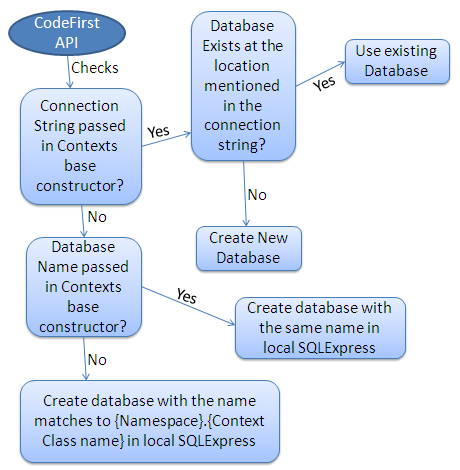
**Database Initialization:**

. Here, We will learn how Code first decides the database name and server while initializing a database.

The following figure shows a database initialization workflow, based on the parameter passed in the base constructor of context class, which is derived from DbContext:



As per the above figure, base constructor of the context class can have following parameter.

1. No Parameter
2. Database Name
3. Connection String Name

## No Parameter:

If you do not specify the parameter in the base constructor of the context class then it creates a database in your local SQLEXPRESS server with a name that matches your {Namespace}.{Context class name}. For example, Code First will create a database named *SchoolDataLayer.Context* for following context class:

namespace SchoolDataLayer

{

public class Context: DbContext

{

public Context(): base()

{

}

}

}

## Database Name:

You can also specify the database name as a parameter in a base constructor of the context class. If you specify database name parameter then Code First creates a database with the name you specified in base constructor in the local SQLEXPRESS database server. For example, Code First will create a database named *MySchoolDB* for the following context class.

namespace SchoolDataLayer

{

public class Context: DbContext

{

public Context(): base("MySchoolDB")

{

}

}

}

## ConnectionString Name:

You can also define connection string in app.config or web.config and specify connection string name starting with "name=" in the base constructor of the context class. Consider the following example where we pass *name=SchoolDBConnectionString* parameter in the base constructor.

namespace SchoolDataLayer

{

public class Context: DbContext

{

public Context() : base("name=SchoolDBConnectionString")

{

}

}

}

App.config:

<?xml version="1.0" encoding="utf-8" ?>

<configuration>

<connectionStrings>

<add name="SchoolDBConnectionString"

connectionString="Data Source=.;Initial Catalog=SchoolDB-ByConnectionString;Integrated Security=true"

providerName="System.Data.SqlClient"/>

</connectionStrings>

</configuration>

In the above Context class, we specify a connection string name as a parameter. Please note that connection string name should start with "name=" otherwise it will consider it as a database name. The database name in the connection string in app.config is *SchoolDB-ByConnectionString*. So Code First will create a new *SchoolDB-ByConnectionString* database or use existing *SchoolDB-ByConnectionString* database at local SQL Server. Make sure that you include providerName = "System.Data.SqlClient" in the connection string.

Thus, Code First use base constructor parameter to initialize a database.