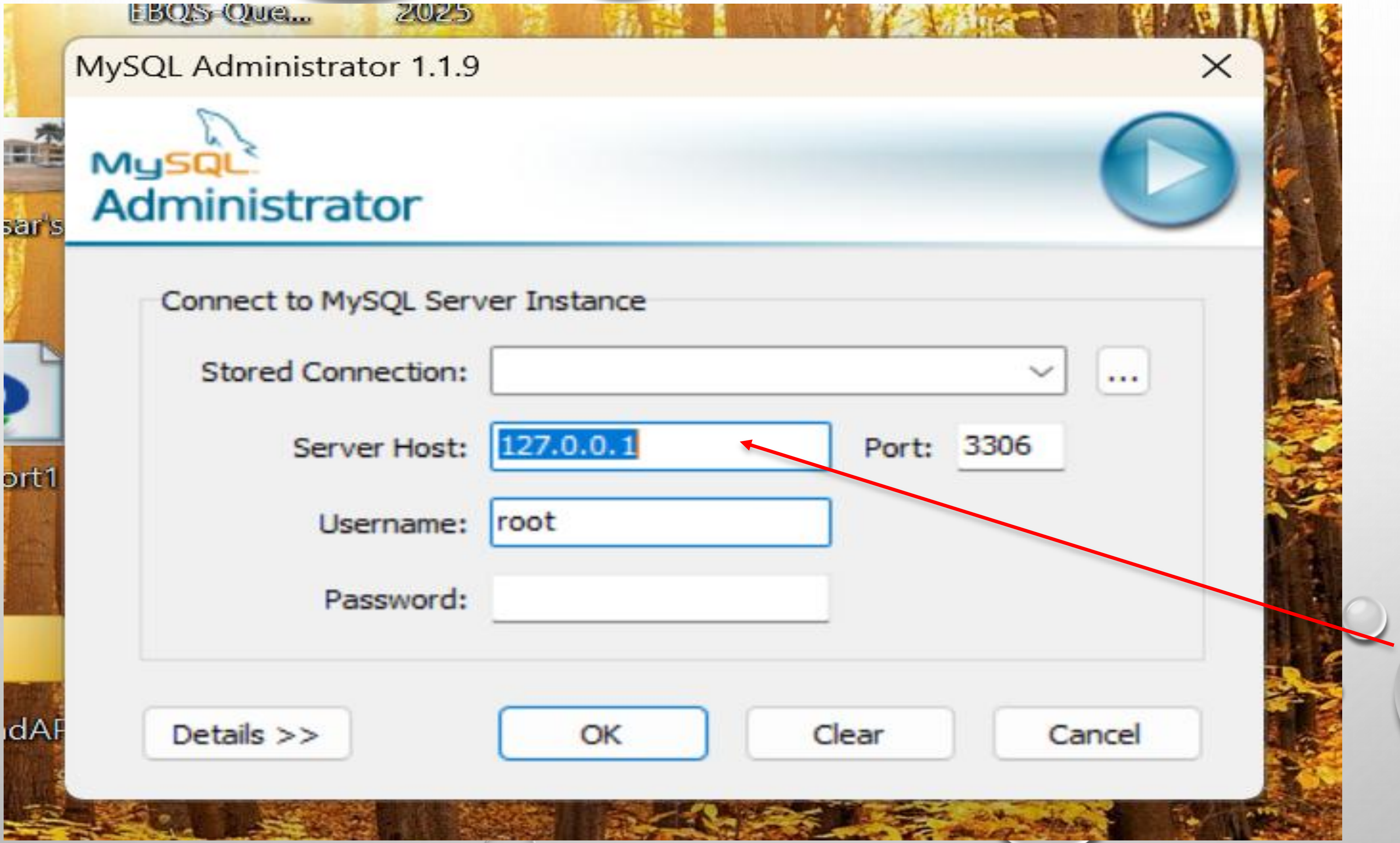
WORKING WITH MYSQL DATABASE AND VISUAL STUDIO

BUSINESS APPLICATION PROGRAMMING



MySQL Administrator 1.1.9

Connect to MySQL Server Instance

Stored Connection:

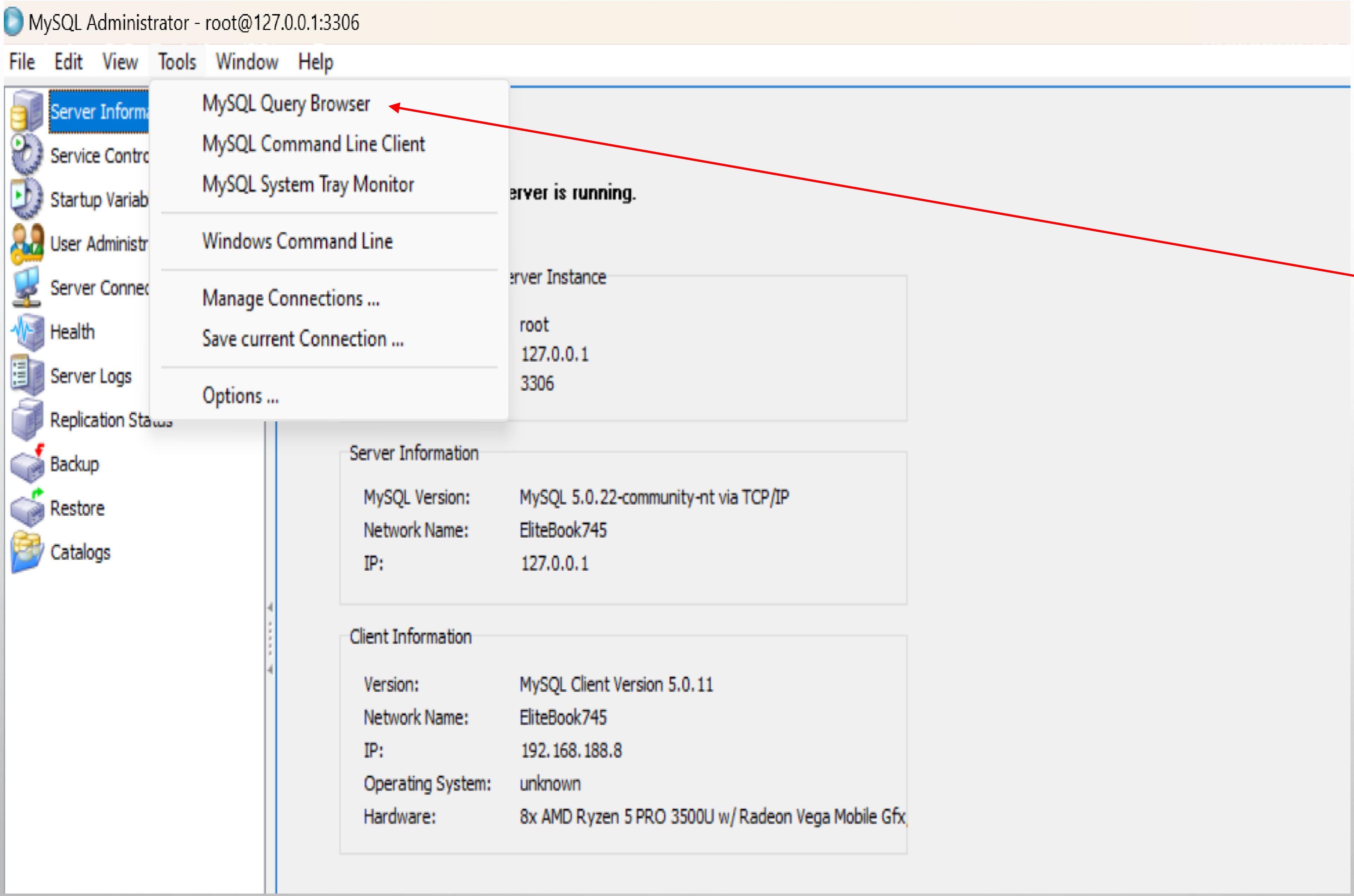
Server Host: 127.0.0.1

Port: 3306

Username: root

Password:

Server host : localhost
or its IP Address
127.0.0.1
User name: root



MySQL Administrator - root@127.0.0.1:3306

erver is running.

erver Instance

root
127.0.0.1
3306

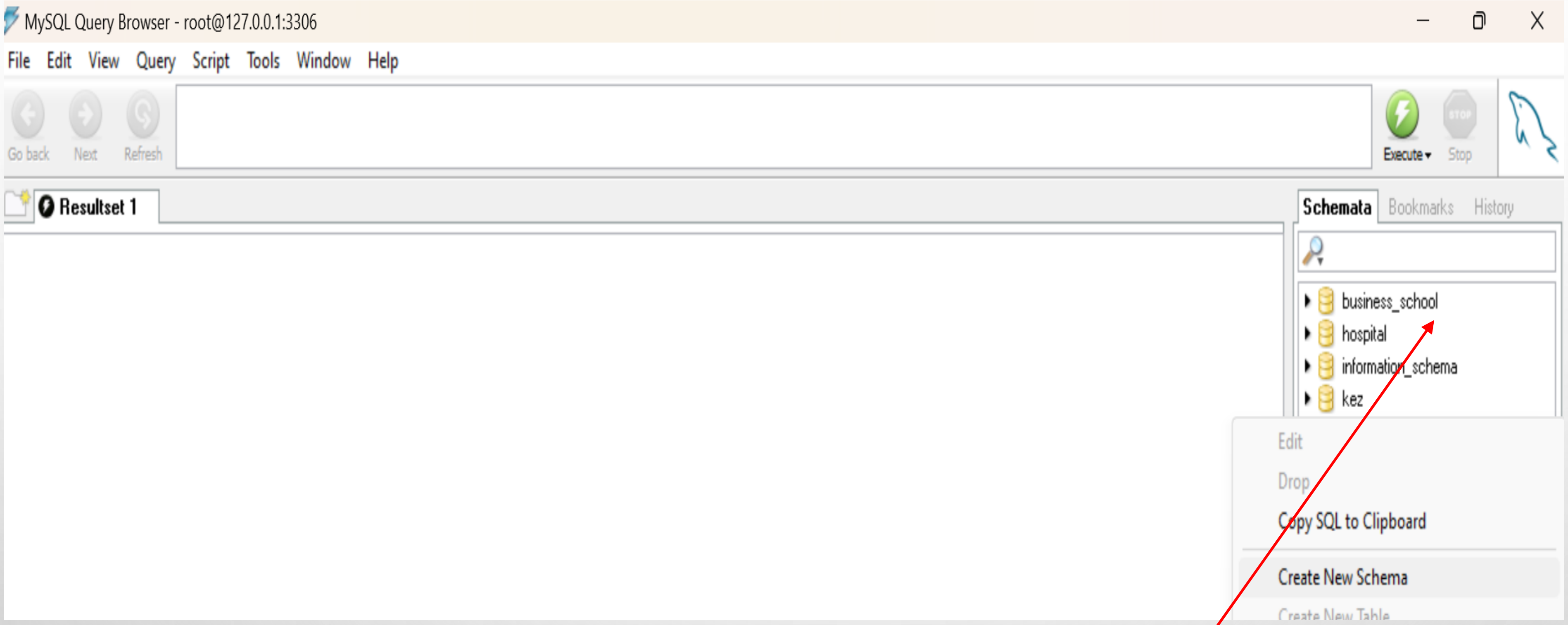
Server Information

MySQL Version: MySQL 5.0.22-community-nt via TCP/IP
Network Name: EliteBook745
IP: 127.0.0.1

Client Information

Version: MySQL Client Version 5.0.11
Network Name: EliteBook745
IP: 192.168.188.8
Operating System: unknown
Hardware: 8x AMD Ryzen 5 PRO 3500U w/ Radeon Vega Mobile Gfx

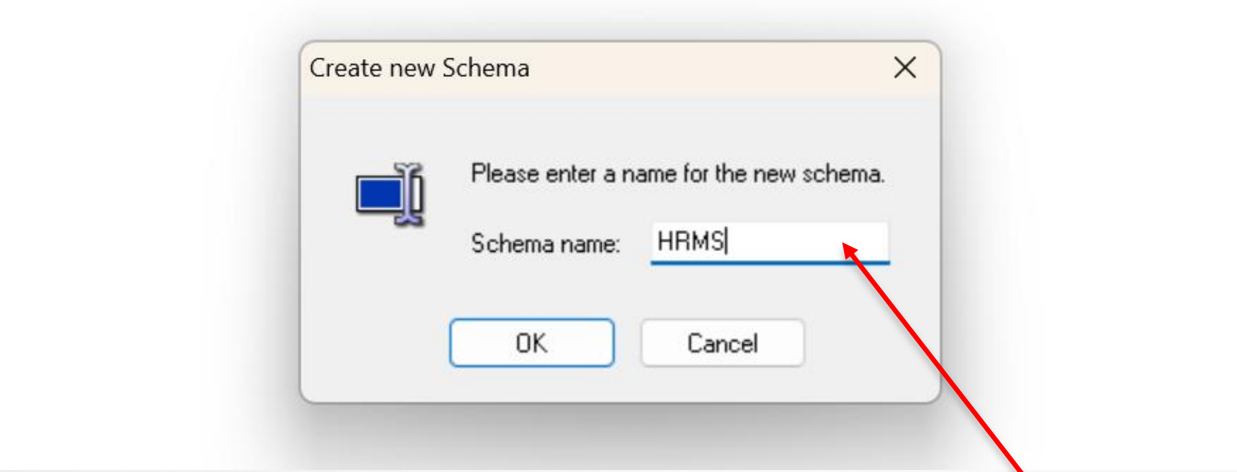
Click on MySQL Query Browser



In MySQL, **schemata** is the plural of **schema**, which refers to the logical structure of a database. A schema defines how data is organized, including tables, views, indexes, stored procedures, and relationships.

This part list all the schemata (databases) created.

To create a new schema right click under schemata and click on create new schema

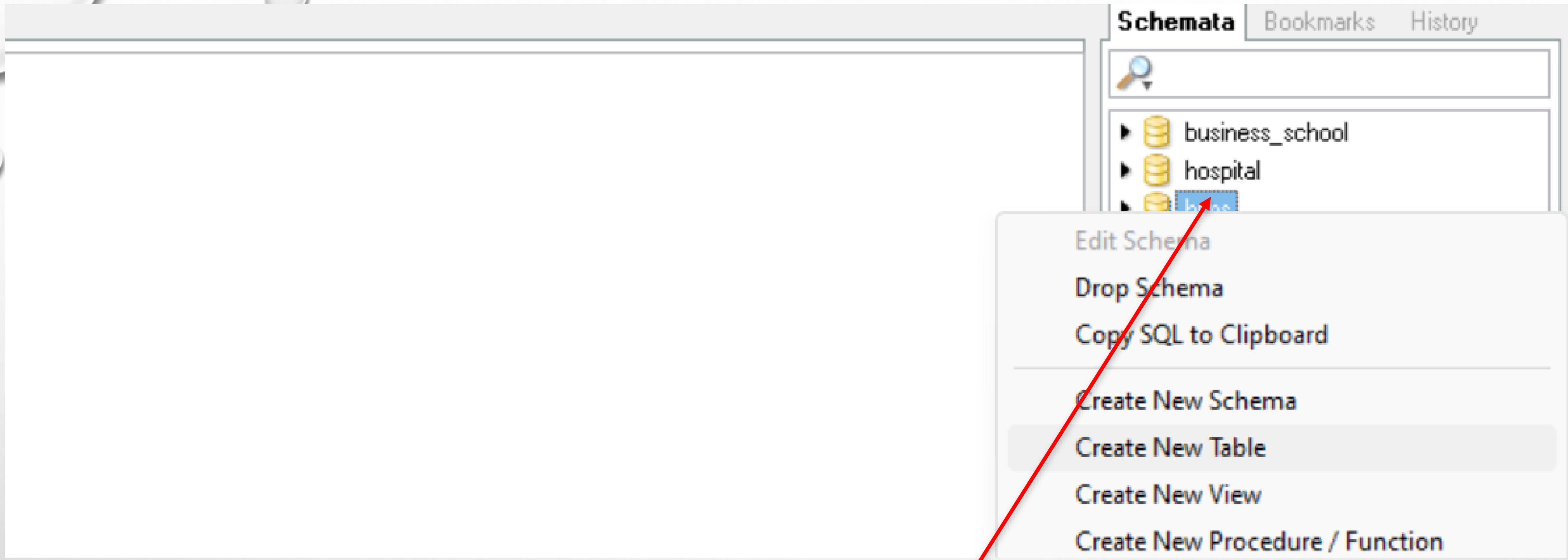


Create new Schema

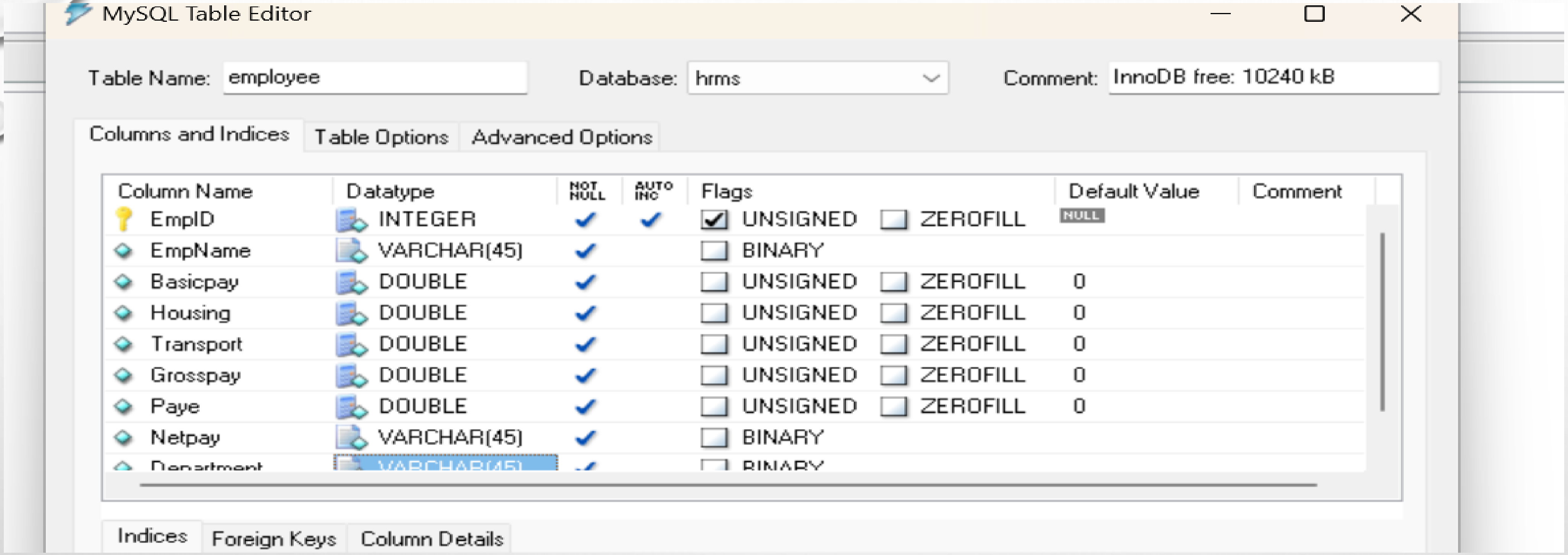
Please enter a name for the new schema.

Schema name: HRMS

Then give a new schema a name for example Human resource management system (HRMS) and then after click Ok.



Since the database organises data inform of tables, you have to right click on the new Schema created and then click on Create New Table



MySQL Table Editor

Table Name: employee

Database: hrms

Comment: InnoDB free: 10240 kB

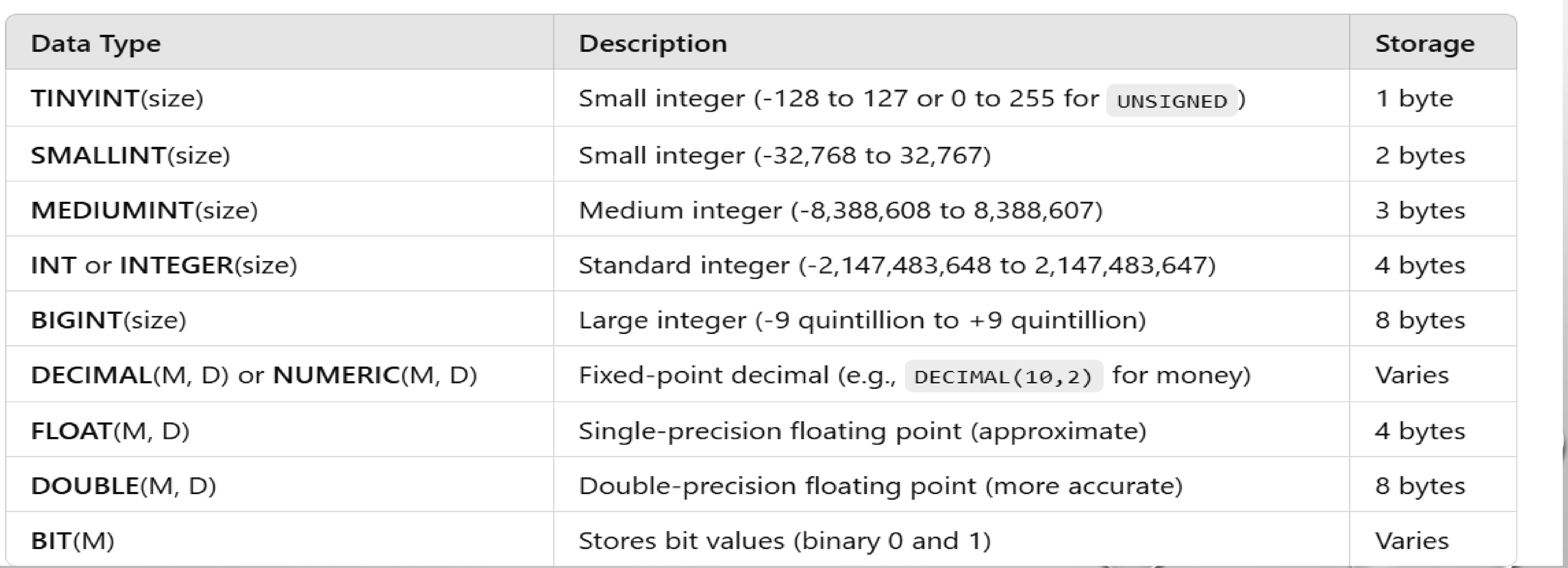
Column Name	Datatype	NOT NULL	AUTO INC	Flags		Default Value	Comment
EmpID	INTEGER	✓	✓	☑ UNSIGNED	☐ ZEROFILL	NULL	
EmpName	VARCHAR(45)	✓		☐ BINARY			
Basicpay	DOUBLE	✓		☐ UNSIGNED	☐ ZEROFILL	0	
Housing	DOUBLE	✓		☐ UNSIGNED	☐ ZEROFILL	0	
Transport	DOUBLE	✓		☐ UNSIGNED	☐ ZEROFILL	0	
Grosspay	DOUBLE	✓		☐ UNSIGNED	☐ ZEROFILL	0	
Paye	DOUBLE	✓		☐ UNSIGNED	☐ ZEROFILL	0	
Netpay	VARCHAR(45)	✓		☐ BINARY			
Department	VARCHAR(45)	✓		☐ BINARY			

Give the table a name proceed to specifying the column names, Datatypes among others. In MySQL, NOT NULL and AUTO_INCREMENT are constraints used to define specific behaviours for table columns.1. The NOT NULL constraint ensures that a column cannot have NULL values. It forces every row to have a valid value in that column. 2. The AUTO_INCREMENT attribute automatically generates a unique number for each new row. It is typically used for primary keys.

After all these click on Apply Changes then Execute then after click on Close button

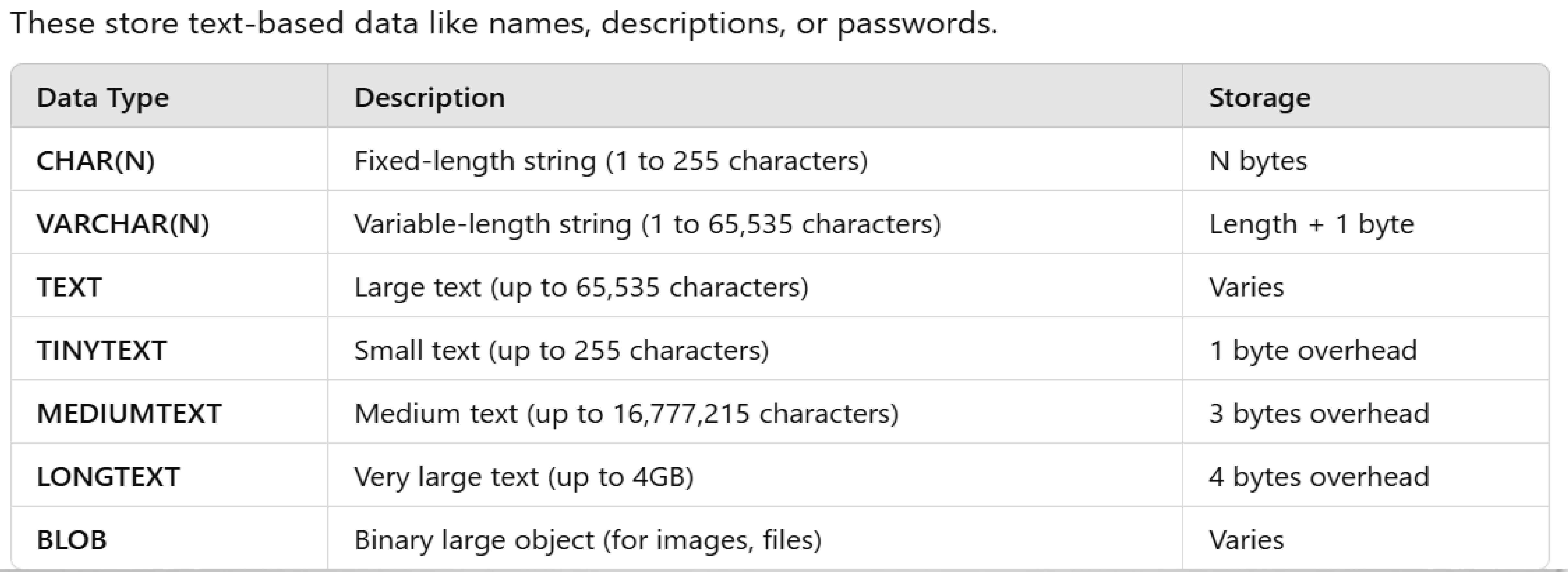
DATATYPES IN MYSQL

- In **MYSQL,** Data Types Define The Kind of Values that can be Stored in a Table Column. They are Categorized into **Numeric, String (Text), Date/Time, And Spatial** Data Types.
- 1. NUMERIC DATA TYPES



Data Type	Description	Storage
TINYINT(size)	Small integer (-128 to 127 or 0 to 255 for `UNSIGNED`)	1 byte
SMALLINT(size)	Small integer (-32,768 to 32,767)	2 bytes
MEDIUMINT(size)	Medium integer (-8,388,608 to 8,388,607)	3 bytes
INT or **INTEGER**(size)	Standard integer (-2,147,483,648 to 2,147,483,647)	4 bytes
BIGINT(size)	Large integer (-9 quintillion to +9 quintillion)	8 bytes
DECIMAL(M, D) or **NUMERIC**(M, D)	Fixed-point decimal (e.g., `DECIMAL(10,2)` for money)	Varies
FLOAT(M, D)	Single-precision floating point (approximate)	4 bytes
DOUBLE(M, D)	Double-precision floating point (more accurate)	8 bytes
BIT(M)	Stores bit values (binary 0 and 1)	Varies

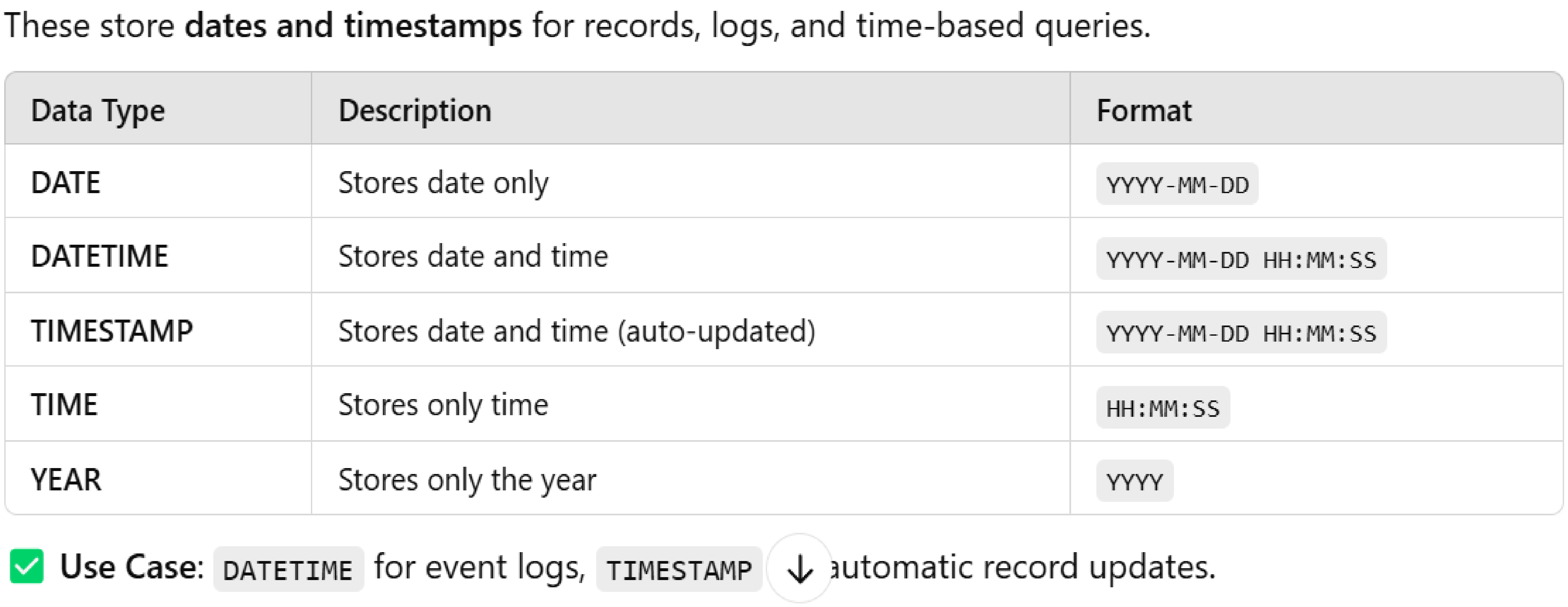
2. STRING (CHARACTER) DATA TYPES



These store text-based data like names, descriptions, or passwords.

Data Type	Description	Storage
CHAR(N)	Fixed-length string (1 to 255 characters)	N bytes
VARCHAR(N)	Variable-length string (1 to 65,535 characters)	Length + 1 byte
TEXT	Large text (up to 65,535 characters)	Varies
TINYTEXT	Small text (up to 255 characters)	1 byte overhead
MEDIUMTEXT	Medium text (up to 16,777,215 characters)	3 bytes overhead
LONGTEXT	Very large text (up to 4GB)	4 bytes overhead
BLOB	Binary large object (for images, files)	Varies

3. DATE AND TIME DATA TYPES

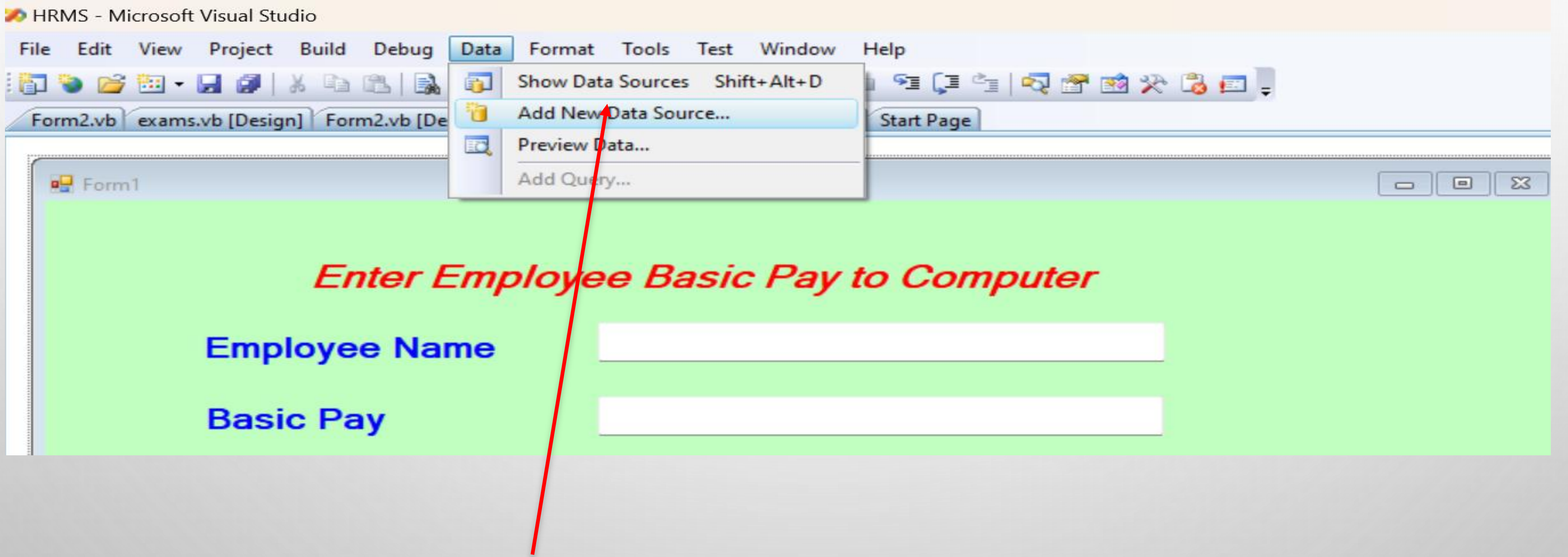


These store **dates and timestamps** for records, logs, and time-based queries.

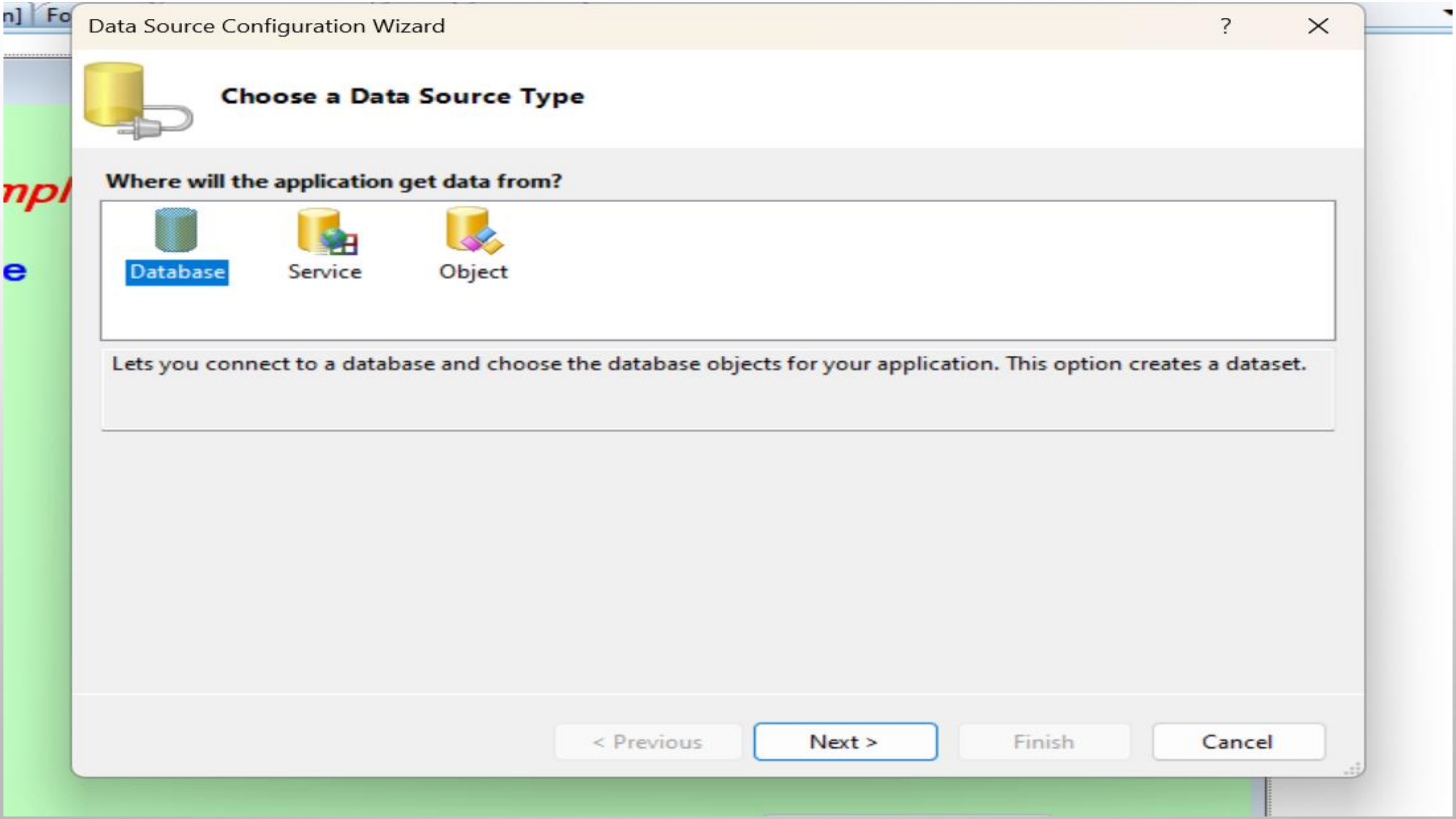
Data Type	Description	Format
DATE	Stores date only	`YYYY-MM-DD`
DATETIME	Stores date and time	`YYYY-MM-DD HH:MM:SS`
TIMESTAMP	Stores date and time (auto-updated)	`YYYY-MM-DD HH:MM:SS`
TIME	Stores only time	`HH:MM:SS`
YEAR	Stores only the year	`YYYY`

✅ **Use Case:** `DATETIME` for event logs, `TIMESTAMP` ↓ automatic record updates.

CONNECTING VB APPLICATION TO MYSQL DATABASE



Click on Data tab then Add new Data Sources



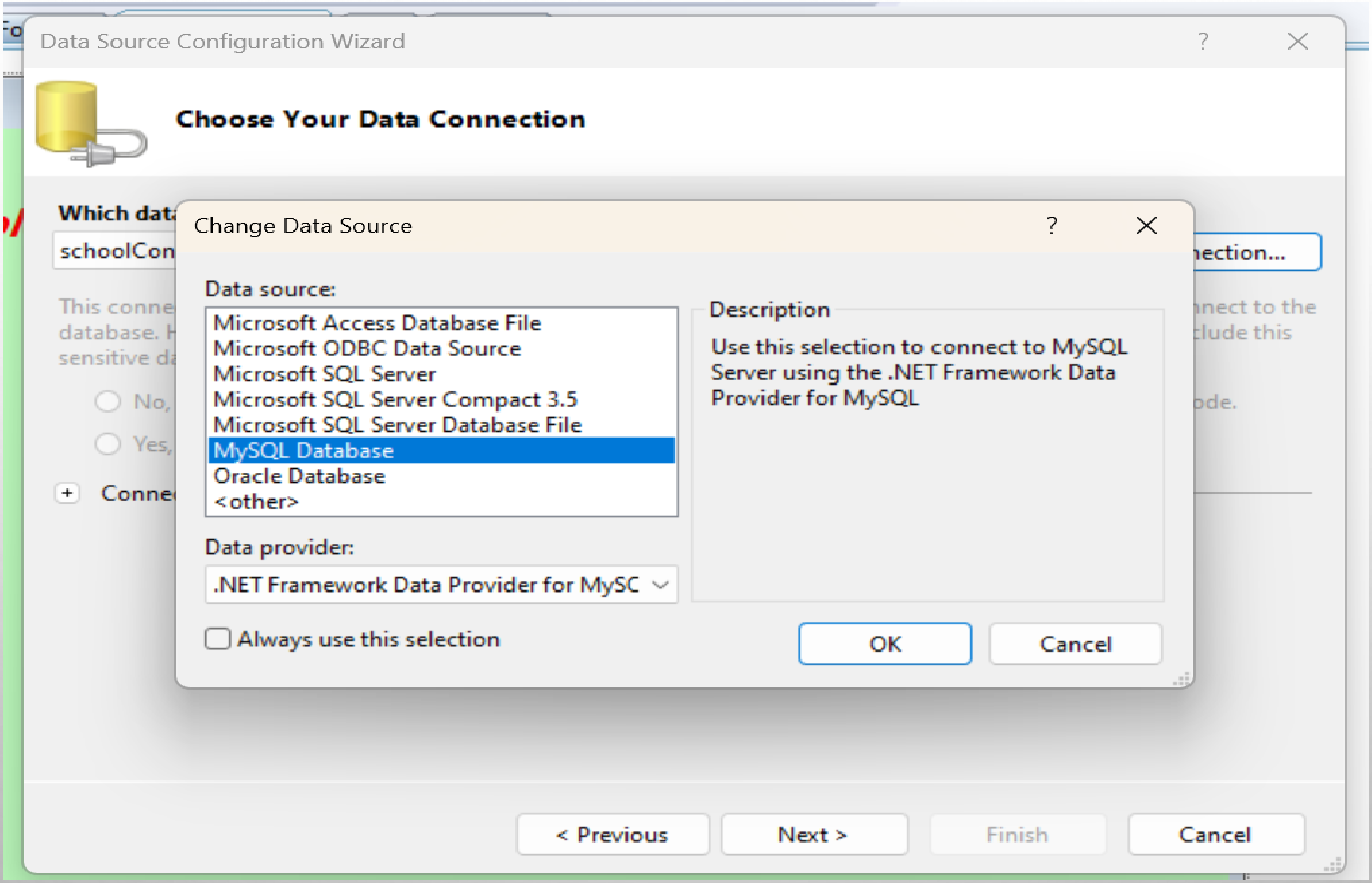
Data Source Configuration Wizard

Choose a Data Source Type

Where will the application get data from?

Database

Lets you connect to a database and choose the database objects for your application. This option creates a dataset.



Data Source Configuration Wizard

Choose Your Data Connection

Which data

schoolCon

This conne
database. H
sensitive da

○ No,

○ Yes,

+ Conne

nect to the
clude this

ode.

Change Data Source

Data source:

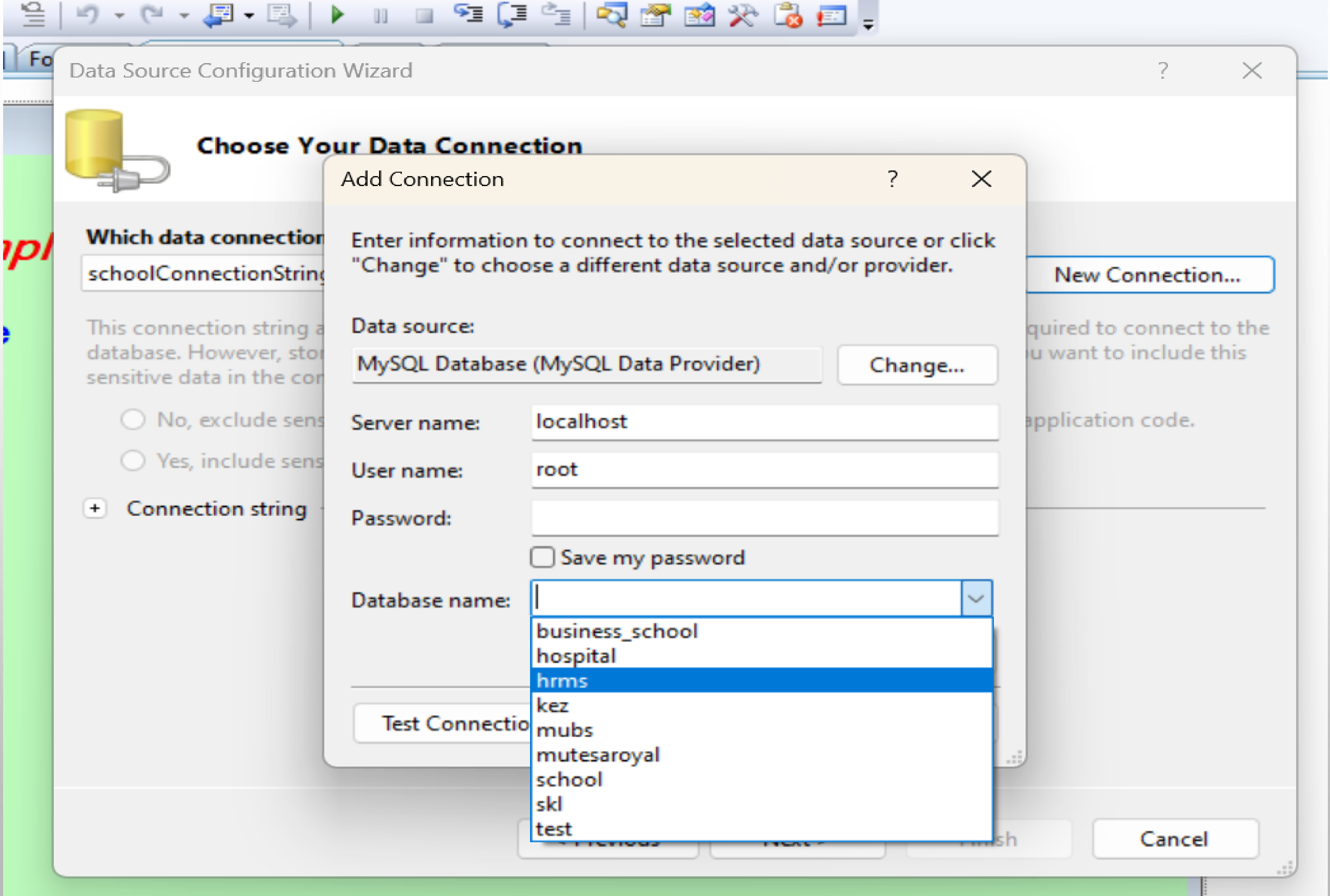
- Microsoft Access Database File
- Microsoft ODBC Data Source
- Microsoft SQL Server
- Microsoft SQL Server Compact 3.5
- Microsoft SQL Server Database File
- MySQL Database
- Oracle Database
- <other>

Description

Use this selection to connect to MySQL Server using the .NET Framework Data Provider for MySQL

Data provider: .NET Framework Data Provider for MySC

□ Always use this selection



Choose Your Data Connection

Which data connection

schoolConnectionString

This connection string a ... quired to connect to the database. However, stor ... u want to include this sensitive data in the con ... application code.

Add Connection

Enter information to connect to the selected data source or click "Change" to choose a different data source and/or provider.

Data source:

MySQL Database (MySQL Data Provider)

Server name: localhost

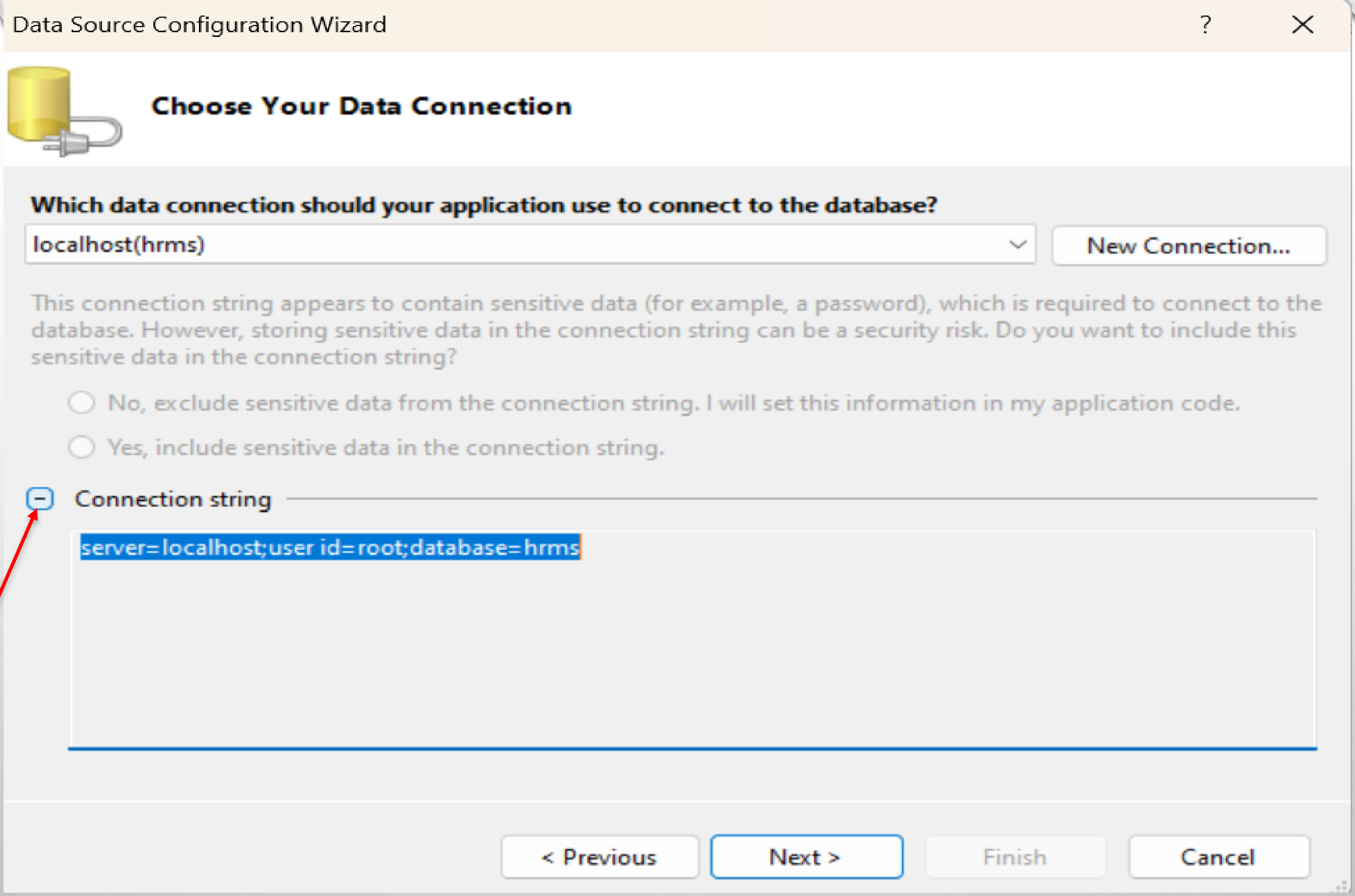
User name: root

Password:

□ Save my password

Database name:

- business_school
- hospital
- hrms
- kez
- mubs
- mutesaroyal
- school
- skl
- test



Data Source Configuration Wizard

Choose Your Data Connection

Which data connection should your application use to connect to the database?

localhost(hrms)

This connection string appears to contain sensitive data (for example, a password), which is required to connect to the database. However, storing sensitive data in the connection string can be a security risk. Do you want to include this sensitive data in the connection string?

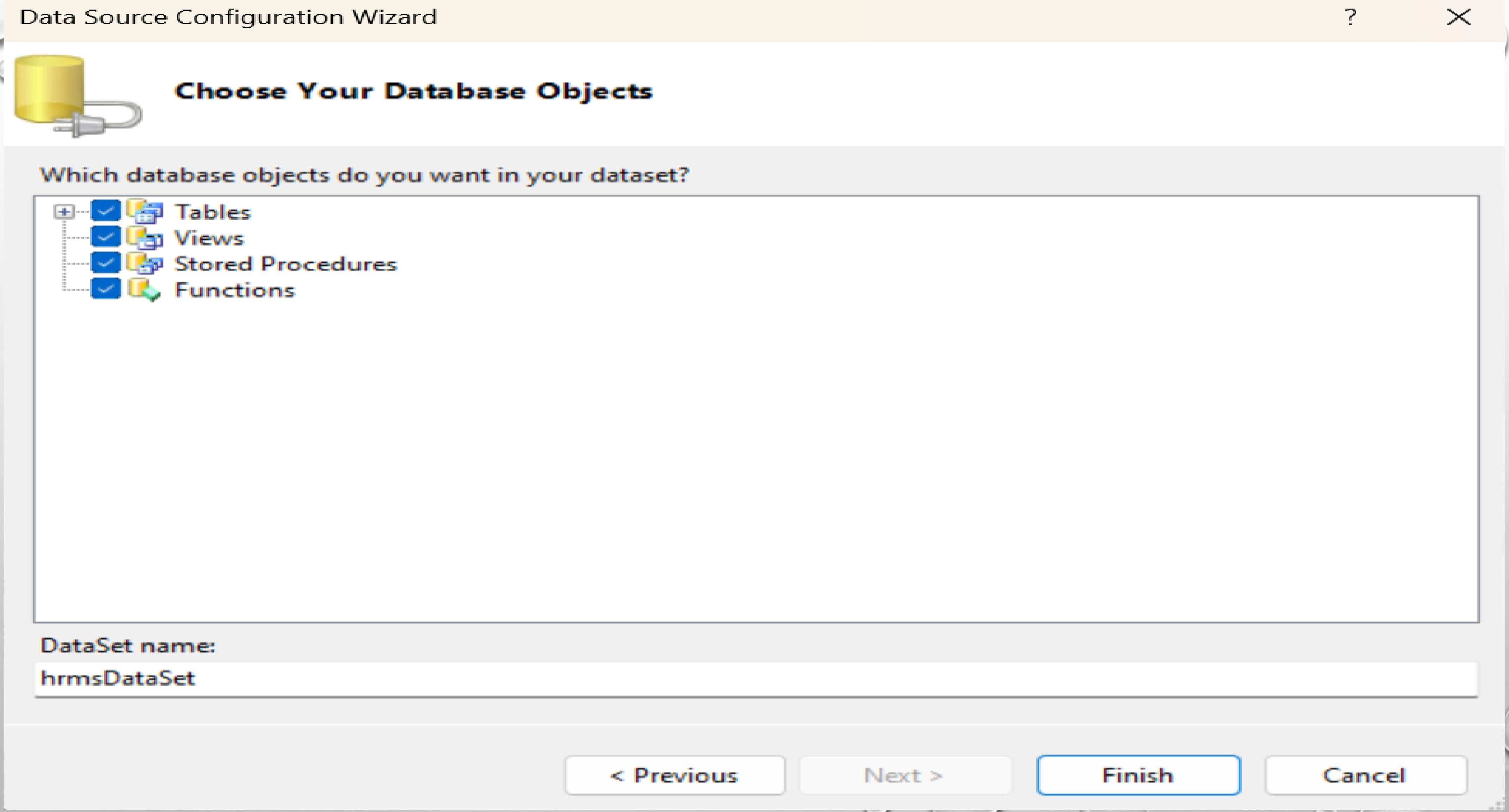
○ No, exclude sensitive data from the connection string. I will set this information in my application code.

○ Yes, include sensitive data in the connection string.

Connection string

```
server=localhost;user id=root;database=hrms
```

Click on the + against connection string then after copy the connection string text



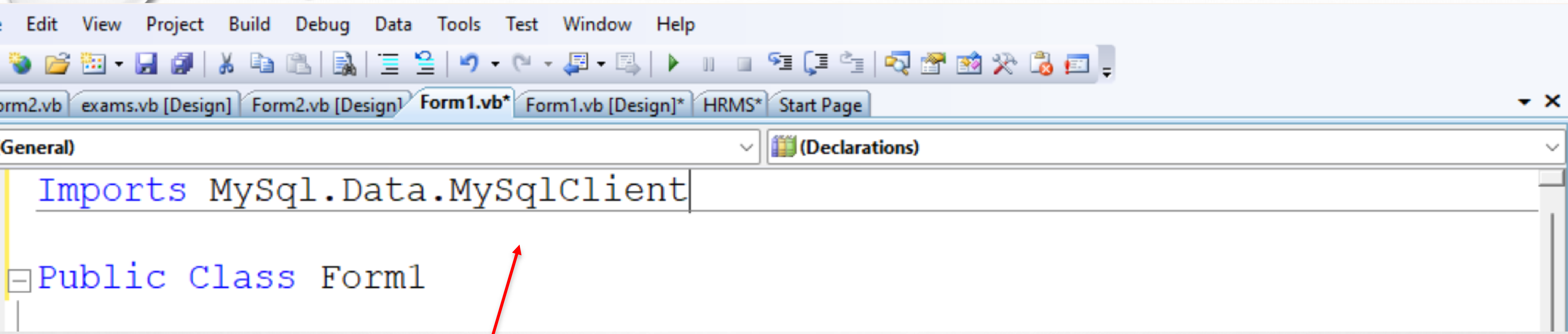
Data Source Configuration Wizard

Choose Your Database Objects

Which database objects do you want in your dataset?

- ☑ Tables
- ☑ Views
- ☑ Stored Procedures
- ☑ Functions

DataSet name: hrmsDataSet



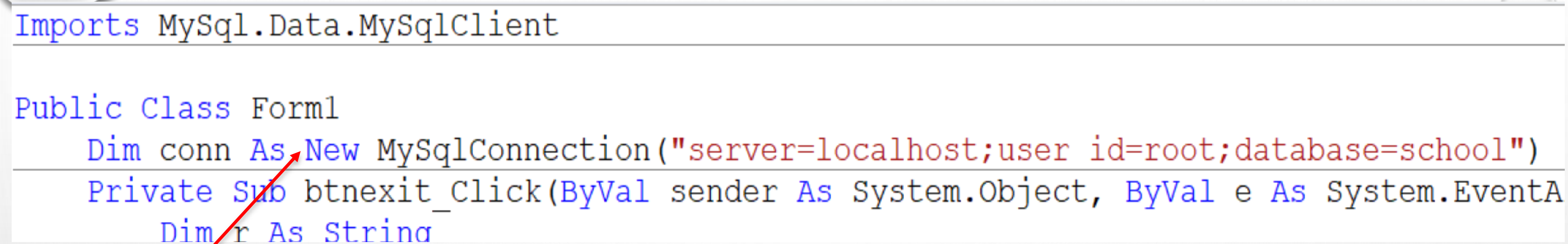
```
Imports MySql.Data.MySqlClient

Public Class Form1
```

This means that the program is importing the MySqlClient library, which provides classes for connecting to and interacting with a MySQL database.

Imports Statement – This allows you to use classes from a namespace without specifying the full path every time.

MySql.Data.MySqlClient Namespace – It is part of the MySQL Connector/NET, a library that enables .NET applications to interact with MySQL databases.



```
Imports MySql.Data.MySqlClient

Public Class Form1
    Dim conn As New MySqlConnection("server=localhost;user id=root;database=school")
    Private Sub btnexit_Click(ByVal sender As System.Object, ByVal e As System.EventA
        Dim r As String
```

Creates and initializes a new connection to a MySQL database using the MySqlConnection class.

Common Classes in MySql.Data.MySqlClient:

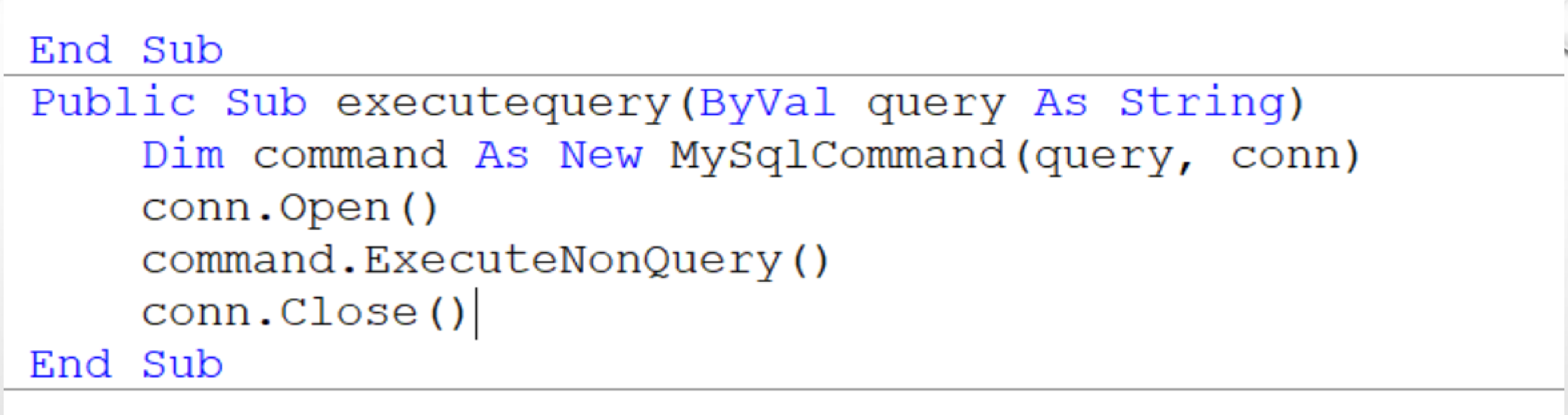
MySqlConnection – Used to connect to a MySQL database.

MySqlCommand – Used to execute SQL queries (SELECT, INSERT, etc.).

MySqlDataReader – Reads query results row by row.

MySqlDataAdapter – Fills datasets with query results.

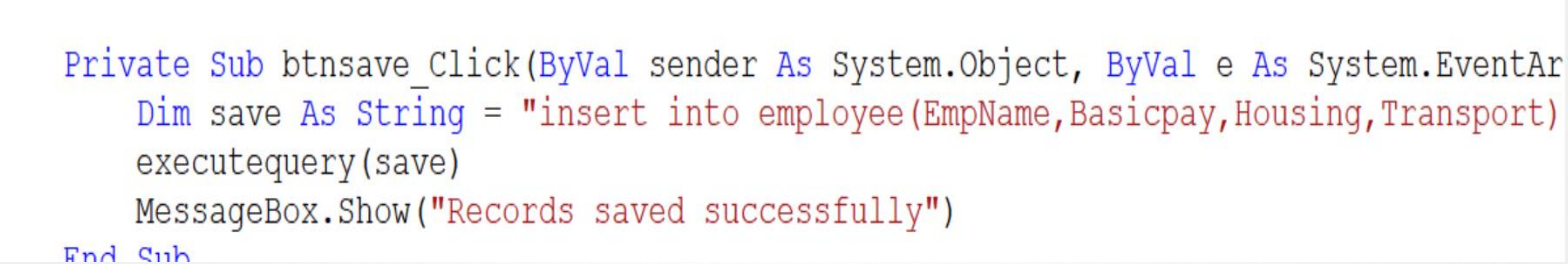
This executequery subroutine in **Visual Basic (VB)** is a method that executes an SQL query using a MySQL database connection.



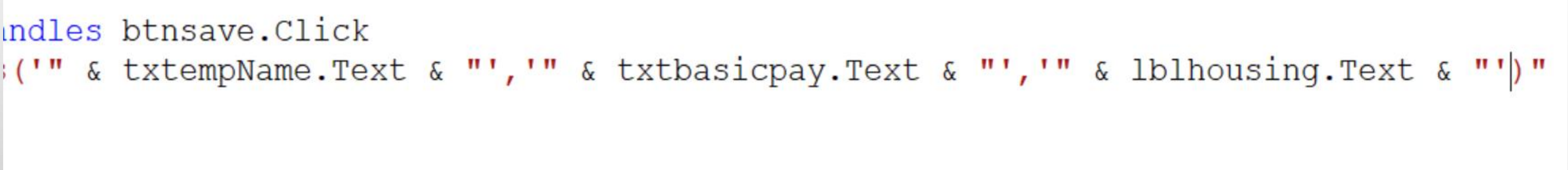
```
End Sub
Public Sub executequery(ByVal query As String)
    Dim command As New MySqlCommand(query, conn)
    conn.Open()
    command.ExecuteNonQuery()
    conn.Close()
End Sub
```

Create a Sub function called “executequery” which is called for to execute the query.
The executequery subroutine in Visual Basic (VB) is a method that executes an SQL query using a MySQL database connection.

ExecuteNonQuery() is used for queries that do not return data, such as: INSERT, UPDATE, DELETE, CREATE, and DROP

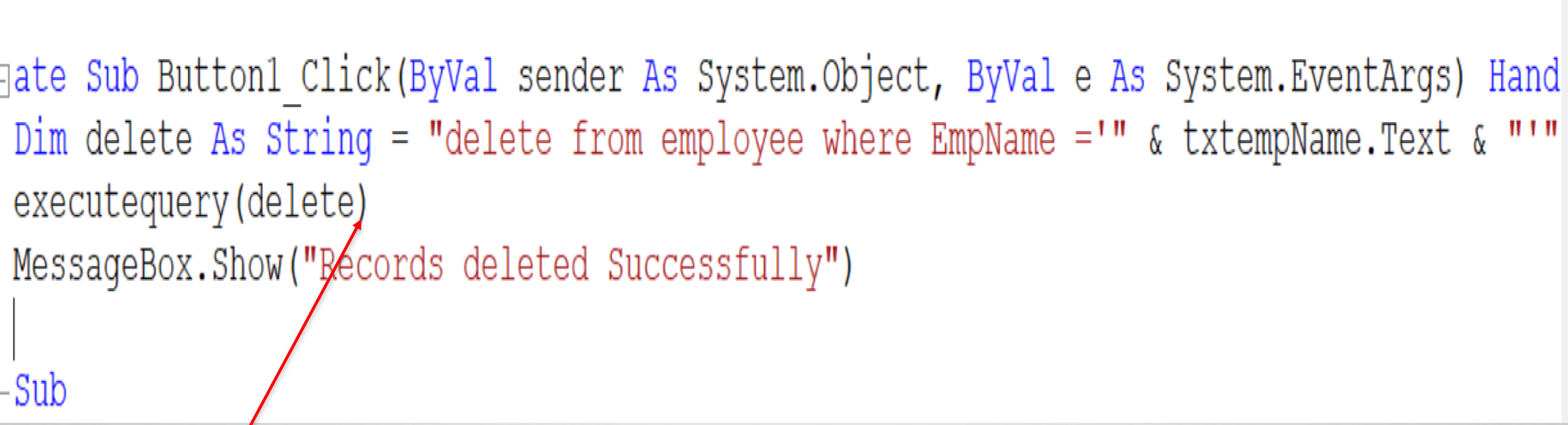


```
Private Sub btnsave_Click(ByVal sender As System.Object, ByVal e As System.EventAr
    Dim save As String = "insert into employee(EmpName,Basicpay,Housing,Transport)
    executequery(save)
    MessageBox.Show("Records saved successfully")
End Sub
```



```
ndles btnsave.Click
('" & txtempName.Text & "','" & txtbasicpay.Text & "','" & lblhousing.Text & "')"
```

```
 Dim save As String = "insert into employee(EmpName,Basicpay,Housing,Transport)values('" & txtempName.Text
& "','" & txtbasicpay.Text & "','" & lblhousing.Text & "','" & lbltransport.Text & "')"
        executequery(save)
        MessageBox.Show("Records saved successfully")
```



```
ate Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Hand
Dim delete As String = "delete from employee where EmpName ='" & txtempName.Text & "'"
executequery(delete)
MessageBox.Show("Records deleted Successfully")

Sub
```

Executequery subfunction is also called here to execute the delete statement, we can call it in other statements like update, insert among others