# 什么是报表

[forms for reporting to the higher organizations] 向上级报告情况的表格。

简单的说：报表就是用表格、图表等形式来动态显示数据，并为使用者提供浏览、打印、导出和分析功能，可以用公式表示为：“报表 = 多样的格式 + 动态的数据”。

# 报表要完成的功能

1、多样格式动态的展示数据。多样格式可以是表格，也可以是各种图形（折线图 柱状图、散点图、饼图、K线图等）。

1. 以直观的格式展示数据（表格、图形）。
2. 提供导出或者保存功能。
3. 方便对结果进行打印。

汽车销量 降水量 平均气温

# 常用的报表开发库ECharts介绍

## ECharts简介

官网地址：<https://echarts.apache.org/zh/index.html>

ECharts，一个使用 JavaScript 实现的开源可视化库，可以流畅的运行在 PC 和移动设备上，兼容当前绝大部分浏览器（IE8/9/10/11，Chrome，Firefox，Safari等），底层依赖轻量级的矢量图形库 ZRender，提供直观，交互丰富，可高度个性化定制的数据可视化图表。

## 丰富的可视化类型

### 内置图形

ECharts 提供了常规的折线图、柱状图、散点图、饼图、K线图，用于统计的盒形图，用于地理数据可视化的地图、热力图、线图，用于关系数据可视化的关系图、treemap、旭日图，多维数据可视化的平行坐标，还有用于 BI 的漏斗图，仪表盘，并且支持图与图之间的混搭。

### 自定义图形

除了已经内置的包含了丰富功能的图表，ECharts 还提供了自定义系列，只需要传入一个renderItem函数，就可以从数据映射到任何你想要的图形，更棒的是这些都还能和已有的交互组件结合使用而不需要操心其它事情。

## 使用步骤

### 获取 ECharts

ECharts 正在 Apache 开源基金会孵化中

地址：<https://echarts.apache.org/zh/index.html>



### 引入 ECharts

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <!-- 引入 ECharts 文件 -->  <script src="echarts.min.js"></script>  </head>  </html> |

### 绘制一个简单的图表

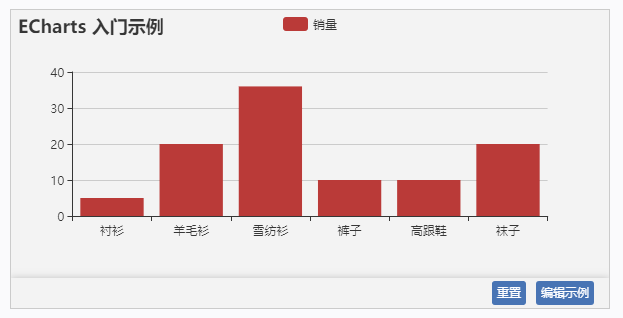
在绘图前我们需要为 ECharts 准备一个具备高宽的 DOM 容器。

|  |
| --- |
| <body>  <!-- 为 ECharts 准备一个具备大小（宽高）的 DOM -->  <div id="main" style="width: 600px;height:400px;"></div>  </body> |

然后就可以通过 [echarts.init](http://echarts.baidu.com/api.html#echarts.init) 方法初始化一个 echarts 实例并通过 [setOption](http://echarts.baidu.com/api.html#echartsInstance.setOption) 方法生成一个简单的柱状图，下面是完整代码。

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <title>ECharts</title>  <!-- 引入 echarts.js -->  <script src="echarts.min.js"></script>  </head>  <body>  <!-- 为ECharts准备一个具备大小（宽高）的Dom -->  <div id="main" style="width: 600px;height:400px;"></div>  <script type="text/javascript">  // 基于准备好的dom，初始化echarts实例  var myChart = echarts.init(document.getElementById('main'));  // 指定图表的配置项和数据  var option = {  title: {  text: 'ECharts 入门示例'  },  tooltip: {},  legend: {  data:['销量']  },  xAxis: {  data: ["衬衫","羊毛衫","雪纺衫","裤子","高跟鞋","袜子"]  },  yAxis: {},  series: [{  name: '销量',  type: 'bar',  data: [5, 20, 36, 10, 10, 20]  }]  };  // 使用刚指定的配置项和数据显示图表。  myChart.setOption(option);  </script>  </body>  </html> |

这样你的第一个图表就诞生了！



举一反三：

如果要制作其他柱形图，都可以采用这种格式，只需要把标题和名称改了就行，根据实际情况传入数据，就可以得到你想要的柱形图。

# 几个echarts 版本

一般来说，可以直接从 [CDN](https://www.jsdelivr.com/package/npm/echarts) 中获取构建后的 Apache ECharts (incubating)TM，也可以从 [GitHub](https://github.com/apache/incubator-echarts/releases) 中的 echarts/dist 文件夹中获取构建好的 echarts，这都可以直接在浏览器端项目中使用。这些构建好的 echarts 提供了下面这几种定制：

* 完全版：echarts/dist/echarts.js，体积最大，包含所有的图表和组件，所包含内容参见：echarts/echarts.all.js。
* 常用版：echarts/dist/echarts.common.js，体积适中，包含常见的图表和组件，所包含内容参见：echarts/echarts.common.js。
* 精简版：echarts/dist/echarts.simple.js，体积较小，仅包含最常用的图表和组件，所包含内容参见：echarts/echarts.simple.js。

我们也可以自己构建 echarts，能够仅仅包括自己所需要的图表和组件。可以用这几种方式自定义构建：

* [在线自定义构建](https://echarts.apache.org/zh/builder.html)：比较方便。

# Echarts实例

<https://echarts.apache.org/examples/zh/index.html>

直方图

折线图

曲线图

面积图

饼状图

环形图

# 异步加载数据

## 异步加载原理

入门示例中的数据是在初始化后setOption中直接填入的，但是很多时候可能数据需要异步加载后再填入。ECharts 中实现异步数据的更新非常简单，在图表初始化后不管任何时候只要通过 jQuery 等工具异步获取数据后通过 setOption 填入数据和配置项就行。

|  |
| --- |
| var myChart = echarts.init(document.getElementById('main'));  $.get('data.json').done(function (data) {  myChart.setOption({  title: {  text: '异步数据加载示例'  },  tooltip: {},  legend: {  data:['销量']  },  xAxis: {  data: ["衬衫","羊毛衫","雪纺衫","裤子","高跟鞋","袜子"]  },  yAxis: {},  series: [{  name: '销量',  type: 'bar',  data: [5, 20, 36, 10, 10, 20]  }]  });  }); |

## 完整异步加载例子前端代码

|  |
| --- |
| <%@ page language=*"java"* contentType=*"text/html; charset=UTF-8"*  pageEncoding=*"UTF-8"*%>  <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">  <html>  <head>  <meta http-equiv=*"Content-Type"* content=*"text/html; charset=UTF-8"*>  <title>Insert title here</title>  </head>  <body>  <div>  年份选择：<select id=*"yearselect"*>  <option value=*"2020"*>2020</option>  <option value=*"2021"*>2021</option>  <option value=*"2022"*>2022</option>  <option value=*"2023"*>2023</option>  </select>  <div id=*"main"* style="width: *600px*;height:*400px*;"></div>  </div>  <script src=*"../js/jquery-1.11.0.js"*></script>  <script src=*"../js/echarts.min.js"*></script>  <script>  **var** myChart = echarts.init(document.getElementById('main'));  **var** option = {  xAxis: {  type: 'category',  data: []  },  yAxis: {  type: 'value'  },  series: [{  data: [],  type: 'line'  }]  };  myChart.setOption(option);  **function** getajaxdata(objyear) {  $.ajax({  type: "post",  url: "../ajax",  data: { rnum:9, cyear: objyear },  timeout: 5000,  dataType: "json",  async: **true**,//默认设置为true，所有请求均为异步请求  //cache：true,//默认为true（当dataType为script时，默认为false）设置为false将不会从浏览器缓存中加载请求信息。  success: **function** (data) {  //{"datamonths":["1月","2月","3月"],"dataitems":[100,200,150]}  **var** optionhasvalue = {  xAxis: {  type: 'category',  data: data.datamonths  },  yAxis: {  type: 'value'  },  series: [{  data: data.dataitems,  type: 'line'    }]  };  myChart.setOption(optionhasvalue);  }  });  }  getajaxdata(2020);  $("#yearselect").change(**function** () {  **var** v = $(**this**).val();  getajaxdata(v);  });  </script>  </body>  </html> |

## 后端动态生成json格式的数据

|  |
| --- |
| //每月进货总金额报表  **protected** **void** getSaleSumPricesByMonth(HttpServletRequest request, HttpServletResponse response) **throws** ServletException, IOException {  String StrSql="SELECT DATE\_FORMAT(ctime,'%Y-%m') as ctime,sum(sumprice) as sumprice FROM tborderhead where year(ctime)=? GROUP BY ctime";  String cyear=request.getParameter("cyear");  List<Object> params= **new** ArrayList<Object>();  params.add(cyear);  DBHelper db=**new** DBHelper();  List<Map<String, Object>> reslist = **null**;  String html="";  //{"datamonths":["1月","2月","3月"],"dataitems":[100,200,150]}  **try** {  reslist=db.executeQuery(StrSql, params);  String html\_datamonths="[";  String html\_dataitems="[";  **int** i=1;  **for** (Map<String, Object> m : reslist) {  **if**(i==reslist.size())  {  html\_datamonths+="\""+m.get("ctime")+"\"";  html\_dataitems+=m.get("sumprice");  }  **else**  {  html\_datamonths+="\""+m.get("ctime")+"\",";  html\_dataitems+=m.get("sumprice")+",";  }  i++;  }  html\_datamonths+="]";  html\_dataitems+="]";  html="{\"datamonths\":"+html\_datamonths+",\"dataitems\":"+html\_dataitems+"}";  } **catch** (SQLException e) {  e.printStackTrace();  }  response.setCharacterEncoding("utf-8");  response.setContentType("text/json;charset=utf-8");  response.getWriter().write(html);    } |

# 分类汇总

# loading 动画

如果数据加载时间较长，一个空的坐标轴放在画布上也会让用户觉得是不是产生 bug 了，因此需要一个 loading 的动画来提示用户数据正在加载。

ECharts 默认有提供了一个简单的加载动画。只需要调用 showLoading 方法显示。数据加载完成后再调用 hideLoading 方法隐藏加载动画。

|  |
| --- |
| myChart.showLoading();  $.get('data.json').done(function (data) {  myChart.hideLoading();  myChart.setOption(...);  }); |

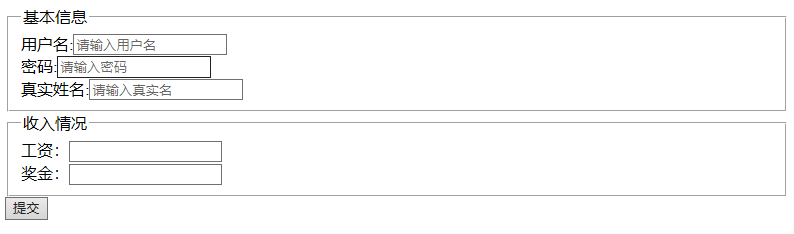
# 数据的动态更新

ECharts 由数据驱动，数据的改变驱动图表展现的改变，因此动态数据的实现也变得异常简单。所有数据的更新都通过 setOption实现，你只需要定时获取数据，setOption 填入数据，而不用考虑数据到底产生了那些变化，ECharts 会找到两组数据之间的差异然后通过合适的动画去表现数据的变化。

# 打印技术

# 组合查询技术

# fieldset分组标签介绍



|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | |  | <!DOCTYPE html> | |  | <html> | |  | <head> | |  | <meta charset="UTF-8"> | |  | <title></title> | |  | </head> | |  | <body> | |  | <form action="index.html" method="get" id="form1" > | |  | <fieldset id="f1"> | |  | <!--让用户看到的提示信息--> | |  | <legend style="text-align: center;">基本信息</legend> | |  |  | |  |  | |  | <!--基本信息--> | |  | 用户名:<input type="text" name="userName" id="" placeholder="请输入用户名" /> | |  | <br/> | |  | 密码:<input type="password" name="pwd" autofocus="autofocus" placeholder="请输入密码"/> | |  | <br/> | |  | 真实姓名:<input type="text" name="userName2" id="" placeholder="请输入真实名" /> | |  | <br/> | |  | </fieldset> | |  | <!--收入情况--> | |  | <fieldset> | |  | <legend>收入情况</legend> | |  | 工资：<input type="text" value="" name="gz" /><br/> | |  | 奖金：<input type="text" value="" name="jj"/><br/> | |  | </fieldset> | |  |  | |  |  | |  | <input type="submit" value="提交" /> | |  | </form> | |  |  | |  | </body> | |  | </html> | |

<http://www.htmleaf.com/>

搜索关键字表单美化