

轻松用 Seaborn 进行数据可视化

Seaborn 是一个数据可视化库，可帮助在Python中创建有趣的数据可视化。大多数数据分析需要识别趋势和建立模型。本文将帮助您开始使用 Seaborn库创建数据可视化。

Seaborn 是一个出色的数据可视化库，它让我们的生活变得轻松。首先，您应该在 jupyter notebook 中键入以下命令。

In [1]:

```
import pandas as pd # Pandas
import numpy as np # Numpy
import matplotlib.pyplot as plt # Matplotliblibrary
import seaborn as sns # Seaborn Library
%matplotlib inline
sns.set()
```

直方图（Distplot）

sns.distplot () 结合直方图并绘制核密度估计图。这里 bin 区间大小是自动计算的。

```
$$ sns.distplot(data[“variablename”]) $$
```

我们将使用以下代码在 jupyter notebook 中加载数据集。

In [2]:

```
# Load the Dataset in Python
tips = sns.load_dataset("tips")
tips.head()
```

Out[2]:

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

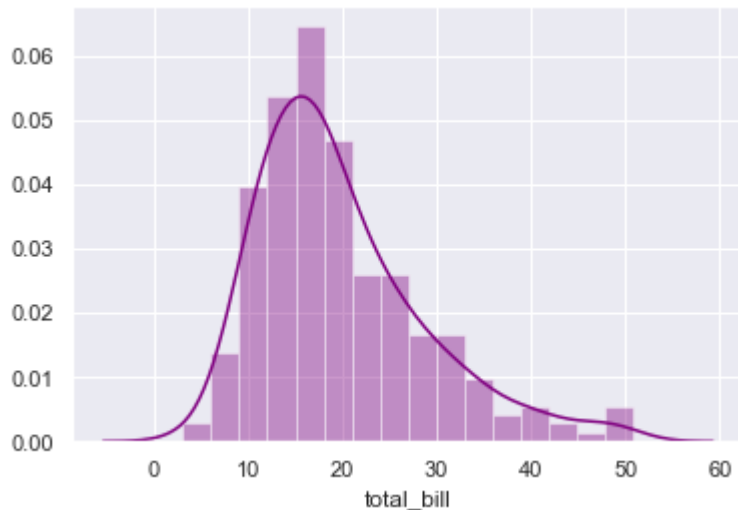
现在，由于我们已经加载了数据集，我们将使用 “total_bill” 变量创建第一个图。让我们从 tips数据集创建 “total_bill” 变量的 distplot。

In [3]:

```
sns.distplot(tips["total_bill"], bins=16, color="purple")  
# Binsize is calculated using square-root of row count.
```

Out[3]:

<matplotlib.axes._subplots.AxesSubplot at 0x1a21fc43d0>



现在，我们来对上述代码进行进一步描述：

- **sns.distplot**—这个命令将启动 distplot 的初始创建
- **tips["total_bill"]**—从 tips 数据集（数据框）中取出列（total_bill）。在这里，我们应该观察一下，可以使用方括号来拉取列值，并且列名应该用引号括起来（双引号/单引号）都被接受。

我们推测—“total_bill”变量本质上是倾斜的，大多数帐单值都在 \ \$ 10 - \ \$ 20 范围内。

联合分布图（Jointplot）

联合分布图（Jointplot）采用两个变量并一起创建直方图和散点图。让我们看一下 jointplot 的语法。

```
$$sns.jointplot(x = , y =, data=)$$
```

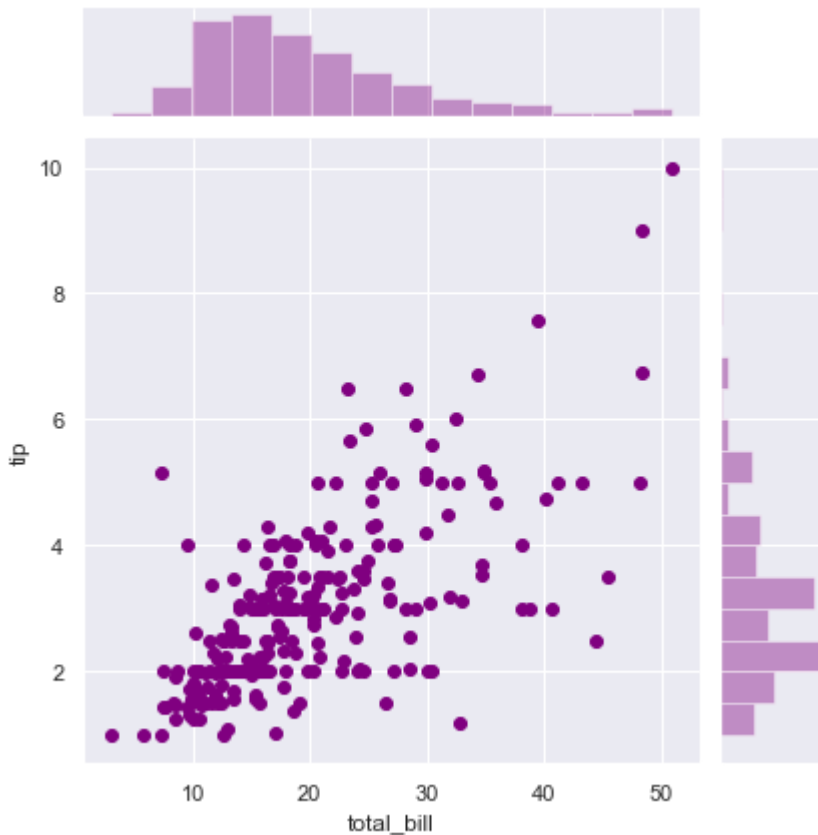
让我们从 tips 数据集创建 total_bill 和 tip 变量的联合分布图。通常，任何餐厅的小费金额取决于总账单/账单大小。让我们看看这个情景下是什么样的。代码如下：

In [4]:

```
sns.jointplot(x = "total_bill", y = "tip", data = tips, color="purple")
```

Out[4]:

<seaborn.axisgrid.JointGrid at 0x1a230c0890>



如上所述，散点图似乎显示总账单和小费金额之间的强相关性。在它的顶部，我们可以看到各个变量的直方图。

Jointplot :: kind = "hex"

直方图的双变量类比称为“hexbin”图，因为它显示了六边形区间内的观察计数。此图对于相对较大的数据集最有效。也称为Hexbin Plots。

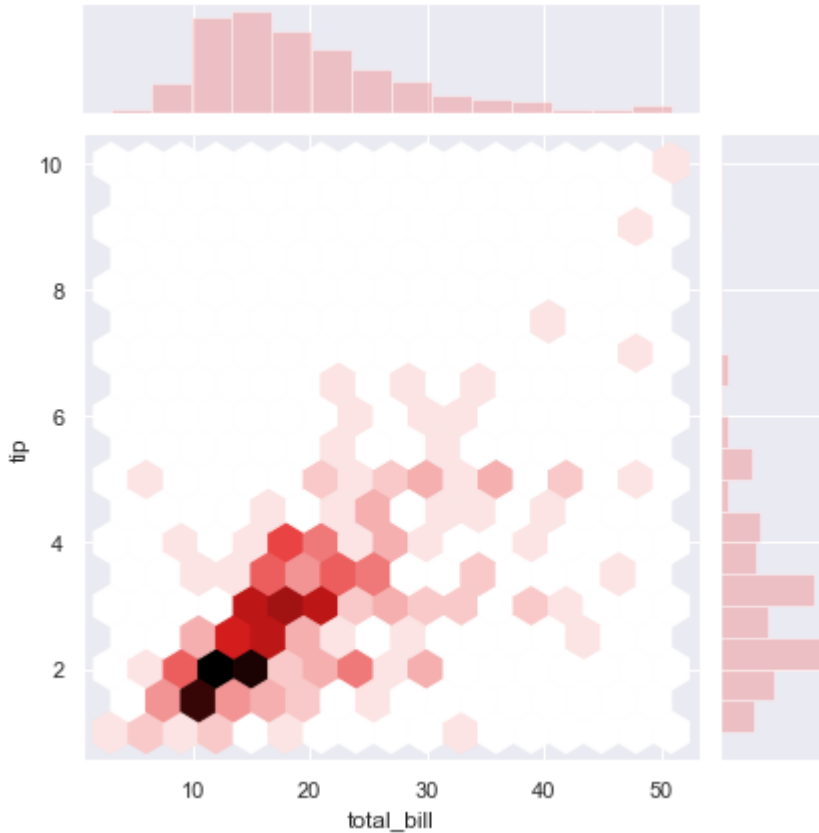
```
$$sns.jointplot(x = , y =, data=, kind="hex")$$
```

In [5]:

```
# Jointplot - Scatterplot and Histogram  
sns.jointplot(x = "total_bill", y = "tip", data = tips, kind = "hex",  
color="lightcoral")
```

Out[5]:

<seaborn.axisgrid.JointGrid at 0x1a2326a250>



有几种类型的值可以放在 `sns.jointplot` 中来创建不同的图。默认情况下，联合分布图显示散点图。现在，在上面的情节图中，它显示了六边形。六边形的深色表示数据点的高密度，其中较浅的颜色表示较少的点。

`kind` 参数值可以是以下取值：

```
kind : { "scatter" | "reg" | "resid" | "kde" | "hex" }
```

下面，我们来看看 `kind="kde"` 的情形。

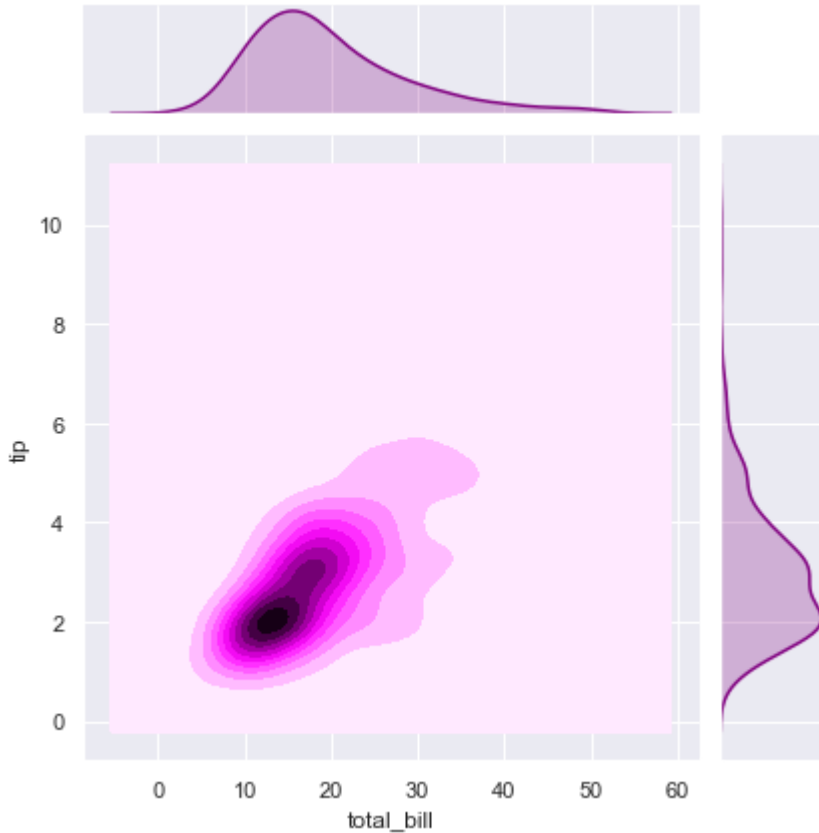
Jointplot :: kind ="kde"

In [6]:

```
# Jointplot - Scatterplot and Histogram
sns.jointplot(x = tips["total_bill"], y = tips["tip"], kind = "kde",
color="purple") # contour plot
```

Out[6]:

<seaborn.axisgrid.JointGrid at 0x1a2354d850>



上面显示的图表称为轮廓图。轮廓图（有时称为“水平图”）是一种在二维平面上显示三维表面的方法。它绘制了y轴上的两个预测变量X Y和轮廓的响应变量Z。

矩阵图（Pairplot）

矩阵图基本上绘制了变量之间的成对关系。它支持用“hue”来为类别变量绘图着色。

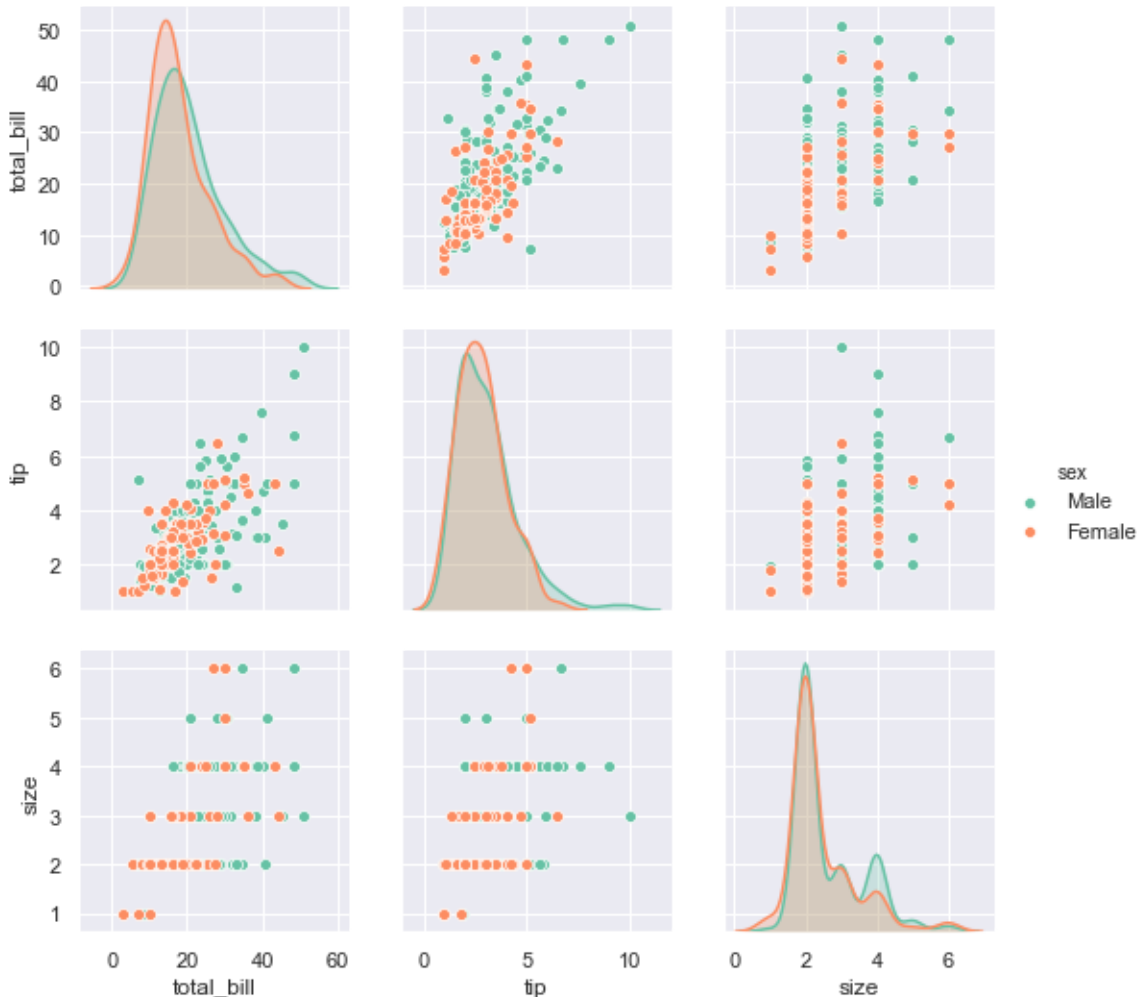
```
$$$sns.pairplot("dataframe")$$
```

In [7]:

```
# Pairplot of Tips
sns.pairplot(tips, hue = "sex", palette="Set2")
# this will color the plot gender wise
```

Out[7]:

<seaborn.axisgrid.PairGrid at 0x1a23804490>



下面我们来了解下矩阵图的含义。 对角线部分显示了具有核密度估计的 distplot图或直方图。 矩阵图的上部和下部显示散点图。“hue”使用列的类别为绘图着色。

hue = “sex”— 设置为按不同的性别进行着色

palette = “Set2” - “Set2” 是颜色的一个系列。

条形图（Barplot）

条形图用于绘制分类列和数字列。 它在可视化中创建了条形。 让我们用“性别”创建一个“total_bill”的条形图，让我们看看哪类人支付更多。

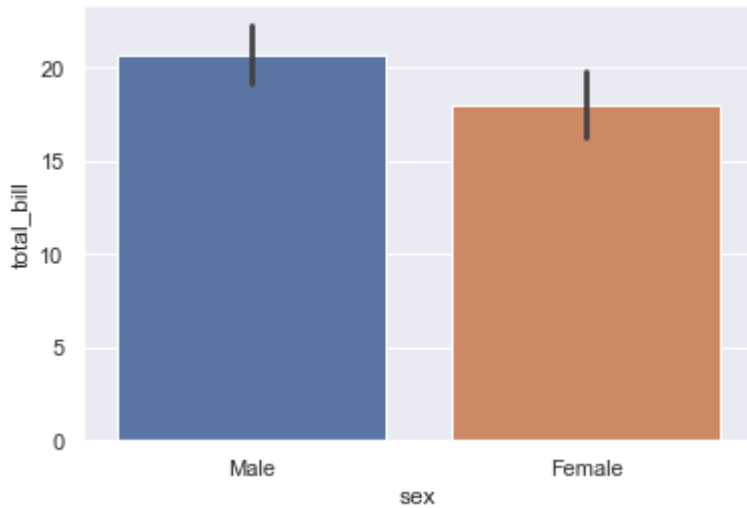
```
$$sns.barplot(x = , y =, data=)$$
```

In [8]:

```
# Barplot
sns.barplot(x="sex", y="total_bill", data=tips)
# Inference - Total Bill Amount for males is more than Females.
```

Out[8]:

<matplotlib.axes._subplots.AxesSubplot at 0x1a23d93b90>

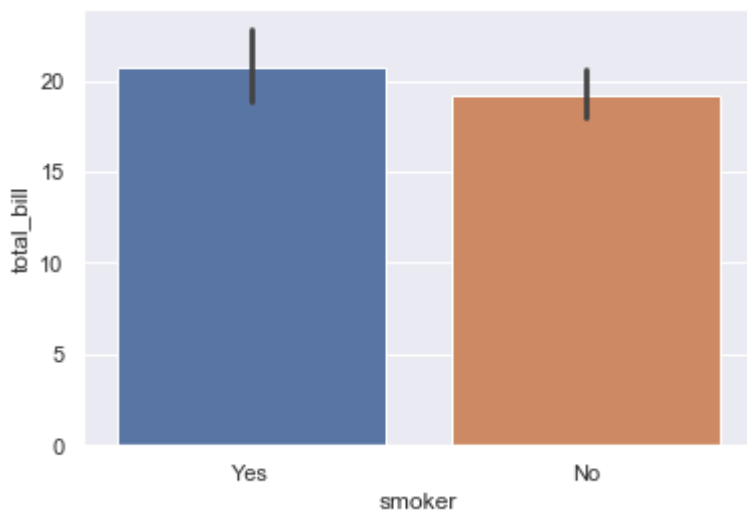


In [9]:

```
# Lets Plot Smoker Vs Total Bill :: The purpose is to find out if
# Smokers pay more bill than Non Smokers
sns.barplot(x="smoker", y="total_bill", data=tips)
# Inference - More Bill for Smokers
```

Out[9]:

<matplotlib.axes._subplots.AxesSubplot at 0x1a23ff2990>

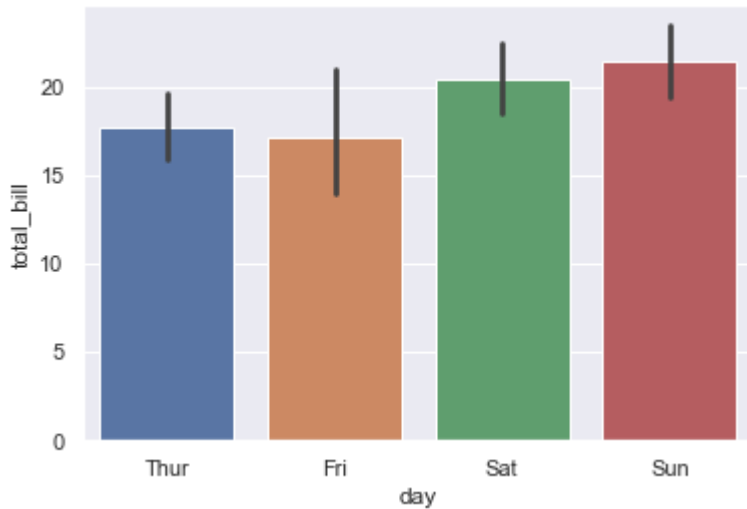


In [10]:

```
# Lets Find If There is more Bill In Weekend or Weekdays
sns.barplot(x = "day", y = "total_bill", data =tips)
# People tend to visit more on weekends
```

Out[10]:

<matplotlib.axes._subplots.AxesSubplot at 0x1a241cfa90>



箱形图 (Boxplot)

箱形图 (Boxplot) 是给定数据集的五点汇总统计的直观表示。五个数字摘要包括：

- Minimum 最小值
- First Quartile 1/4 值
- Median (Second Quartile) 中位数
- Third Quartile 3/4 值
- Maximum 最大值

此外，值得注意的是，为分类 - 连续变量创建了一个箱线图，这意味着如果x轴是分类的并且y轴是连续的，则应创建箱线图或小提琴图。

让我们从 tips数据集创建一个“day”和“total_bill”的箱线图。

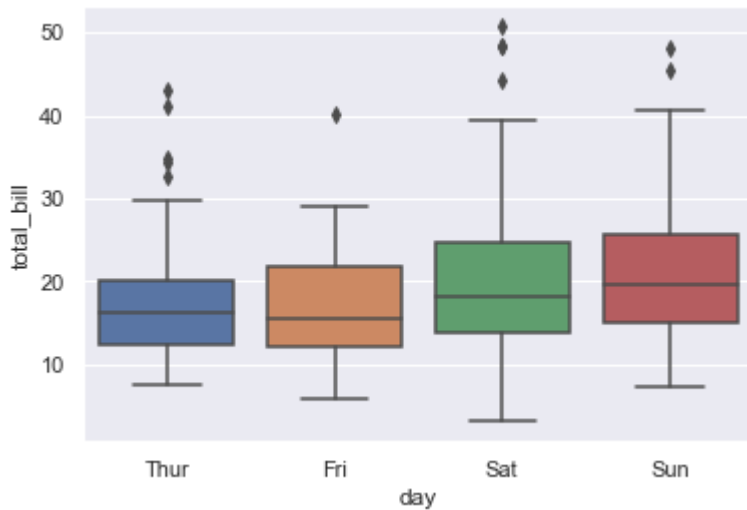
```
$$sns.boxplot(x = , y =, data=)$$
```


In [11]:

```
# Boxplot
sns.boxplot(x = "day", y = "total_bill", data=tips)
```

Out[11]:

<matplotlib.axes._subplots.AxesSubplot at 0x1a243f34d0>

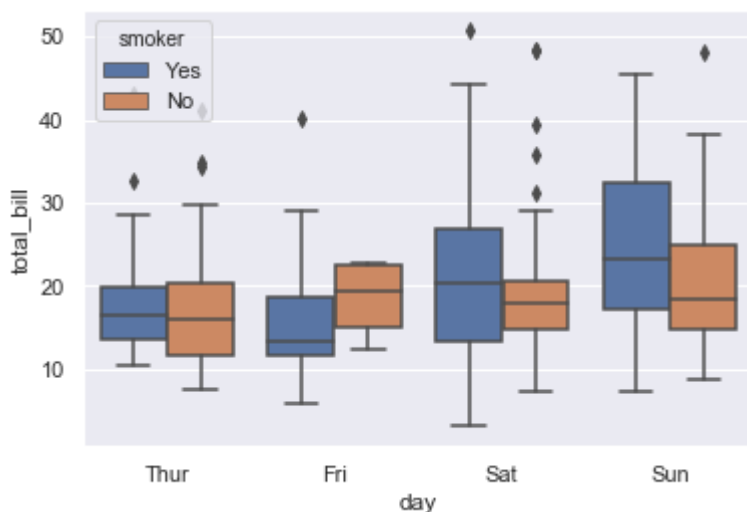


In [12]:

```
# Add hue to split the barplot. Making it more fancier
sns.boxplot(x = "day", y = "total_bill", data=tips, hue = "smoker")
# On Friday people have more bill if they are a Non smoker vs smoker
```

Out[12]:

<matplotlib.axes._subplots.AxesSubplot at 0x1a244d2c50>



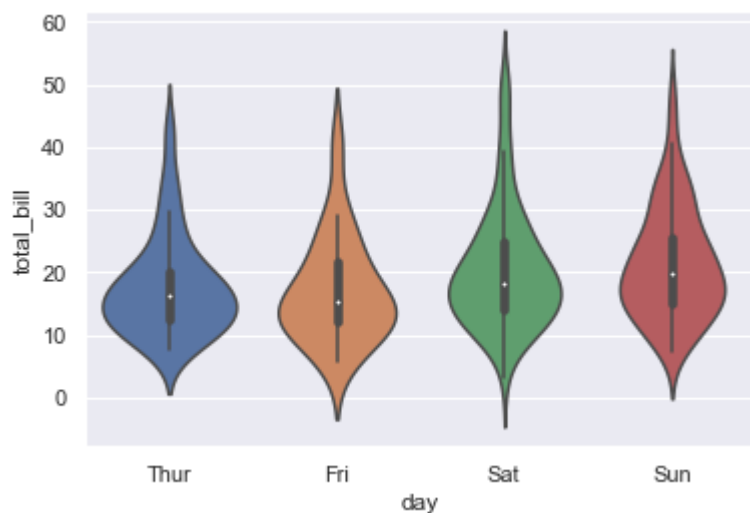
hue = "smoker": - 它为吸烟者和非吸烟者创造了一个箱线图。例如：在星期五的情况下，可以清楚地看到，与当天的吸烟者相比，非吸烟者的食物费用更多。

In [13]:

```
# Violin Plots
sns.violinplot(x = "day", y = "total_bill", data = tips)
```

Out[13]:

<matplotlib.axes._subplots.AxesSubplot at 0x1a24713090>



小提琴图跟箱形图有些类似。他们之间的描述可以参考下面的图示内容：

LM Plot

sns.lmplot 是绘制一个在散点图中进行线性回归拟合的直线。它遵循普通的最小二乘法，线代表最佳拟合线。为了更好地理解这一点，建议进一步了解一下线性回归的相关知识。

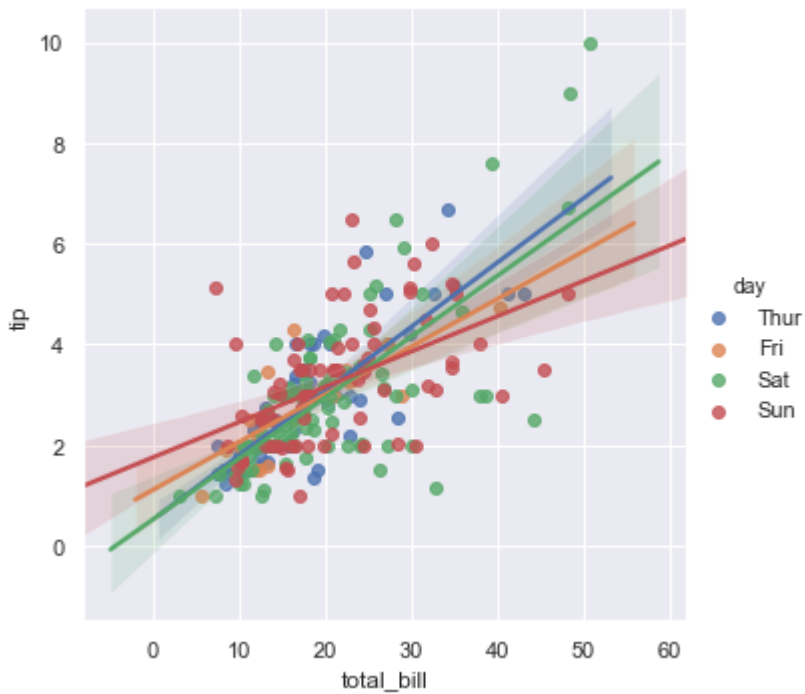
代码如下：

In [14]:

```
# LM Plot  
sns.lmplot(x = "total_bill", y = "tip", data = tips, hue="day")
```

Out[14]:

<seaborn.axisgrid.FacetGrid at 0x1a247cbc10>



上图显示了不同日期的total_bill变量的线性回归拟合，如图例中所示，这是在 sns.lmplot 中使用 hue =“day” 获得的。

恭喜！您已完成 Seaborn 初学者教程。希望这篇文章能够提供有关 Seaborn 的基本知识，并且可以帮助您创建所有这些图。

In []: