

Google Chart

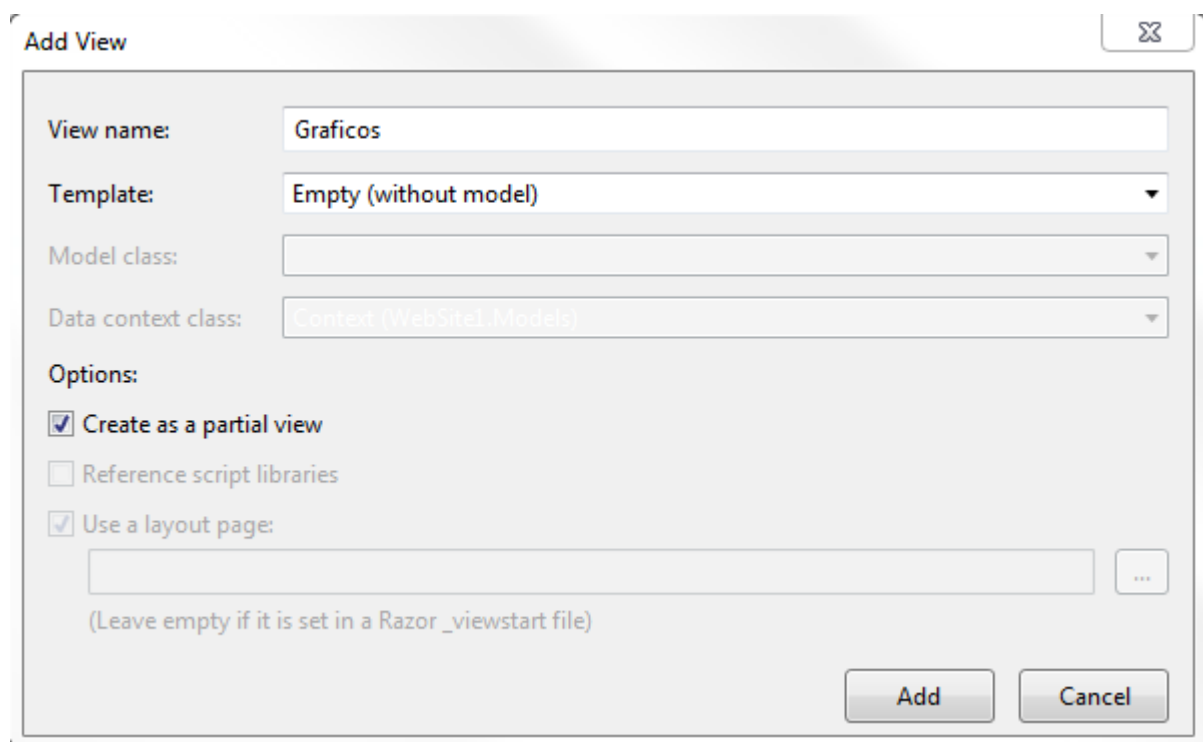
Adicionando Action

No PublicoController, adicionar o trecho

```
public ActionResult Graficos()
{
    return View();
}
```

Adicionar View

Botão direito na Action acima, Add View.



Alterando Graficos.chhtml

Definir o Layout (sem necessidade de Login)

```
@{
    Layout = "~/Views/Shared/_LayoutUser.cshtml";
    ViewBag.Title = "Gráficos";
}
```

Definir área do gráfico

```
<h2>Gráficos</h2>

<h3>Gráfico 1 - Estados por Região</h3>
<div id="divGráficoPizza"></div>
```

Criar o script para o Gráficos

```
@section Scripts{
    @Scripts.Render("http://www.google.com/jsapi")
    @*<script type="text/javascript" src="https://www.google.com/jsapi"></script>*@

<script type="text/javascript">

    //Carrega o tipo de gráfico a ser utilizado
    google.load('visualization', '1.0', { 'packages': ['corechart'] });
    google.setOnLoadCallback(drawChart);

    function drawChart() {
        var data = new google.visualization.DataTable();

        //Adiciona as colunas do gráfico
        data.addColumn('string', 'Região');
        data.addColumn('number', 'Quantidade');

        //Adiciona as linhas do gráfico
        data.addRows([
            ['Norte', 7],
            ['Sul', 3],
            ['Centro Oeste', 3],
            ['Sudeste', 4],
            ['Nordeste', 9]
        ]);

        //Opções adicionais do gráfico
        var options = {
            'title': '',
            'width': 400,
            'height': 300
        };

        //Define onde será renderizado o gráfico, no caso no elemento divGraficoPizza
        var chart = new
        google.visualization.PieChart(document.getElementById('divGraficoPizza'));

        //Renderiza o Gráfico com as opções adicionais definidas
        chart.draw(data, options);
    }
</script>
}
```

Executar a aplicação

Acessar a URL <http://localhost:porta/Publico/Graficos>

Adicionando mais um gráfico (em Graficos.cshtml)

Adicionar abaixo do Gráfico 1

```
<h3>Gráfico 2</h3>
<div id='graficoPizza'></div>
```

Adicionar um HtmlHelper

Na pasta **Repositories**, adicionar uma classe chamada **HtmlHelpers.cs**

Essa classe possui o método **GraficoPizza** que define a estrutura do gráfico.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Web;

namespace System.Web.Mvc
{
    public static class HtmlHelpers
    {
        public static string GraficoPizza(this HtmlHelper html,
                                         string nomeGrafico,
                                         Dictionary<object, object> colunas,
                                         Dictionary<object, object> linhas,
                                         int width,
                                         int height)
        {
            StringBuilder sb = new StringBuilder();

            //sb.AppendFormat("<div id='{0}'></div>", nomeGrafico);
            sb.AppendLine("<script language='javascript'>");
            sb.AppendLine("google.load('visualization', '1.0', { 'packages':
['corechart'] });");
            sb.AppendLine("google.setOnLoadCallback(drawChart);");
            sb.AppendLine("function drawChart() {");
            sb.AppendLine("var data = new google.visualization.DataTable();");

            foreach (var coluna in colunas)
                sb.AppendLine(String.Format("data.addColumn('{0}', '{1}');", coluna.Key,
coluna.Value));

            foreach (var linha in linhas)
                sb.AppendLine(String.Format("data.addRow(['{0}', {1}]);", linha.Key,
linha.Value));

            sb.AppendLine("var options = { 'title': 'Título do Gráfico,'");
            sb.AppendLine(String.Format("    'width': '{0}',", width));
            sb.AppendLine(String.Format("    'height': '{0}' ", height));
            sb.AppendLine("    }; ");
            sb.AppendLine(String.Format("var chart = new
google.visualization.PieChart(document.getElementById('{0}'));", nomeGrafico));
            sb.AppendLine("chart.draw(data, options);");
            sb.AppendLine("}");
            sb.AppendLine("</script>");

            return sb.ToString();
        }
    }
}
```

Adicionando dados ao gráfico

Adicionar mais uma classe na pasta Repositores, denominada **DataChart.cs**

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;

namespace WebSite1.Repositories
{
    public class DataChart
    {
        public static Dictionary<object, object> SelecionaColunasGraficoPizza()
        {
            Dictionary<object, object> dic = new Dictionary<object, object>();
            dic.Add("string", "Pessoal");
            dic.Add("number", "Quantidade");

            return dic;
        }

        public static Dictionary<object, object> SelecionaLinhasGraficoPizza()
        {
            Dictionary<object, object> dic = new Dictionary<object, object>();

            dic.Add("Professores", 63);
            dic.Add("Alunos", 1200);
            dic.Add("Funcionários", 18);
            dic.Add("Estagiários", 12);
            dic.Add("Outras", 6);

            return dic;
        }
    }
}
```

Neste arquivo é possível acessar os dados do banco de dados para que sejam retornados ao gráfico.

Adicionando a chamada ao gráfico

No arquivo Graficos.cshtml, dentro do Scripts, adicionar o trecho

```
@section Scripts{
    @Scripts.Render("http://www.google.com/jsapi")
    @*<script type="text/javascript" src="https://www.google.com/jsapi"></script>*@

    @f
        //GraficoPizza(nomeDaDIV, colunas, linhas, width, height)
        string grafico = Html.GraficoPizza("graficoPizza",
        WebSite1.Repositories.DataChart.SelecionaColunasGraficoPizza(),
        WebSite1.Repositories.DataChart.SelecionaLinhasGraficoPizza(), 400, 300);
        //grafico = grafico.Replace("\r\n", "");
    }
    @Html.Raw(grafico.ToString())
```

Executar a aplicação

Acessar a URL <http://localhost:porta/Publico/Graficos>

Buscando informações do Banco de dados

Na classe DataChart.cs, adicione mais um método

```
public static Dictionary<object, object> SeleccionaLinhasBanco()
{
    Dictionary<object, object> dic = new Dictionary<object, object>();

    WebSite1.Models.Context db = new Models.Context();

    var result = (from p in db.Clientes
                  join f in db.Cidades on p.CidadeID equals f.CidadeID
                  group p by p.Cidade.Nome into g
                  select new { CidadeNome = g.Key, Quantidade =
g.Count() }).OrderBy(o => o.CidadeNome);

    foreach (var item in result)
    {
        dic.Add(item.CidadeNome, item.Quantidade);
    }

    return dic;
}
```

Altere a chamada do gráfico (em Graficos.cshtml)

Altere de ...DataChart.SeleccionaLinhasGráficoPizza()

Para ...DataChart.SeleccionaLinhasBanco()

```
@f
//GráficoPizza(nomeDaDIV, colunas, linhas, width, height)
string grafico = Html.GráficoPizza("graficoPizza",
WebSite1.Repositories.DataChart.SeleccionaLinhasGráficoPizza(),
WebSite1.Repositories.DataChart.SeleccionaLinhasBanco(), 400, 300);
//grafico = grafico.Replace("\r\n", "");
}
@Html.Raw(grafico.ToString())
```

Executar a aplicação

Acessar a URL <http://localhost:porta/Publico/Graficos>

Google Chart

<https://developers.google.com/chart/>

Exemplo com Bar Chart

<https://developers.google.com/chart/interactive/docs/gallery/barchart#examples>

Exemplos de Linq

<https://code.msdn.microsoft.com/101-LINQ-Samples-3fb9811b>

Uma outra forma de usar Google Chart

<https://code.google.com/archive/p/googlechartsharp/downloads>