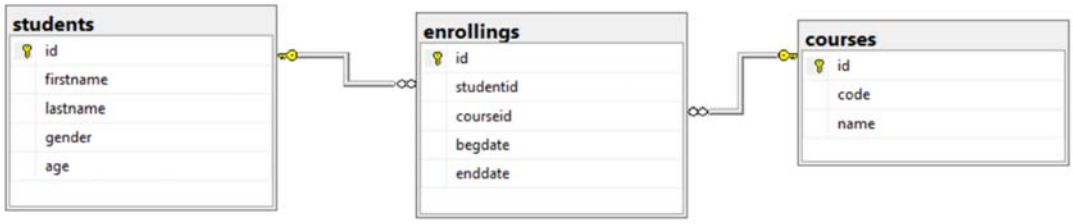
Database Access (1)

1- Introduction

- In .NET, accessing to data sources might be done via classes collected to the Micorsoft ADO.NET
- Each type of data source supports with its data provider
- Every data provider contains a set of classes which are not only implemented to a set of interfaces defined in the namespace System.Data, but also provider its own funcitnalities
- All of these classes including in groups of Connections, Commands, Parameters, Data Adapters, and Data Readers which are worked on data source
- Relational database, called **enrolldb**, will be studied including following table schema

  students(id:char(36), firstname:char(30), lastname:char(30), gender:int, age: byte)
  courses(id:char(36), code:char(20), name:char(30))
  enrollings(id:char(36), studentid:char(36), courseid:char(36), begdate:datetime, enddate:datetime)



- Scripts to create the table named **students**

```
create table students
(
	id varchar(36) primary key,
	firstname nvarchar(30),
	lastname nvarchar(30),
	gender int null,
	age tinyint
)
```

- Scripts to create the table named **courses**

```
Create table courses
(
	id varchar(36) primary key,
	code varchar(20) unique not null,
	name nvarchar(30) not null
)
```

- Scripts to create the table named **enrollings**

```
create table enrollings
(
	id varchar(36) primary key,
	studentid varchar(36),
	courseid varchar(36),
	begdate datetime,
	enddate datetime,
	constraint FK_Enrollings_Students foreign key (studentid) references students(id) on
                                                    update cascade on delete no action,
	constraint FK_Enrollings_Courses foreign key (courseid) references courses(id) on
                                                    update cascade on delete no action,
)
```

2- Connecting to Database

- A database is connected to for purpose
  A. retrieving data/information from the database
  B. submitting changes to the database
  C. adding new data to the database
  D. deleting existing data in the database
- Connecting to a database is done vis connection string varied based on type of database
- Some connection strings used to connect to SQL server's database

A. Connection string for SQL server authentication:

"[provider=sqloledb;]data source=<computername>/<servername>;[initial catalog=<dbname>;]
user id=<username>;password=<password>;encrypt=<true/false>"
Eg. ConnStr1 = "data source=mylaptop/sqlexpress;initial catalog=enrolldb; user id=sa;
password=123456;encrypt=false"

"[provider=sqloledb;] server=<computername>/<servername>;[database=<dbname>;]
uid=<username>;pwd=<password>;encrypt=<true/false>"
Eg. ConnStr2 = "server=localhost;database=enrolldb; uid=sa; pwd=123456;encrypt=false"

B. Connection string for Windows Authentication

"[provider=sqloledb;]data source=<computername>/<servername>;[initial catalog=<dbname>;]
trusted_connection=true;encrypt=<true/false>"
Eg. ConnStr3 = "data source=mylaptop/sqlexpress;initial catalog=enrolldb;
trusted_connection=true;encrypt=false"

"[provider=sqloledb;]server=<computername>/<servername>;[database=<dbname>;]
trusted_connection=true;encrypt=<true/false>"
Eg. ConnStr4 = "server=.;database=enrolldb; trusted_connection=true;encrypt=false"

- Instance method Open() of the class System.Data.SqlClient.SqlConnection is used to open connection to SQL server's database. The class SqlConnection is shipped in the nuget package "Microsoft.Data.SqlClient" with lastest stable version "5.2.0".

- Example: Connecting to database defined by the connection string, connStr2

```
var conn = new SqlConnection();
conn.ConnectionString = connStr2;
conn.Open();
```

- Example: Connecting to database defined by the connection string, connStr4

```
var conn= new SqlConnection(connStr4);
conn.Open();
```

3- Creating Database Schema

- For a connection to SQL server with a default database
  A. A new database might be created via command string/script “create database <databasename>”
  B. Changing the current database, use the instance method ChangeDatabase() of the class class System.Data.SqlClient.SqlConnection
  C. Creating a table in connecting database, use the command string/script “create table ….”
- A command string/script will be executed to make effection in connecting database via the instance method ExecuteNonQuery() of the class System.Data.SqlClient.Command.
- Examples

A. Opening a connection to server

```
conn = new SqlConnection();
conn.ConnectionString = “server=localhost;trusted_connection=true;encrypt=false”;
conn.Open();
```

B. Creating a database named “enrolldb”

```
cmd = new SqlCommand();
cmd.Connection = conn;
cmd.CommandText = “create database enrolldb”;
cmd.CommandType = System.Data.CommandType.Text;
cmd.ExecuteNonQuery();
```

C. Changing the current databse to the database named “enroldb”

```
conn.ChangeDatabase(“enrolldb”);
```

D. Creating a table name “courses” in the connecting database

```
cmd =new SqlCommand();
cmd.Connection = conn;
cmd.CommandText = “create table {0}(id varchar(36) primary key,code varchar(20) unique not
                                    null,name nvarchar(30) not null)”;
cmd.CommandType = System.Data.CommandType.Text;
cmd.ExecuteNonQuery();
```