



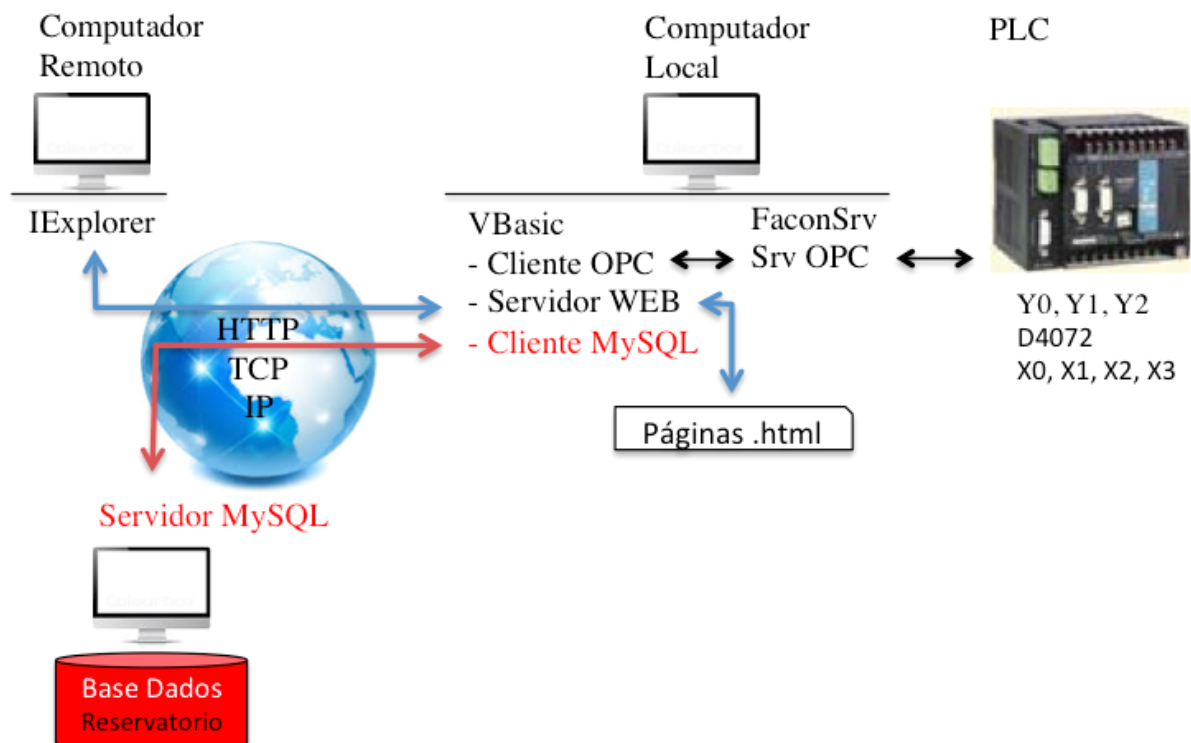
Universidade de Aveiro
Departamento de
Engenharia Mecânica

Informática Industrial 2022/2023

TRABALHO PRÁTICO Nº 11

Bases de Dados (MySQL)

Neste trabalho, vamos continuar a monitorizar e a controlar remotamente o funcionamento do PLC “Reservatório”, mas os dados transferidos, de e para o PLC, devem ser armazenados numa base de dados MySQL.



Conforme a figura ilustra, além do PLC que controla o processo industrial, dispomos de três computadores. A partir do computador remoto o utilizador pode usar um BrowserWEB para monitorizar e controlar o PLC/Reservatório (SCADA). O computador local possui duas aplicações: a aplicação desenvolvida em VBasic e o FaconSrv. Utilizando o FaconSrv como servidor OPC, a aplicação VB pode ler e escrever nos Itens do PLC como no trabalho nº 6. Utilizando objetos do tipo TcpClient, TcpListener, NetworkStream, e StreamReader, a aplicação VB pode atuar como um servidor WEB, respondendo aos pedidos HTTP do Browser, com o conteúdo dos documentos HTML disponíveis no computador local (como no trabalho nº10).

Neste trabalho em concreto, a aplicação VB utiliza objetos do tipo MySqlConnection, MySqlCommand e MySQLDataReader para aceder ao computador/Servidor MySQL, ler e escrever na base de dados “reservatorio”.

1 INTRODUÇÃO

Recorda-se que a partir da aplicação remota (Browser), usada no trabalho anterior, era possível controlar as saídas do PLC: Y0 (Motor), Y1 (EV_in) e Y2 (EV_out). Era também possível monitorizar o estado das entradas digitais: X0, X1, X2, X3 (sensores de nível).

Isso era conseguido através da aplicação local, desenvolvida em VBasic:

- Que recebia pela internet as ordens enviadas pelo “Browser” (mensagens HTTP/TCP/IP), e que através do Faconsrv controlava as saídas do PLC (Y0, Y1 e Y2).
- Que também através do Faconsrv, lia o estado das saídas e das entradas digitais do PLC (Y0, Y1, Y2, X0, X1, X2, X3) e enviava essa informação para o “Browser”.

Saídas digitais do PLC:

- Y0 - Controlo do motor “Y0_Motor”
- Y1 - Controlo da electroválvula de entrada de água no reservatório “Y1_EV_In”.
- Y2 - Controlo da electroválvula de saída de água do reservatório “Y2_EV_Out”.

Entradas digitais do PLC:

Quatro entradas digitais do PLC estão ligadas a 4 sensores de nível de água, que estão acoplados ao reservatório.

X0 - sensor “X0_AlarmEmpty”,

X1 – sensor “X1_Empty”,

X2 – sensor “X2_Full” e

X3 - sensor “X3_AlarmMax”.

Assume-se que os sensores ficam ativos quando detetam água fazendo com que o PLC, na entrada digital respetiva, receba 24 V.

2 INTRODUÇÃO MYSQL

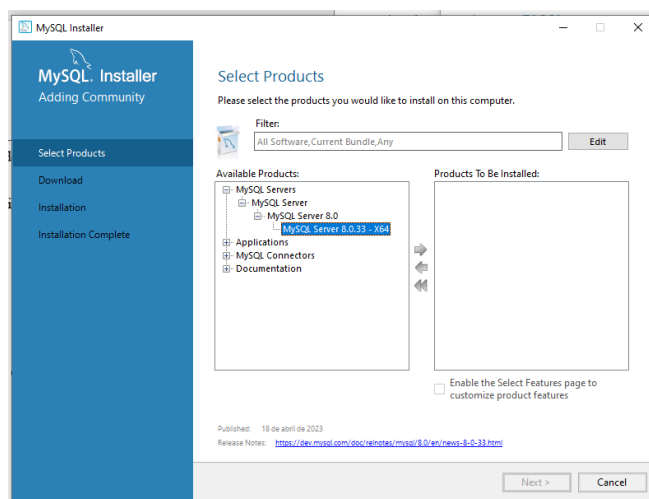
Nesta secção vamos introduzir o MySQL Workbench como ferramenta de manipulação de bases de dados MySQL.

Pode descarregar o instalador do MySQL Server e do Workbench diretamente do e-learning: *mysql-installer-community-8.0.33.0.msi*.

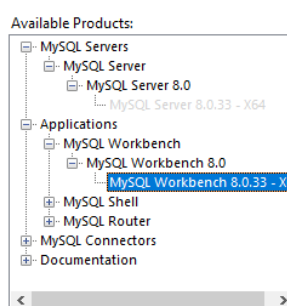
Use o instalador para instalar **apenas** os seguintes pacotes:

- MySQL Server
 - o servidor da base de dados;
- MySQL Workbench
 - utilitário para gerir bases de dados MySQL;
- Connector.NET
 - biblioteca .NET para ser possível comunicar com a base de dados com o código VB;

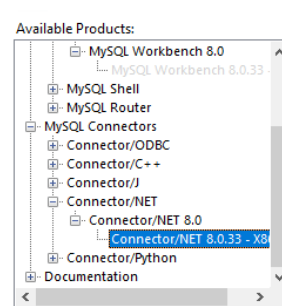
Servidor



Workbench



Connector



2.1 CRIAR A BASE DE DADOS

Use o Workbench para criar um *schema* com o nome *alunos*. Vamos usar esta base de dados para apenas para aprender os comandos essenciais de SQL.

Nesse *schema* crie uma tabela com o nome *dados_alunos*.

A tabela deve conter as seguintes colunas:

- id, do tipo INT como chave primária auto incrementada
- nome, do tipo VARCHAR(255)
- numero_mecanografico, do tipo INT
- telefone, do tipo INT

2.2 INSERIR INFORMAÇÃO NA BASE DE DADOS

A introdução de informação na base de dados SQL é efetuada com recurso ao comando *insert*.

Por exemplo a introdução de uma linha na nossa tabela é efetuada com a seguinte instrução:

```
insert into dados_alunos (nome, numero_mecanografico, telefone) values ('João Silva', 98451, 965244664);
```

Não é necessário inserir o valor numa coluna marcada como auto incremental.

A inserção pode ser parcial como por exemplo:

```
insert into dados_alunos (nome, telefone) values ('João Silva', 965244664);
```

Apenas podem ser omitidas as colunas não marcadas como “Not Null” quando a tabela foi inicialmente construída.

É possível introduzir várias linhas numa única transação, como por exemplo:

```
insert into dados_alunos (nome, numero_mecanografico, telefone) values  
  
('João Silva', 98451, 965244644),  
  
('Sandra Santos', 98745, 945678463),  
  
('Miguel Costa', 97486, 917356356);
```

A vantagem da deste método de introdução é a velocidade com que a base de dados consegue processar a informação.

2.3 LER INFORMAÇÃO DA BASE DE DADOS

Para obter informação da base de dados é usado o comando `select`.

O seguinte exemplo seleciona todas as linhas da tabela `dados_alunos`:

```
select * from dados_alunos;
```

É possível especificar quais as colunas que se deseja selecionar como por exemplo:

```
select nome, numero_mecanografico from dados_alunos;
```

É sempre preferível especificar quais as colunas a obter por essa operação ser menos custosa na base de dados.

A base de dados suporta um conjunto extenso de operações que é possível efetuar numa operação de `select`, alguns exemplos:

```
select distinct(nome) from dados_alunos;
```

```
select max(numero_mecanografico) from dados_alunos;
```

Nos casos em que a tabela possui um conjunto muito elevado de linhas é possível ordenar e limitar a informação devolvida da base de dados, por exemplo:

```
select nome, numero_mecanografico from dados_alunos order by numero_mecanografico  
desc limit 2;
```

Esta query seleciona o nome e número mecanográfico da tabela de alunos ordenando os resultados por número mecanográfico por ordem descendente e limitando a apenas 2 resultados.

2.4 ATUALIZAR DADOS DA TABELA

O comando update é utilizado para atualizar uma linha da base de dados, por exemplo:

```
update dados_alunos set nome = 'João Coelho Silva' where id = 1;
```

A base de dados MySQL não permite por default updates que não identifiquem a linha a modificar por uma chave única (como o nosso id), deste modo não é possível atualizar o nome sabendo apenas o mecanográfico do aluno, dado que o número mecanográfico não foi definido como chave única.

2.5 REMOVER DADOS DA TABELA

Para remover linhas da base de dados é usado o comando delete:

```
delete from dados_al where id = 1;
```

2.6 EXERCÍCIOS

2.6.1 Inserir os seguintes dados na tabela dados_alunos por via do comando insert.

Nome	Mecanográfico	Telefone
Maria	95876	912456869
José	94335	913548666
Ana	93568	961548668
Luis	96315	915483584
Ricardo	94666	933547895

2.6.2 Obter os dados do aluno com o mecanográfico mais baixo (use uma subquery).

2.6.3 Adicionar o último nome do aluno alterando a linha correspondente.

2.6.4 Remover o aluno com o mecanográfico mais elevado.

3 COMUNICAR COM BASE DE DADOS COM VB

Use o seguinte programa como exemplo de como ser efetuada a ligação com a base dados a partir do VB.

Leitura Registos		
1	João Coelho Silva	96512484
2	João	96512484
3	João Silva	965244644
4	Sandra Santos	945678463
5	Miguel Costa	917356356

Registo

Id:

Nome:

Nº Telefone:

Figura 1. Interface exemplo.

```
1 Imports MySql.Data.MySqlClient
2 'Para executar este programa necessitamos instalar o MySQL
3 ' Programa: MySQL Installer
4 ' Localização do Ficheiro: C:\Program Files (x86)\MySQL\Samples and Examples 5.6.15
5
6 Public Class Form1
7     Inherits System.Windows.Forms.Form
8
9     Dim cn As New MySqlConnection
10    Dim cmd As New MySqlCommand
11    Dim data_reader As MySqlDataReader
12
13    Private Sub Btn_Connect_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Btn_Connect.Click
14        ' Ligação e Abertura da Base de Dados e Selecção do Esquema
15        cn.ConnectionString = "Server=localhost; User Id=root; Password=123456; Database=alunos"
16
17        Try
18            If cn.State = ConnectionState.Closed Then
19                cn.Open()
20                MsgBox("Ligação Correcta à Base de Dados alunos...")
21            End If
22        Catch ex As Exception
23            cn.Close()
24            MsgBox("Ligação Incorrecta à Base de Dados alunos...")
25        End Try
26        cmd.Connection = cn
27    End Sub
28
29    Private Sub Btn_LerBaseDados_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Btn_LerBaseDados.Click
30        ' Leitura sequencial dos dados existentes na base de dados
31        Dim str As String
32
33        'Selecção da tabela da base de dados para leitura dos dados
34        cmd.CommandText = "SELECT * FROM dados_al"
35        data_reader = cmd.ExecuteReader
36
37        ' Leitura sequencial da tabela...
38        ' Guarda o conteúdo dos registos na variável str
39        While data_reader.Read
40            str = str & data_reader("id") & vbTab & data_reader("nome") & vbTab & data_reader("telefone") & vbCrLf
41        End While
42
43        ' Visualiza os dados de todos os registos
44        Txt_Leitura.Text = str
45        data_reader.Close()
46    End Sub
47
48    Private Sub Btn_InserirRegisto_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Btn_InserirRegisto.Click
49        ' Insere Registo numa Base de Dados
50        cmd.CommandText = "Insert into dados_al(id,nome,telefone) Values('" & Txt_id.Text & "','" & Txt_nome.Text & "','" & Txt_telefone.Text & "')"
51
52        cmd.ExecuteNonQuery()
53    End Sub
54
55    Private Sub Btn_UpdateRegisto_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Btn_UpdateRegisto.Click
56        ' Update de um registo existente
57        cmd.CommandText = "UPDATE dados_al Set nome='" & Txt_nome.Text & "', telefone='" & Txt_telefone.Text & "' where id='" & CInt(Txt_id.Text) & "'"
58
59        cmd.ExecuteNonQuery()
60    End Sub
61
```

```

62
63
64 Private Sub Btn_LerRegisto_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Btn_LerRegisto.Click
65     ' Leitura de um Registo Especifico
66     Dim str As String
67
68     ' Procura do Registo pretendido
69     cmd.CommandText = "Select * From dados_al Where id='" & Txt_id.Text & "'"
70     'cmd.CommandText = "Select * From alunos_telef Where nome='xx' "
71
72     data_reader = cmd.ExecuteReader
73
74     ' Leitura do Registo
75     data_reader.Read()
76     Txt_nome.Text = data_reader("nome")
77     Txt_telefone.Text = data_reader("telefone")
78     data_reader.Close()
79 End Sub
80
81 Private Sub Btn_ApagarRegisto_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Btn_ApagarRegisto.Click
82     ' Apagar Registo através da chave de indexação
83     cmd.CommandText = "Delete From dados_al Where id='" & Txt_id.Text & "'"
84     cmd.ExecuteNonQuery()
85 End Sub
86
87 Private Sub Btn_MaxId_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Btn_MaxId.Click
88     ' Procura o registo com o max id
89     cmd.CommandText = "Select max(id) From dados_al"
90
91     ' executa o comando para leitura do máximo id do registo
92     Dim total As Integer = cmd.ExecuteScalar
93     Txt_MaxId.Text = total.ToString
94 End Sub
95
96 Private Sub Btn_Terminar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Btn_Terminar.Click
97     ' Fim da aplicação
98 End
99 End Sub
100
101 Private Sub Form1_FormClosed(ByVal sender As Object, ByVal e As System.Windows.Forms.FormClosedEventArgs) Handles Me.FormClosed
102     ' MsgBox("Fechar Aplicação...")
103 End Sub
104
105 Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
106     ' MsgBox("A Abrir aplicação...")
107 End Sub
108 End Class

```

4 TRABALHO DE CASA

Neste trabalho (uma semana), pretende-se guardar na base de dados MySQL o estado das saídas e das entradas do PLC ao longo do tempo, de 10 em 10 segundos ou sempre que uma delas se altere.

4.1 APLICAÇÃO LOCAL

Deve alterar a aplicação local, desenvolvida no trabalho anterior, para que além de fazer o que fazia, possa aceder também ao servidor de base de dados “MySQL server”.

Crie uma base de dados “reservatorio” com duas tabelas (Relações): “supervisao” e “controlo”.

Sempre que a aplicação remota enviar uma ordem para ativar ou desativar uma das saídas do PLC, a aplicação local deve inserir na tabela “controlo” uma nova linha (registo) com o novo valor pretendido para as saídas, bem como com a data e a hora em que a ordem (mensagem TCP/IP) foi recebida. Cada linha da tabela tem um número único (#id) que é incrementado sempre que chega uma nova ordem.

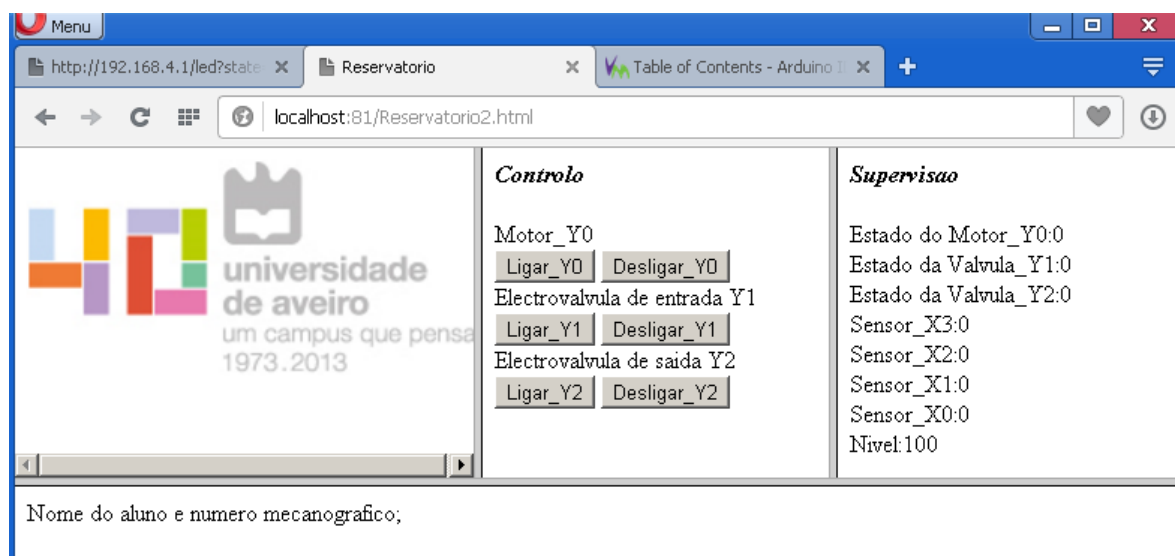
controlo(#id, DataHora, Y0, Y1, Y2)

Sempre que uma das entradas ou saídas digitais Y0, Y1, Y2, X0, X1, X2, X3 for alterada, ou de 10 em 10 segundos, a aplicação local deve guardar o valor de todas elas na tabela “supervisao”

supervisao(#id, DataHora, Y0, Y1, Y2, X0, X1, X2, X3)

4.2 APLICAÇÃO REMOTA

A aplicação remota não deve ser alterada, deve continuar a ter a mesma interface e as mesmas funcionalidades.



4.3 CONHECIMENTOS A ADQUIRIR

Introdução aos gestores de bases, aos objetos em VBasic necessários para aceder aos gestores de bases de dados, e às mensagens SQL.

Familiarização com:

- Linguagem SQL (Structured Query Language)

Tutorial: <http://www.w3schools.com/sql/>

- MySQL

- MySQL **Server** (Gestor de base de dados)
- MySQL **Workbench** (Cliente remoto da base de dados)

Tutorial: <http://dev.mysql.com/doc/workbench/en/>

- MySQL **Connector/NET** (Controlo VBasic)

Tutorial: <http://dev.mysql.com/doc/connector-net/en/connector-net-tutorials-intro.html>

4.4 IMPORTANTE:

- Neste trabalho o PLC não tem programa, todo o controlo e supervisão é efetuado a partir do computador/VBasic remoto, com a intermediação da aplicação local.
- O trabalho será avaliado por questionário individual.
- Deve enviar por elearning os programas desenvolvidos sob pena da nota obtida no questionário não ser considerada.

4.5 BIBLIOGRAFIA:

Database Access with Visual Basic® .NET, Third Edition, By [Jeffrey P. McManus](#), [Jackie Goldstein](#), Addison Wesley, 2003 - Capítulo 4: ADO.NET Data Provider

[MySQL Stored Procedure Programming](#), By Steven Feuerstein, Guy Harrison Publisher: **O'Reilly** Pub Date: **March 2006** - Capítulo 17

Database Access with Visual Basic® .NET, Third Edition, By [Jeffrey P. McManus](#), [Jackie Goldstein](#), Addison Wesley, 2003 - Capítulo 6: ADO.NET - The DataAdapter

An Introduction to Programming Using Visual Basic 2005, By David I. Schneider - University of Maryland, Prentice Hall, 2006 - Capítulo 10 : Database Management

MySQL **Connector/NET** (Controlo VBasic)

Tutorial: <http://dev.mysql.com/doc/connector-net/en/connector-net-tutorials-intro.html>