

In [1]:

```
from pyecharts.charts import Bar, Timeline
from pyecharts import options as opts
import pandas as pd
import numpy as np
```

In []:

```
# pyechart有哪些颜色? 打印颜色池
Bar().options['color']
# '#c23531' 红色
```

In [2]:

```
data_path=open('D:/! 江科2023/01教学/6-数据分析实验2022/
```

In [3]:

```
df_world =pd.read_csv(data_path,index_col=0)
df_world.head()
```

	中国	美国	日本	印度	德国	英国	法国	意大利	俄罗斯	加拿大	澳大利亚	巴西	章
年份													
1960	597	5433	443	370	2158	732	622	404	5547	405	186	152	4
1961	501	5633	535	392	2158	777	675	448	5547	409	197	152	2
1962	472	6051	607	422	2158	812	756	504	5547	422	199	199	2
1963	507	6386	695	484	2158	866	848	577	5547	450	215	230	4
1964	597	6858	817	565	2158	944	940	632	5547	494	238	212	3

1: 构造数据

In [4]:

```
cats0=df_world.columns.to_list() #15个国家-列表, 第一个
len(cats0)
```

15

In [5]:

```
color0=Bar().options['color'][:15] # 15个颜色-列表, 第-
```

In [6]:

```
cats_color=dict(list(zip(cats0,color0))) # 将二元组列  
cats_color
```

```
{'中国': '#c23531',  
 '美国': '#2f4554',  
 '日本': '#61a0a8',  
 '印度': '#d48265',  
 '德国': '#749f83',  
 '英国': '#ca8622',  
 '法国': '#bda29a',  
 '意大利': '#6e7074',  
 '俄罗斯': '#546570',  
 '加拿大': '#c4ccd3',  
 '澳大利亚': '#f05b72',  
 '巴西': '#ef5b9c',  
 '韩国': '#f47920',  
 '瑞典': '#905a3d',  
 '西班牙': '#fab27b'}
```

In [7]:

```
year0=df_world.index.to_list()
```

```
values=[]
▼ for i in year0:
    # 1960年打印15次
▼ for j in cats0:
    # 当此时i=1960, j=中国
    c=cats_color[j]
    values.append([i,j, c ,df_world.loc[i,j]])
print(values)
```

58], [1963, '英国', '#ca8622', 866], [1963, '法国', '#bc
963, '意大利', '#6e7074', 577], [1963, '俄罗斯', '#546570',
3, '加拿大', '#c4ccd3', 450], [1963, '澳大利亚', '#f05b72',
'巴西', '#ef5b9c', 230], [1963, '韩国', '#f47920', 40],
'#905a3d', 202], [1963, '西班牙', '#fab27b', 191], [1964,
31', 597], [1964, '美国', '#2f4554', 6858], [1964, '日本',
17], [1964, '印度', '#d48265', 565], [1964, '德国', '#74
[1964, '英国', '#ca8622', 944], [1964, '法国', '#bda29a',
'意大利', '#6e7074', 632], [1964, '俄罗斯', '#546570', 5
拿大', '#c4ccd3', 494], [1964, '澳大利亚', '#f05b72', 23
西', '#ef5b9c', 212], [1964, '韩国', '#f47920', 35], [19
5a3d', 225], [1964, '西班牙', '#fab27b', 213], [1965, '日
704], [1965, '美国', '#2f4554', 7437], [1965, '日本', '#
[1965, '印度', '#d48265', 596], [1965, '德国', '#749f83',
'英国', '#ca8622', 1018], [1965, '法国', '#bda29a', 1015
利', '#6e7074', 680], [1965, '俄罗斯', '#546570', 5547],

```
▼ # 基于values和列表构建新的二维表
df_gdb=pd.DataFrame(data=values,columns=['年份','国家'])
▼ df_gdb.to_csv('D:/! 江科2023/01教学/6-数据分析实验2022/世
encoding='ANSI')

"写入完成"
```

'写入完成'

2: 选取每年的数据并定义x\y轴数据:

In [10]:

```
df_year=pd.DataFrame(df_gdb.loc[df_gdb['年份']==2023,  
df_year
```

	年份	国家	颜色	GDP
958	2023	瑞典	#905a3d	5376
959	2023	西班牙	#fab27b	12812
955	2023	澳大利亚	#f05b72	16800
957	2023	韩国	#f47920	17000
953	2023	俄罗斯	#546570	18600
954	2023	加拿大	#c4ccd3	21180
956	2023	巴西	#ef5b9c	21270
952	2023	意大利	#6e7074	21860
951	2023	法国	#bda29a	30500
950	2023	英国	#ca8622	33330
948	2023	印度	#d48265	39320
947	2023	日本	#61a0a8	42474
949	2023	德国	#749f83	44560
945	2023	中国	#c23531	178890
946	2023	美国	#2f4554	273564

In [11]:

```
cats_list= df_year['国家'].to_list()
cols_list= df_year['颜色'].to_list()
gdp_list= df_year['GDP'].to_list()
gdp_list
```

```
[5376,
 12812,
 16800,
 17000,
 18600,
 21180,
 21270,
 21860,
 30500,
 33330,
 39320,
 42474,
 44560,
 178890,
 273564]
```

2.1:如何 定义y轴让颜色不一样?

- 每个条子颜色不同，比如中国是红色
- 每个条子长度不同，长度=gdp的值

In [12]:

```
▼ # BarItem()设置
y =[ ]
▼ for k in range(len(cats0)):
▼     y.append(
        # 添加条子设置（颜色、长短）
        opts.BarItem(
            name=cats_list[k],
            value=gdp_list[k],
            itemstyle_opts=opts.ItemStyleOpts(color=cols_
        )
    )

y
```

```
[<pyecharts.options.charts_options.BarItem at 0x20c2ff0
<pyecharts.options.charts_options.BarItem at 0x20c2ff0
<pyecharts.options.charts_options.BarItem at 0x20c2ff0
<pyecharts.options.charts_options.BarItem at 0x20c2ff0
<pyecharts.options.charts_options.BarItem at 0x20c2ff0
<pyecharts.options.charts_options.BarItem at 0x20c2ff0
<pyecharts.options.charts_options.BarItem at 0x20c2ff0
<pyecharts.options.charts_options.BarItem at 0x20c2ff0
<pyecharts.options.charts_options.BarItem at 0x20c2ff0
<pyecharts.options.charts_options.BarItem at 0x20c2ff0
<pyecharts.options.charts_options.BarItem at 0x20c2ff0
<pyecharts.options.charts_options.BarItem at 0x20c2ff0
<pyecharts.options.charts_options.BarItem at 0x20c2ff0
<pyecharts.options.charts_options.BarItem at 0x20c2ff0
<pyecharts.options.charts_options.BarItem at 0x20c2ff0]
```

3: for 循环实现所有年

In [13]:

```
▼ # 先实例化时间轮播图类
tl = Timeline()
```

In [15]:

```
for year in range(1960,2024):
    df_year=pd.DataFrame(df_gdb.loc[df_gdb['年份']==year])
    cats_list= df_year['国家'].to_list()
    cols_list= df_year['颜色'].to_list()
    gdp_list= df_year['GDP'].to_list()
    # BarItem()设置
    y =[ ]
    for k in range(len(cats0)):
        y.append(
            # 添加条子设置（颜色、长短）
            opts.BarItem(
                name=cats_list[k],
                value=gdp_list[k],
                itemstyle_opts=opts.ItemStyleOpts(color=cols_list[k])
            )
        )
    #绘制条形图
    bar=( Bar()
        .add_xaxis(cats_list)
        .add_yaxis("", # 系列名称为空
            y_axis=y,
            category_gap='50%', #调整柱子粗细
            label_opts=opts.LabelOpts(position='right')
        ).reversal_axis()#翻转成条形图
        .set_global_opts(title_opts=opts.TitleOpts(title=f'{year}年GDP分布图',
            subtitle=f'年份: {year}',
            axis_opts=opts.AxisOpts(naming_method='none')
        ))
    # 将bar_year条形图加到 初始化后的tl-时间轴轮播图上
    tl.add(bar, '{}年'.format(year))
    tl.add_schema(play_interval=300,is_loop_play=False)
```

In []:

```
tl.render_notebook()
```

In [16]:

```
tl.render(r'D:\! 江科2023\01教学\6-数据分析实验2022/世界GDP动态分析
```

```
'D:\\! 江科2023\\01教学\\6-数据分析实验2022\\世界GDP动态分析
```

In []: