●官网

<https://pyecharts.org/#/>

●画廊地址

<https://gallery.pyecharts.org/#/>

●基本用法

from pyecharts.charts import Line  
  
# 创建一个折线图对象  
line = Line()  
  
# 给折线图对象添加x轴的数据  
line.add\_xaxis(["中国", "美国", "日本"])  
  
# 给折线图对象添加y轴的数据  
line.add\_yaxis("GDP", [30, 20, 25])  
  
# 通过render方法，将代码生成为图像  
line.render()

执行以上代码后，会自动生成一个render.html的文件。用浏览器打开查看图表

●

from pyecharts.charts import Line  
  
# 创建一个折线图对象  
line = Line()  
  
# 给折线图对象添加x轴的数据  
line.add\_xaxis(["中国", "美国", "日本"])  
  
# 给折线图对象添加y轴的数据  
line.add\_yaxis("GDP", [30, 20, 25])  
line.add\_yaxis("满意度", [20, 12, 15])  
  
# 通过render方法，将代码生成为图像  
line.render("多条折线图.html") # 这里是生成HTML文件的名称

■全局配置项

文档地址

<https://pyecharts.org/#/zh-cn/global_options>

●案例

from pyecharts.charts import Line  
from pyecharts.options import TitleOpts, LegendOpts, ToolboxOpts, VisualMapOpts  
  
# 创建一个折线图对象  
line = Line()  
  
# 给折线图对象添加x轴的数据  
line.add\_xaxis(["中国", "美国", "日本"])  
  
# 给折线图对象添加y轴的数据  
line.add\_yaxis("GDP", [30, 20, 25])  
  
  
# 设置全局配置项set\_global\_opts  
line.set\_global\_opts(  
 title\_opts=TitleOpts(title="GDP展示", pos\_left="center", pos\_bottom="2%"), # 显示标题  
 legend\_opts=LegendOpts(is\_show=True), # 显示图例（图例默认就是显示的）  
 toolbox\_opts=ToolboxOpts(is\_show=True), # 显示工具箱  
 visualmap\_opts=VisualMapOpts(is\_show=True) #显示视觉映射  
)  
  
# 通过render方法，将代码生成为图像  
line.render()

■柱状图

●基础柱状图

通过reversal\_axis()来使x轴和y轴反转，达到使柱状图横向展示的效果

from pyecharts.charts import Bar  
  
# 创建一个柱状图对象  
bar = Bar()  
  
# 给对象添加x轴的数据  
bar.add\_xaxis(["中国", "美国", "日本"])  
  
# 给对象添加y轴的数据  
bar.add\_yaxis("GDP", [30, 20, 25])  
bar.add\_yaxis("满意度", [20, 12, 15])  
  
bar.reversal\_axis() # 反转x轴和y轴（使柱子横向展示）  
  
# 通过render方法，将代码生成为图像  
bar.render("柱状图.html") # 这里是生成HTML文件的名称

●带有时间线的柱状图

from pyecharts.charts import Bar, Timeline  
  
# 创建一个柱状图对象  
bar = Bar()  
  
# 给对象添加x轴的数据  
bar.add\_xaxis(["中国", "美国", "日本"])  
  
# 给对象添加y轴的数据  
bar.add\_yaxis("GDP", [30, 20, 25])  
bar.add\_yaxis("满意度", [20, 12, 15])  
  
bar.reversal\_axis() # 反转x轴和y轴（使柱子横向展示）  
  
  
# 创建一个柱状图对象  
bar2 = Bar()  
  
# 给对象添加x轴的数据  
bar2.add\_xaxis(["中国", "美国", "日本"])  
  
# 给对象添加y轴的数据  
bar2.add\_yaxis("GDP", [20, 10, 15])  
bar2.add\_yaxis("满意度", [10, 6, 9])  
  
bar2.reversal\_axis() # 反转x轴和y轴（使柱子横向展示）  
  
  
# 创建一个柱状图对象  
bar3 = Bar()  
  
# 给对象添加x轴的数据  
bar3.add\_xaxis(["中国", "美国", "日本"])  
  
# 给对象添加y轴的数据  
bar3.add\_yaxis("GDP", [35, 25, 28])  
bar3.add\_yaxis("满意度", [30, 16, 19])  
  
bar3.reversal\_axis() # 反转x轴和y轴（使柱子横向展示）  
  
# 创建时间线对象  
timeline = Timeline()  
  
# 为timeline对象添加柱状图  
timeline.add(bar, "2020年数据")  
timeline.add(bar2, "2021年数据")  
timeline.add(bar3, "2022年数据")  
  
# 时间线对象自动播放设置  
timeline.add\_schema(  
 play\_interval=1000, # 自动播放间隔，单位毫秒  
 is\_timeline\_show=True, # 是否在播放时显示时间线  
 is\_auto\_play=True, # 是否自动播放  
 is\_loop\_play=True #是否循环播放  
)  
  
# 通过render方法，将代码生成为图像  
timeline.render("柱状图.html") # 这里是生成HTML文件的名称

■地图

●中国地图

from pyecharts.charts import Map  
from pyecharts.options import \*  
  
# 构建地图对象  
map = Map()  
  
data = [  
 ("北京市", 99),  
 ("上海市", 199),  
 ("天津市", 299),  
 ("重庆市", 399),  
 ("广东省", 499),  
 ("湖北省", 10000)  
]  
  
# 参数分别是，地图名称，数据，地图类型（china为中国地图，默认也是中国地图。如果是省地图，可以写省的名字）  
map.add("测试地图", data, "china")  
  
map.set\_global\_opts(  
 visualmap\_opts=VisualMapOpts(  
 is\_show=True,  
 is\_piecewise=True,  
 pieces=[  
 {"min": 1, "max": 9, "label": "1-9人", "color": "#CCFFFF"},  
 {"min": 10, "max": 99, "label": "10-99人", "color": "#FF6666"},  
 {"min": 100, "max": 499, "label": "100-499人", "color": "#990033"},  
 {"min": 500, "label": "500人以上", "color": "#CCCCCC"},  
 ]  
 )  
)  
  
map.render("地图.html")

●河北省地图

from pyecharts.charts import Map  
from pyecharts.options import \*  
  
# 构建地图对象  
map = Map()  
  
data = [  
 ("石家庄市", 99),  
 ("唐山市", 199),  
 ("承德市", 699),  
]  
  
# 参数分别是，地图名称，数据，地图类型（china为中国地图，默认也是中国地图。如果是省地图，可以写省的名字）  
map.add("测试地图", data, "河北")  
  
map.set\_global\_opts(  
 visualmap\_opts=VisualMapOpts(  
 is\_show=True,  
 is\_piecewise=True,  
 pieces=[  
 {"min": 1, "max": 9, "label": "1-9人", "color": "#CCFFFF"},  
 {"min": 10, "max": 99, "label": "10-99人", "color": "#FF6666"},  
 {"min": 100, "max": 499, "label": "100-499人", "color": "#990033"},  
 {"min": 500, "label": "500人以上", "color": "#CCCCCC"},  
 ]  
 )  
)  
  
map.render("地图.html")