

ANTES DE INICIAR, ESTOU UTILIZANDO O MYSQL (PODE SER DO WAMP, XAMPP, EASYPHP OU O PRÓPRIO MYSQL RODANDO),

ACESSO AO BANCO DE DADOS – PARTE 1 – SQLS E CONSULTAS, PREENCHENDO OBJETOS DIVERSOS

PRECISA INSTALAR NO VISUAL STUDIO, O MYSQL CONNECTOR, QUE PODE SER BAIXADO DO SITE DO MYSQL

( <http://dev.mysql.com/downloads/connector/net/> )

CRIE UM BANCO (O MEU CHAMA-SE AULA1) E UMA TABELA (AULAS) CONTENDO 3 CAMPOS (CODIGO INT AUTONUMERAÇÃO, NOME VARCHAR (255) , EMAIL VARCHAR (255))

1 - Criar um projeto vazio.

2 - Criar um WebForm

---

Adicionando a classe para criação da conexão com o banco

1 – Botão direito no nome da Solution que você criou (no Solution Explorer) ADD CLASS.

2 – Dê um nome para a classe e OK

3 – Botão direito no nome da Solution que você criou, ADD Reference.

4 – No VS2010, localize na aba .NET o MySQL.Data. Selecione ele e aperte OK. No 2012 em diante, vá em Examinar, e navegue em Arquivos de Programa, MySQL, MySqlconnector, Assemblies e selecione o MySQL.Data.dll.

5 – Adicione na seção using da classe: using MySql.Data.MySqlClient;

6 – Aí vamos escrever a classe

```
public static MySqlConnection fazerconexao()
{
    MySqlConnection conectar = new MySqlConnection("server=127.0.0.1; database=aulas;
uid=root; pwd=;");
    return conectar;
}
```

7 – Feito isso, vamos retornar ao nosso projeto e testar se a conexão será efetuada sem nenhum problema. Vamos colocar a chamada da conexão dentro do evento Page\_Load de nosso webform. Adicione a chamada ao método fazerconexão() classe bdcomun .

```
MySqlConnection cn = bdcomun.fazerconexao();
cn.Open();
```

8 – Vamos rodar. Se der algum erro na conexão , seremos avisados com uma mensagem de erro.

----- Mostrar registros da tabela do banco -----

1 – Vamos inserir 4 labels e 2 textbox para indicarmos o Código (na label, pois é autonumeração e o cliente não irá alterá-lo), Nome e Email (em textboxes) - (campos da tabela Aula1 no MySQL).

2 – Depois de dar os nomes adequados aos objetos e de montar o layout desejado, vamos inserir (abaixo da chamada da conexão) no Page\_Load, as linhas que irão fazer a busca dos dados no banco pela conexão que abrimos. Antes de tudo, adicione a chamada à referência MySQL no webform principal (`using MySQL.Data.MySqlClient;`)

3 – No Page\_load, adicione:

```
MySQLConnection cn = bdcomun.fazerconexao();
cn.Open();
MySQLCommand cmd = new MySQLCommand();
MySQLDataReader resultado;
cmd.CommandText = "Select * from aula1";
cmd.Connection = cn;
resultado = cmd.ExecuteReader();
resultado.Read();

lbl_cod.Text = resultado[0].ToString();
txt_nome.Text = resultado[1].ToString();
txt_email.Text = resultado[2].ToString();
```

4 – E se quisermos procurar uma pessoa específica no banco de dados? Adicione um TextBox (txt\_procura) e um botão. No evento click do botão coloque o seguinte:

```
MySQLConnection cn = bdcomun.fazerconexao();
cn.Open();
MySQLCommand cmd = new MySQLCommand();
MySQLDataReader resultado;
cmd.CommandText = "Select * from aula1 where nome like '%" + txt_procura.Text +
"%'";
cmd.Connection = cn;
resultado = cmd.ExecuteReader();
if (resultado.HasRows)
{
    resultado.Read();

    lbl_cod.Text = resultado[0].ToString();
    txt_nome.Text = resultado[1].ToString();
    txt_email.Text = resultado[2].ToString();
}
else
{
    lbl_cod.Text = "";
    txt_nome.Text = "";
    txt_email.Text = "";
}
}
```

---

Para mostrar todos os dados de uma SQL em uma tabela (GridView);

1) Adicione um objeto GridView (está na aba Data).

Adicione nas bibliotecas a System.data, com `using System.Data;` após a linha do `using MySQL.Data.MySqlClient;`

- 2) Vamos criar um método para carregar os dados do MySQL e passa-los para o GridView, coloque o seguinte método dentro de seu código:

```
private void preenche_gridview()
{
    MySqlConnection cn = bdcomun.fazerconexao();
    cn.Open();
    MySqlDataAdapter adp = new MySqlDataAdapter("Select * from aula1 order by cod", cn);
    DataTable dt = new DataTable();

    adp.Fill(dt);
    if (dt.Rows.Count > 0)
    {
        grd_dados.DataSource = dt;
        grd_dados.DataBind();
    }
}
```

- 3) Vamos colocar a chamada ao método preenche\_gridview() em um Botão. Adicione um botão e no evento click coloque o nome do método preenche\_gridview().
- 

Vamos preencher um DropDownList com os nomes cadastrados no banco de dados e seu código no value

- 1) Insira o DropDownList; Não preencha nada em seus itens. Ative apenas o Enable Autopostback. Insira uma Label também para mostrarmos o código do usuário selecionado.
- 2) Vamos criar um botão para preencher o dropdownlist no evento click.

```
protected void btn_dropdown_Click(object sender, EventArgs e)
{
    MySqlConnection cn = bdcomun.fazerconexao();
    cn.Open();
    MySqlCommand cmd = new MySqlCommand();
    MySqlDataReader resultado;
    cmd.CommandText = "Select cod,nome from aula1 order by nome";
    cmd.Connection = cn;
    resultado = cmd.ExecuteReader();
    if (resultado.HasRows)
    {
        ddl_nomes.Items.Add(new ListItem("Selecione uma opção","0"));

        while (resultado.Read())
        {
            ddl_nomes.Items.Add(new ListItem(resultado[1].ToString(),
resultado[0].ToString()));
        }
    }
}
```

- 3) Vamos colocar o código no dropdownlist.SelectedIndexChanged para que ele atualize a label com o código do nome selecionado

```
protected void ddl_nomes_SelectedIndexChanged(object sender, EventArgs e)
{
    lbl_codigo.Text = ddl_nomes.SelectedItem.Value.ToString();
}
```

---