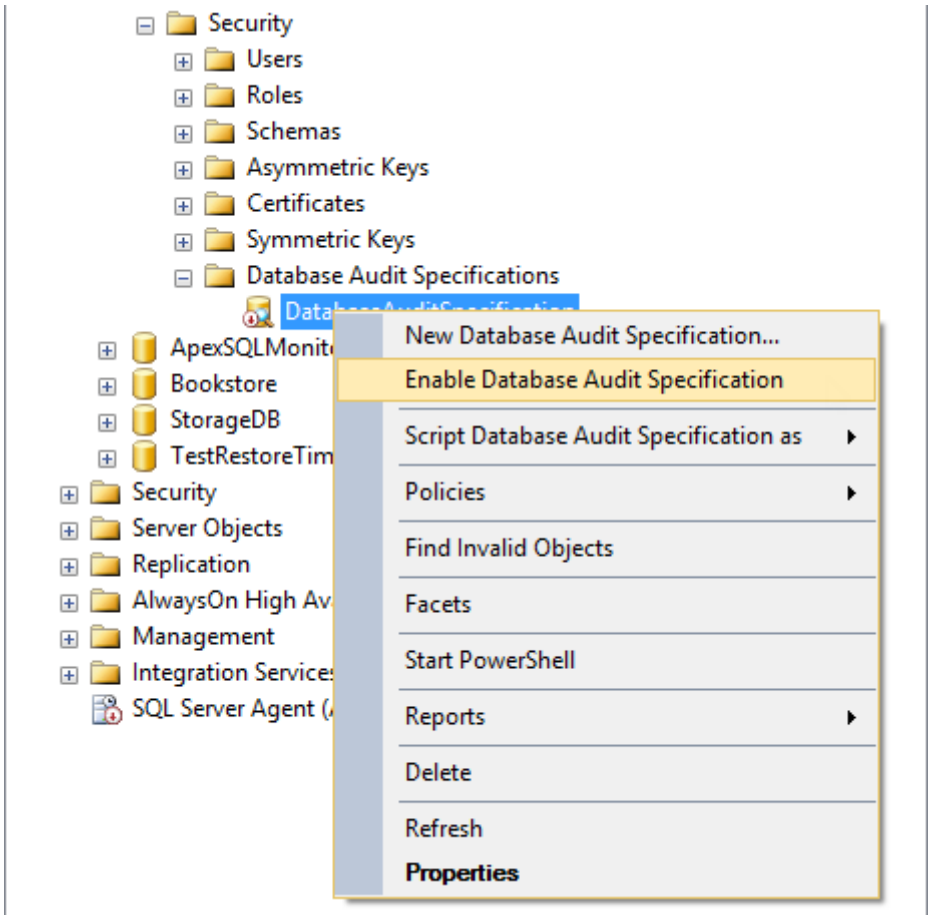


In the drop-down list for Audit Action Type, you can see all actions that can be audited using SQL Server Auditing



In this dialog, you will specify the user accounts to be monitored.

Database audit specifications are disabled, by default. To enable them, select this option in the context menu



Now, all DELETE statements executed against the Person.BusinessEntityAddress will be audited and inserted into files the names of which start with Audit-, such as Audit-AW2012Test_9D93CA4A-8B90-40B8-8B0B-FCBDA77B431D_0_130161593310500000.sqlaudit, and stored in E:\

For high volume databases, it is considered a good practice to save audited info to a file. This file can't be opened directly though, even with a hex editor. To view it, use fn_get_audit_file

For example:

```
SELECT event_time,action_id, statement, database_name, server_principal_name
 FROM fn_get_audit_file( 'E:\Test\Audit-*.sqlaudit' , DEFAULT , DEFAULT);
```

shows the following results:

	event_time	action_id	statement	database_name	server_principal_name
1	2013-06-19 23:48:51.0799953	AUSC			Fujitsu\Milena

The results will include columns but not the actual deleted data. The user, who made the deletion, and when the deletion was made are the main audit elements here, which is a disadvantage for users who need more comprehensive audit data.