In this chapter, we will discuss the different Database Methods in ArangoDB.

To start with, let us get the properties of the Database −

* Name
* ID
* Path

First, we invoke the Arangosh. Once, Arangosh is invoked, we will list the databases we created so far −

We will use the following line of code to invoke Arangosh −

127.0.0.1:8529@\_system> db.\_databases()

Output

[

"\_system",

"song\_collection"

]

We see two databases, one **\_system** created by default, and the second **song\_collection** that we have created.

Let us now shift to song\_collection database with the following line of code −

127.0.0.1:8529@\_system> db.\_useDatabase("song\_collection")

Output

true

127.0.0.1:8529@song\_collection>

We will explore the properties of our song\_collection database.

To find the name

We will use the following line of code to find the name.

127.0.0.1:8529@song\_collection> db.\_name()

Output

song\_collection

To find the id −

We will use the following line of code to find the id.

song\_collection

Output

4838

To find the path −

We will use the following line of code to find the path.

127.0.0.1:8529@song\_collection> db.\_path()

Output

/var/lib/arangodb3/databases/database-4838

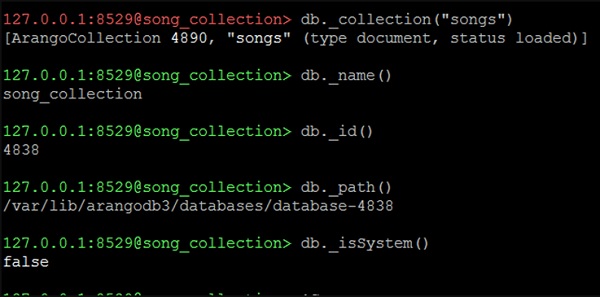
Let us now check if we are in the system database or not by using the following line of code −

127.0.0.1:8529@song\_collection&t; db.\_isSystem()

Output

false

It means we are not in the system database (as we have created and shifted to the song\_collection). The following screenshot will help you understand this.



To get a particular collection, say songs −

We will use the following line of code the get a particular collection.

127.0.0.1:8529@song\_collection> db.\_collection("songs")

Output

[ArangoCollection 4890, "songs" (type document, status loaded)]

The line of code returns a single collection.

Let us move to the essentials of the database operations with our subsequent chapters.