In [1]:

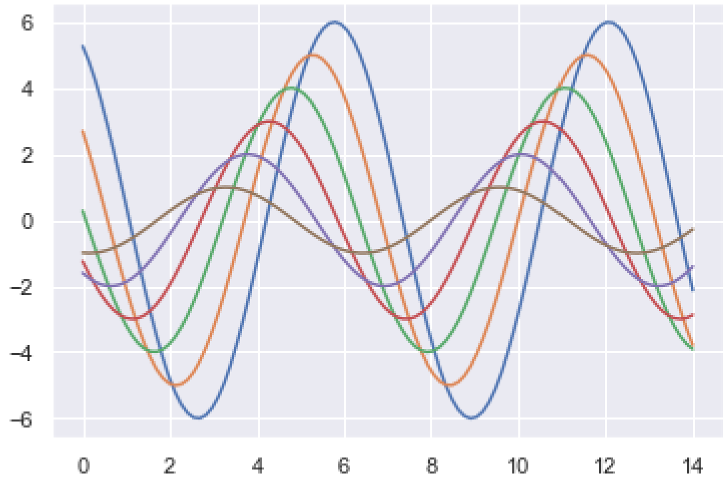
```
import seaborn as sns
import numpy as np
import matplotlib as mpl
import matplotlib.pylab as plt
%matplotlib inline
```

In [5]:

```
def sinplot(flip=1):
    x = np.linspace(0, 14, 100)
    for i in range(1, 7):
        plt.plot(x, np.cos(x + i * .5) * (7 - i) * flip)
```

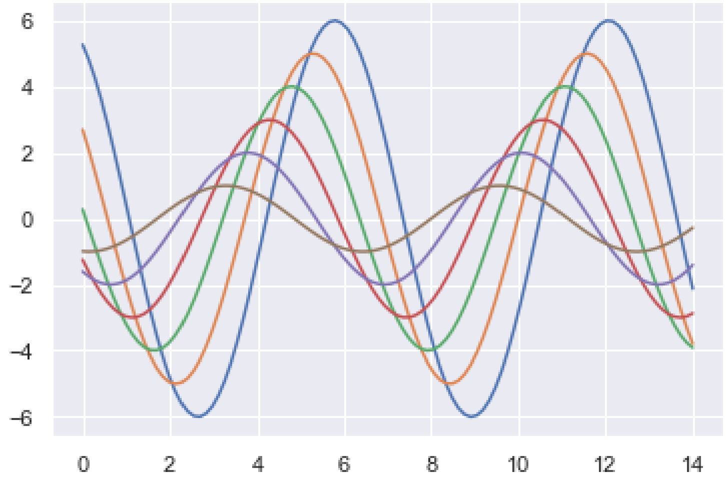
In [6]:

```
sinplot()
```



In [7]:

```
sns.set()
sinplot()
```

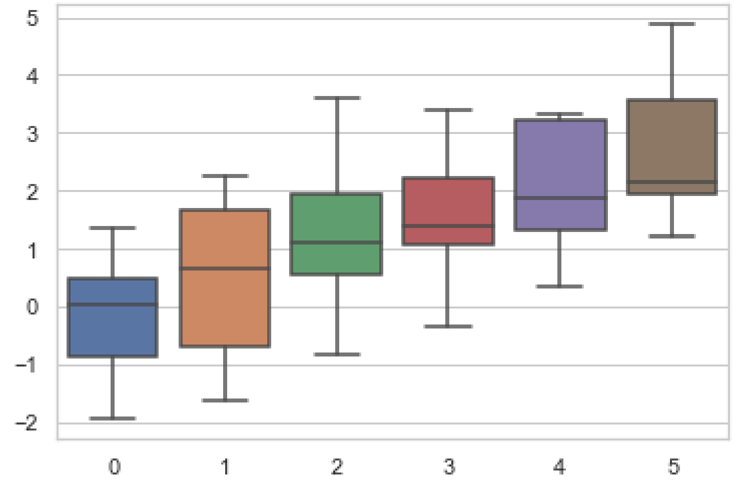


In [8]:

```
sns.set_style("whitegrid")
data = np.random.normal(size=(20, 6)) + np.arange(6) / 2
sns.boxplot(data=data)
```

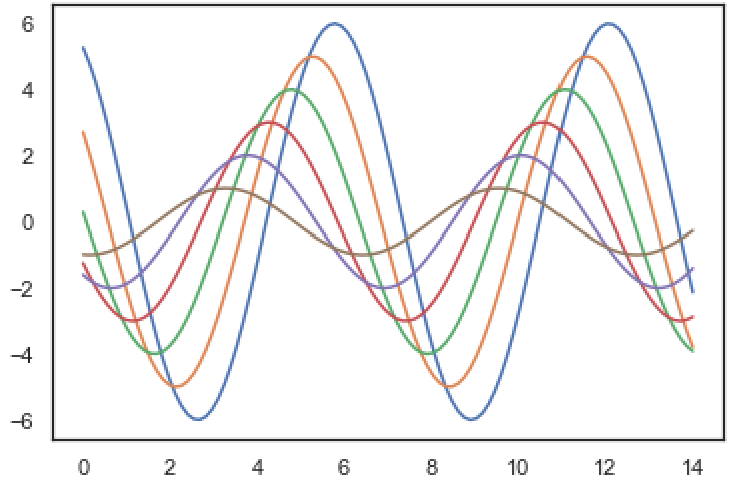
Out[8]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x147d5f7e5c0>
```



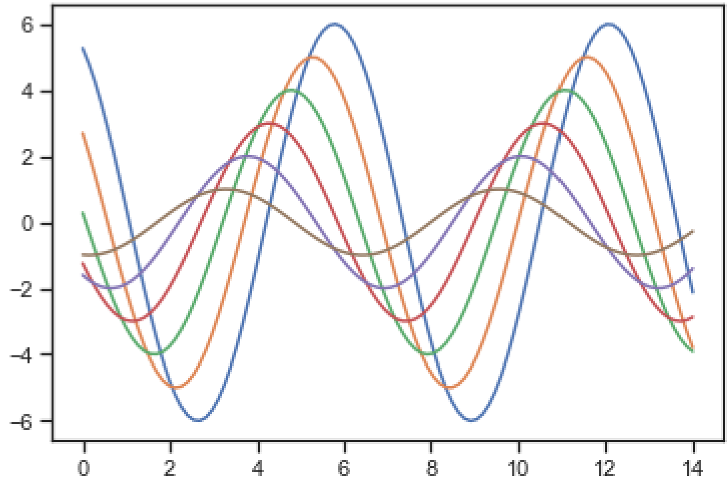
In [9]:

```
sns.set_style("white")
sinplot()
```



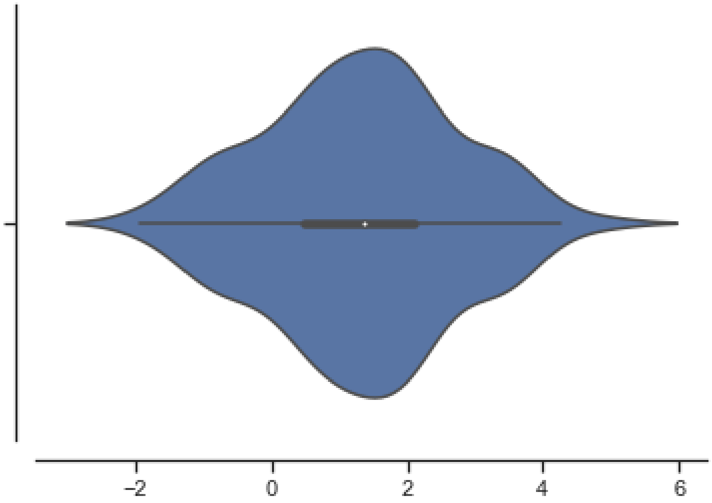
In [10]:

```
sns.set_style("ticks")
sinplot()
```



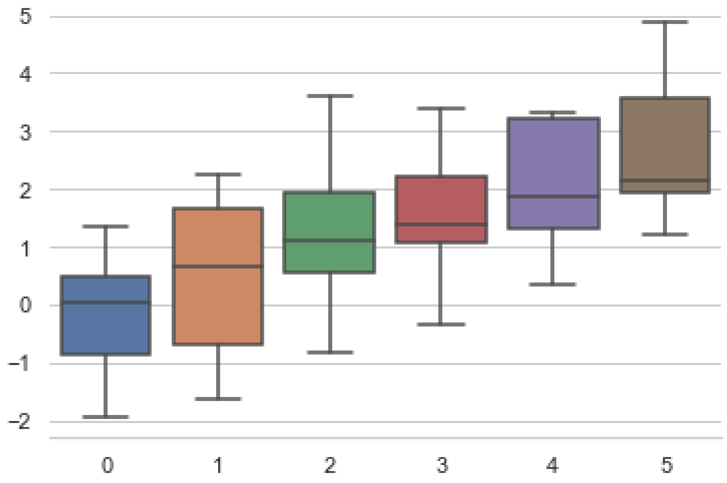
In [15]:

```
sns.violinplot(data)
sns.despine(offset=10)
```



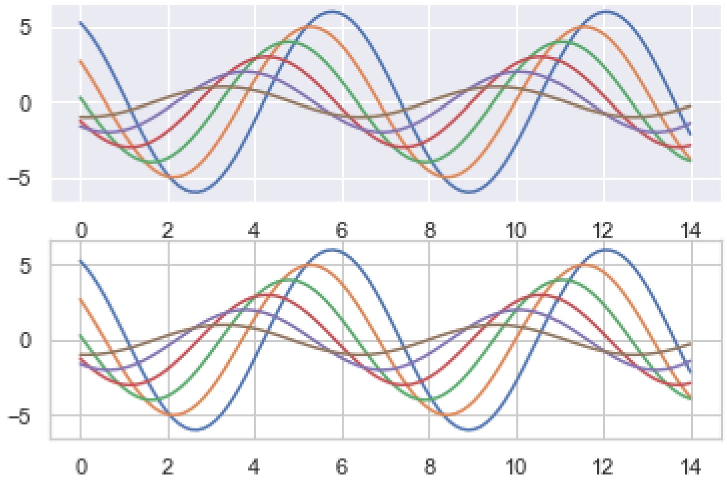
In [16]:

```
sns.set_style("whitegrid")
sns.boxplot(data=data, palette="deep")
sns.despine(left=True)
```



In [17]:

```
with sns.axes_style("darkgrid"):
    plt.subplot(211)
    sinplot()
plt.subplot(212)
sinplot()
```

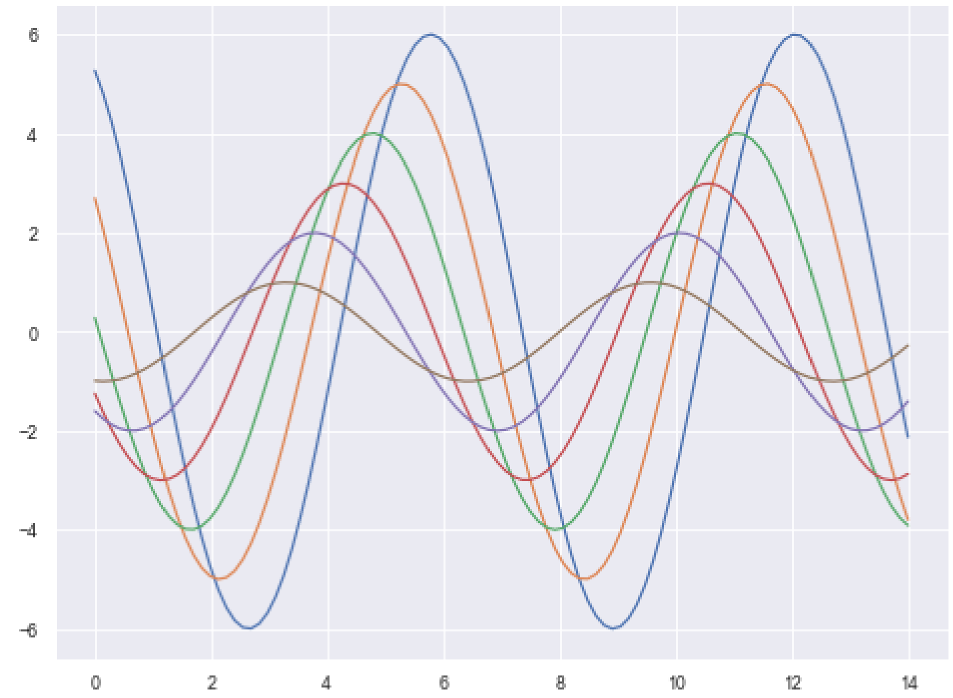


In [18]:

```
sns.set()
```

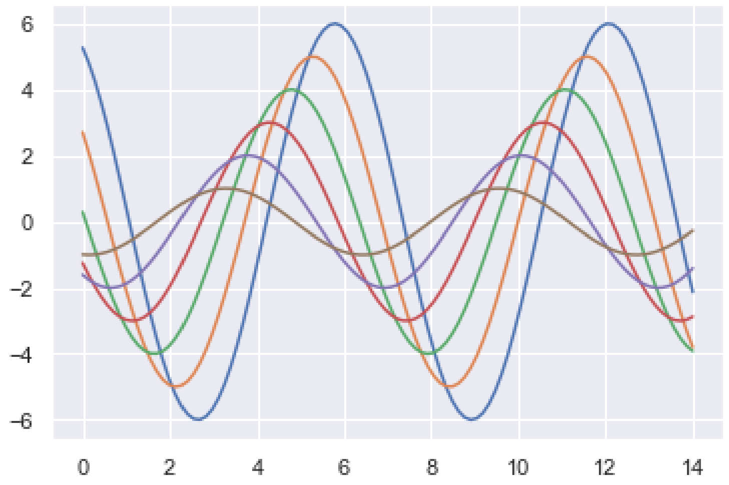
In [19]:

```
sns.set_context("paper")
plt.figure(figsize=(8, 6))
sinplot()
```



In [21]:

```
sns.set_context("notebook", font_scale=1, rc={"line.linewidth":2.5})
sinplot()
```

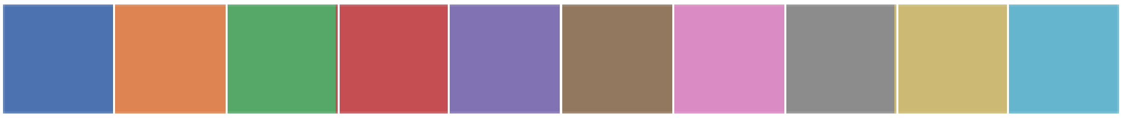


In [22]:

```
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
sns.set(rc={"figure.figsize": (6, 6)})
```

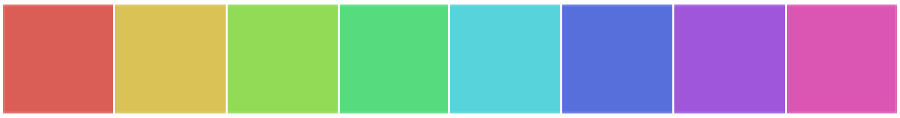
In [23]:

```
current_palette = sns.color_palette()
sns.palplot(current_palette)
```



In [27]:

```
sns.palplot(sns.color_palette("hls",8))
```



In [32]:

```
data = np.random.normal(size=(20, 6)) + np.arange(6) / 2
sns.boxplot(data=data,palette==sns.color_palette("hls",8))
```

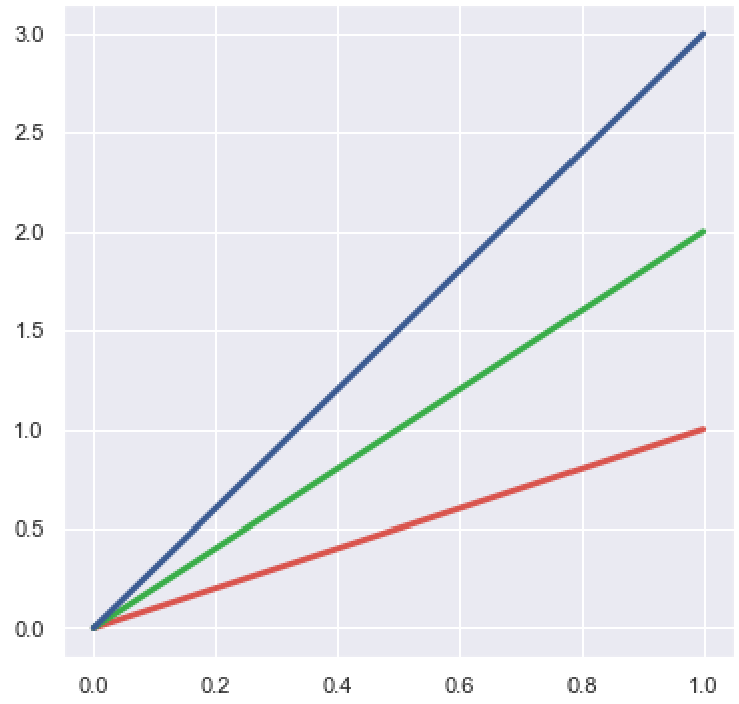
```
  File "<ipython-input-32-f2c3cfe42ebe>", line 2
    sns.boxplot(data=data,palette==sns.color_palette("hls",8))
                         ^
SyntaxError: positional argument follows keyword argument
```

In [36]:

```
plt.plot([0, 1], [0,1], sns.xkcd_rgb["pale red"], lw=3)
plt.plot([0, 1], [0,2], sns.xkcd_rgb["medium green"], lw=3)
plt.plot([0, 1], [0,3], sns.xkcd_rgb["denim blue"], lw=3)
```

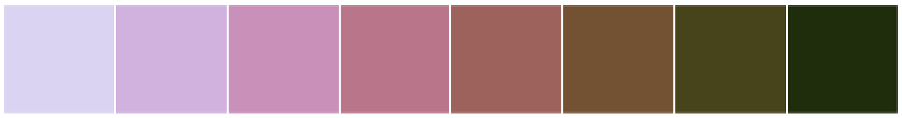
Out[36]:

```
[<matplotlib.lines.Line2D at 0x147d97a2c18>]
```



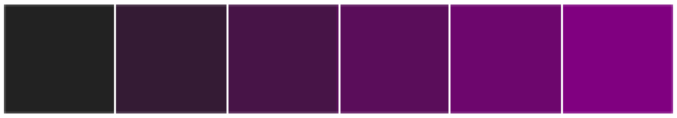
In [37]:

```
sns.palplot(sns.cubehelix_palette(8, start=5, rot=-.75))
```



In [38]:

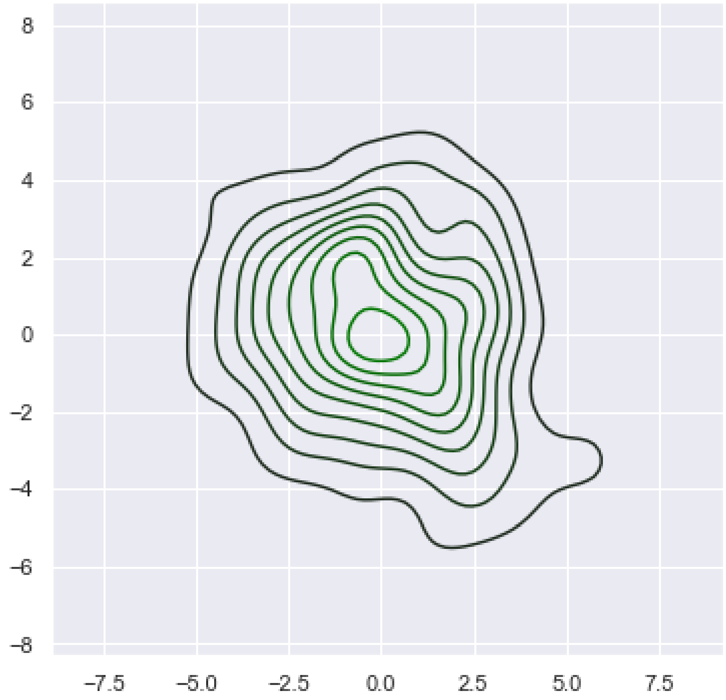
```
sns.palplot(sns.dark_palette("purple"))
```



In [41]:

```
x, y =np.random.multivariate_normal([0,0],[[1,-5],[-5,1]],size=300).T
pal = sns.dark_palette("green", as_cmap=True)
sns.kdeplot(x,y,cmap=pal);
```

```
D:\Anaconda\ancaonda\lib\site-packages\ipykernel_launcher.py:1: RuntimeWarning: cova
riance is not symmetric positive-semidefinite.
  """Entry point for launching an IPython kernel.
```

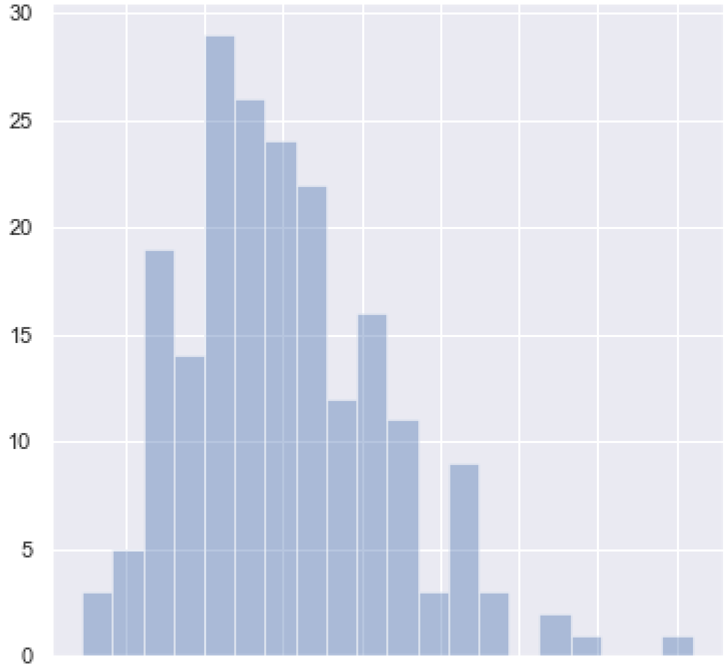


In [43]:

```
x = np.random.normal(size=100)
sns.distplot(x,kde=False)
```

In [48]:

```
sns.distplot(x, bins=20, kde=False)
```



In [51]:

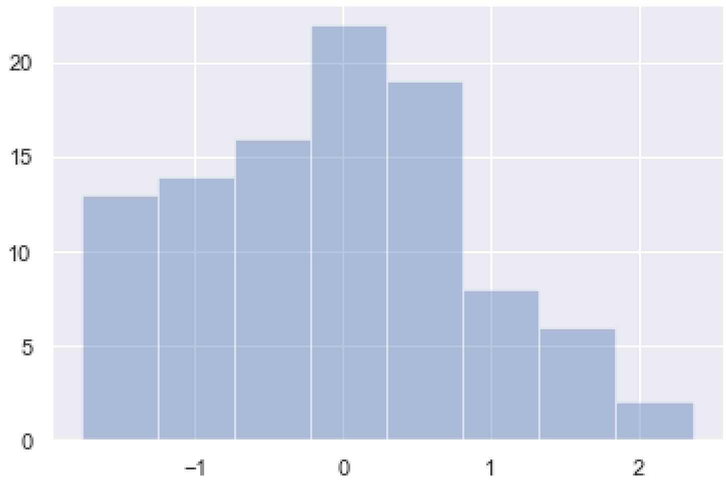
```
%matplotlib inline
import numpy as np
import pandas as pd
from scipy import stats, integrate
import matplotlib.pyplot as plt

import seaborn as sns
sns.set(color_codes=True)
np.random.seed(sum(map(ord, "distributions")))
```

In [53]:

```
x = np.random.normal(size=100)
sns.distplot(x,kde=False)
```

Out[53]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x147db0cd0f0>
```



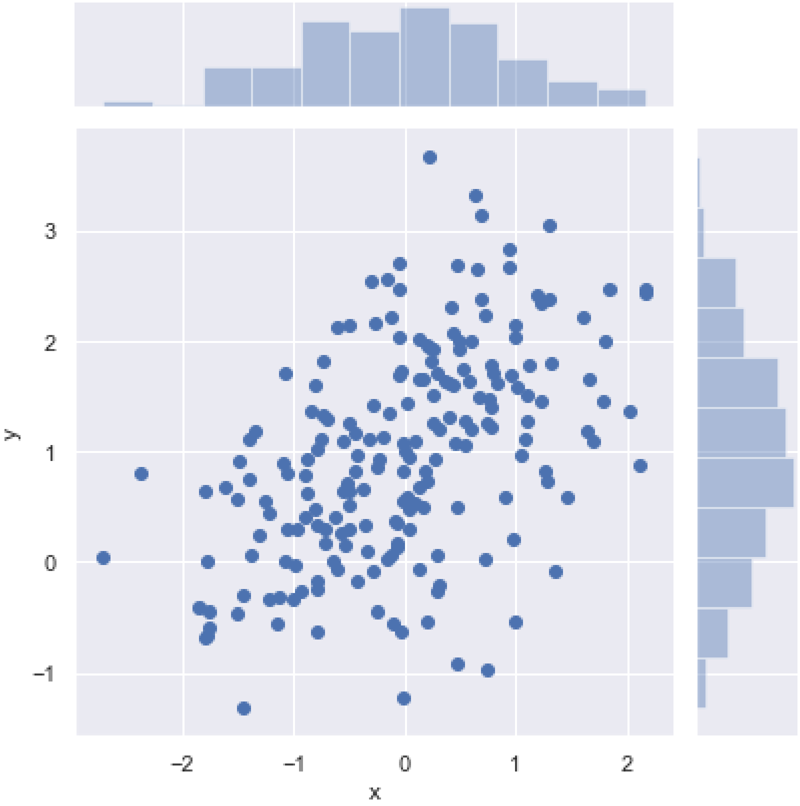
In [57]:

```
mean, cov = [0, 1], [(1, .5), (.5,1)]
data = np.random.multivariate_normal(mean, cov,200)
df = pd.DataFrame(data, columns=["x","y"])
df
```

```
180  -0.870530   0.937931
181   0.912959   0.590534
182  -0.073455   0.161298
183  -0.928615  -0.260290
184   0.834092   1.613593
185   0.550806   1.281319
186  -0.491134   2.150730
187  -0.286326   1.429608
188   0.316406  -0.212208
189   0.745665   1.261303
190   0.459643   1.074581
191  -1.386694   0.057915
192   0.606581   2.004781
```

In [59]:

```
sns.jointplot(x="x", y="y", data=df);
```



In [63]:

```
x, y =np.random.multivariate_normal(mean, cov, 1000).
with sns.axes_style("white"):
    sns.jointplot(x=x,y=y,kind="hex", color="k")
```

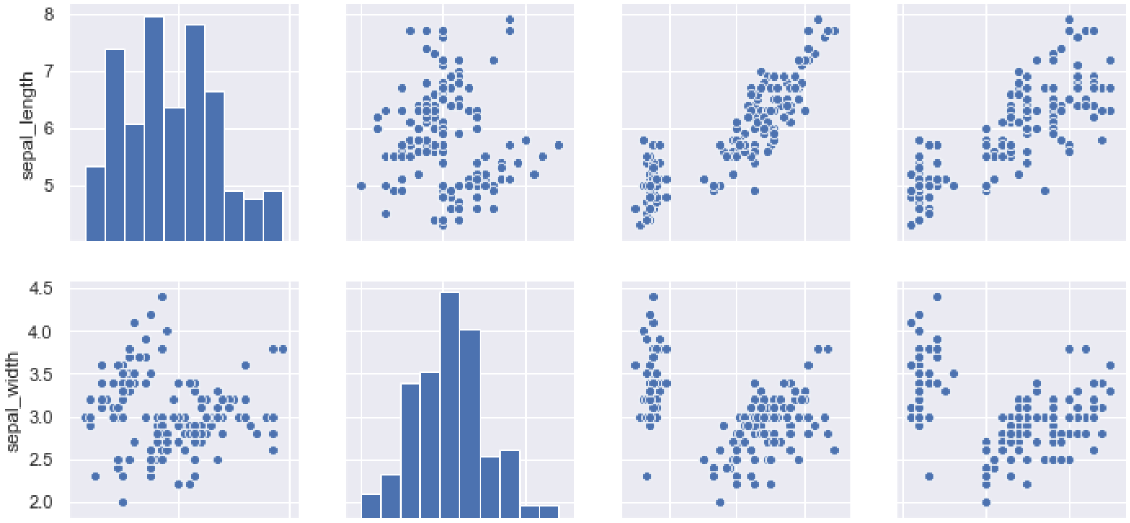
```
  File "<ipython-input-63-773a00d3f207>", line 1
    x, y =np.random.multivariate_normal(mean, cov, 1000).
                                                         ^

SyntaxError: invalid syntax
```

In [64]:

```
iris = sns.load_dataset("iris")
sns.pairplot(iris)
```

Out[64]:

```
<seaborn.axisgrid.PairGrid at 0x147dc7c9ba8>
```



In [65]:

```
%matplotlib inline
import numpy as np
import pandas as pd
import matplotlib as mpl
import matplotlib.pyplot as plt

import seaborn as sns
sns.set(color_codes=True)
np.random.seed(sum(map(ord, "regression")))

tips = sns.load_dataset("tips")

tips.head()
```

Out[65]:

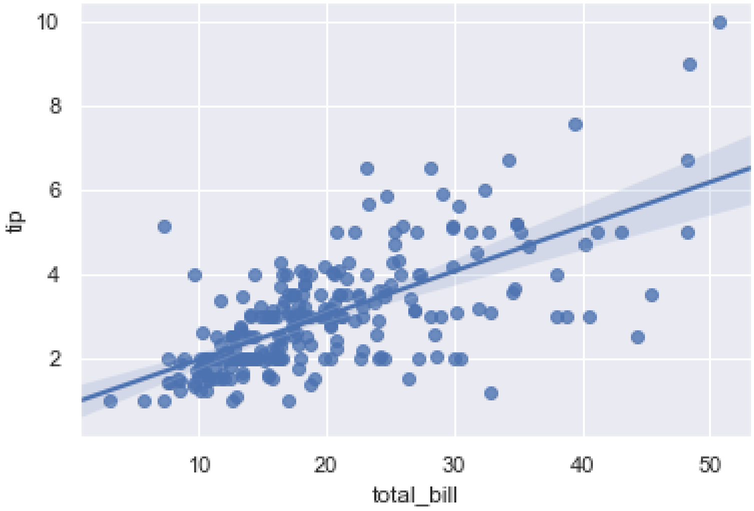
	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

In [67]:

```
sns.regplot(x="total_bill", y ="tip", data=tips)
```

Out[67]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x147ddd78860>
```

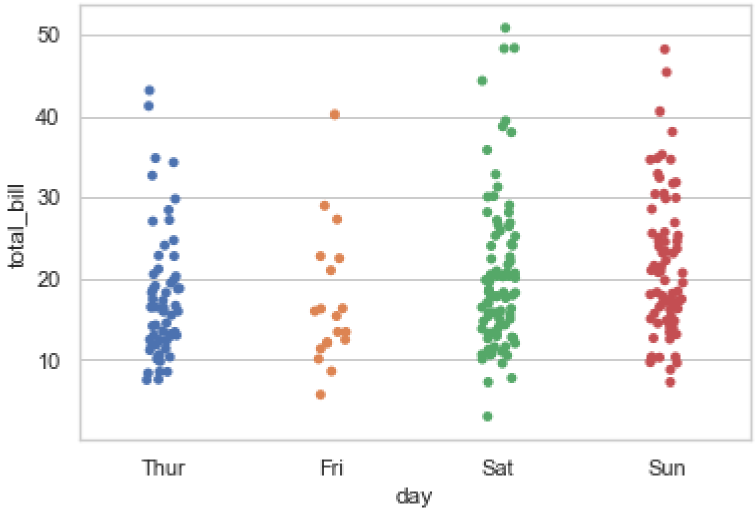


In [68]:

```
%matplotlib inline
import numpy as np
import pandas as pd
import matplotlib as mpl
import matplotlib.pyplot as plt

import seaborn as sns
sns.set(style="whitegrid", color_codes=True)

np.random.seed(sum(map(ord, "categorical")))
titanic = sns.load_dataset("titanic")
tips = sns.load_dataset("tips")
iris = sns.load_dataset("iris")
```

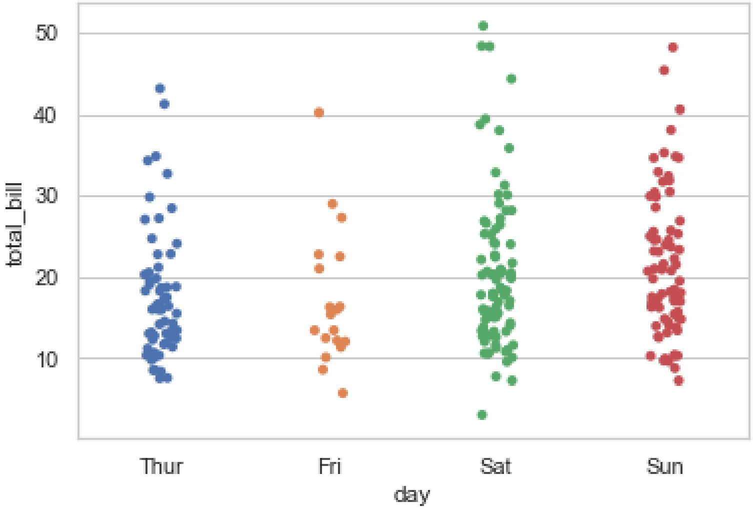
In [72]:

```
sns.stripplot(x="day", y="total_bill", data=tips);
```



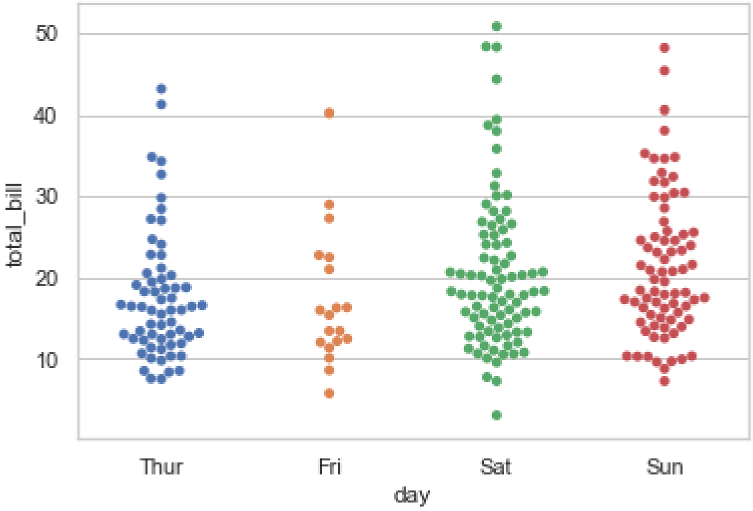
In [73]:

```
sns.stripplot(x="day", y="total_bill", data=tips, jitter=True);
```



In [74]:

```
sns.swarmplot(x="day", y="total_bill", data=tips);
```

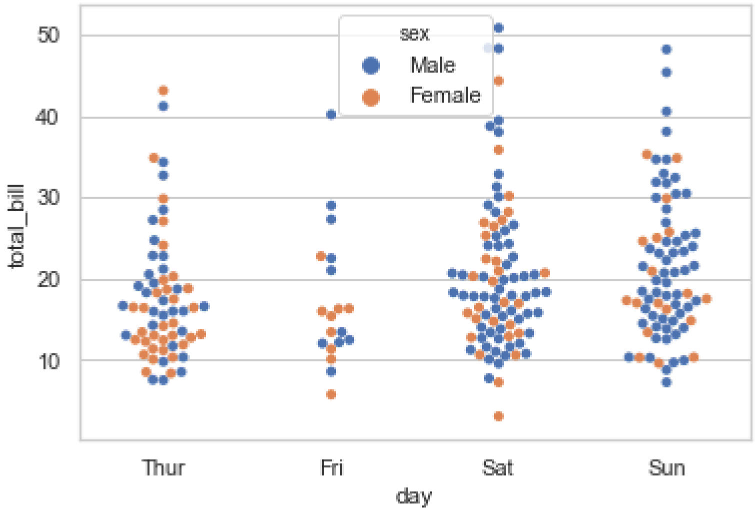


In [76]:

```
sns.swarmplot(x="day", y="total_bill",hue="sex", data=tips)
```

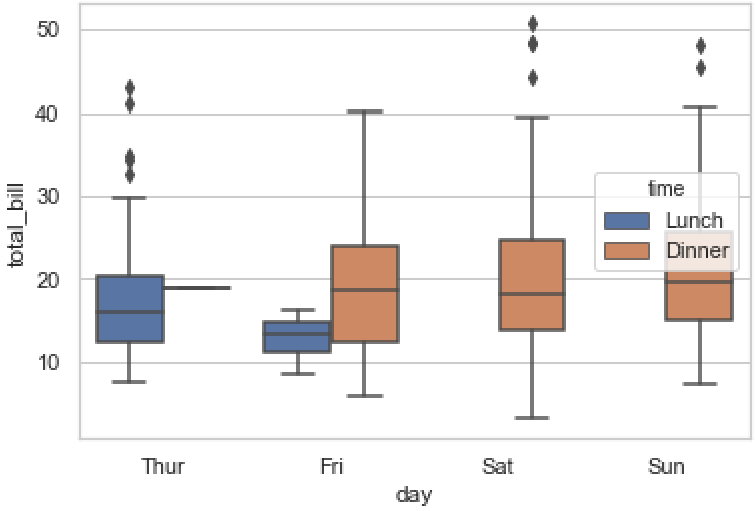
Out[76]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x147dbbdfd68>
```



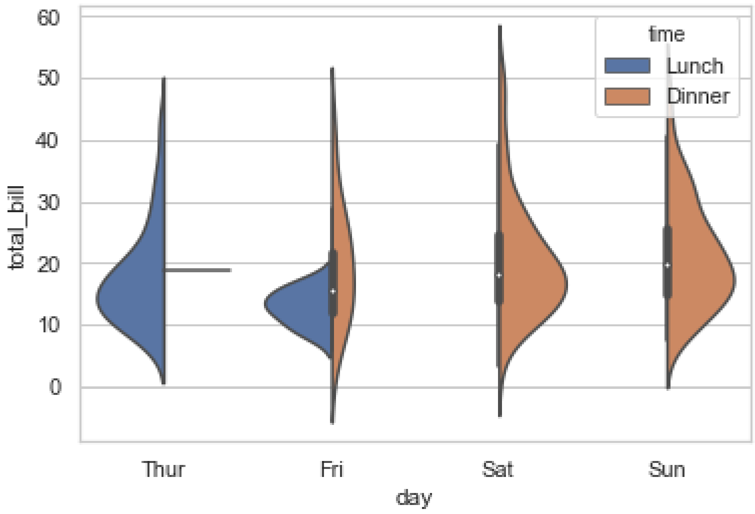
In [79]:

```
sns.boxplot(x="day", y="total_bill", hue="time", data=tips);
```



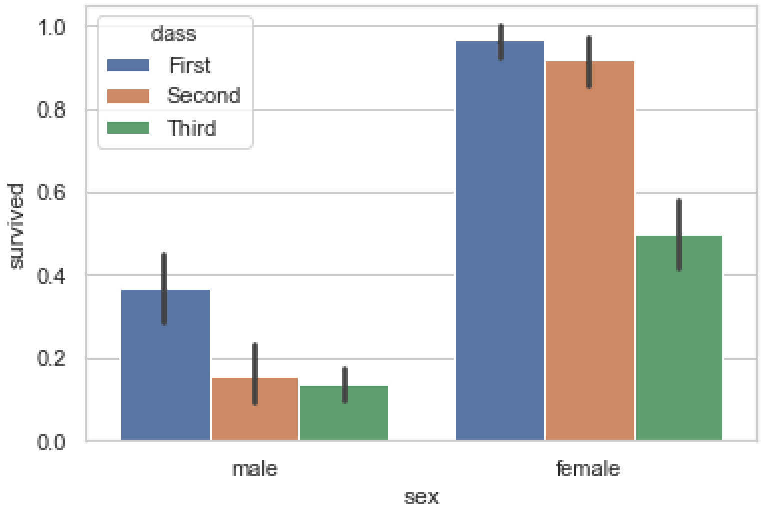
In [84]:

```
sns.violinplot(x="day", y="total_bill",hue="time", data=tips, split="True");
```



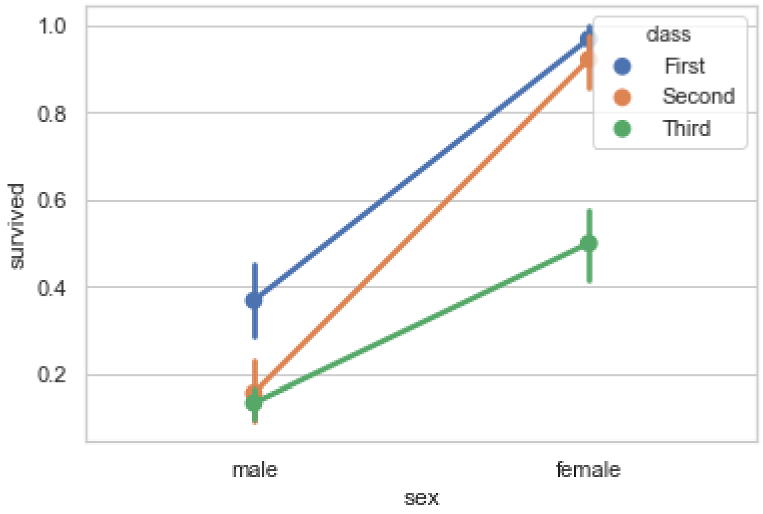
In [91]:

```
sns.barplot(x="sex", y="survived", hue="class",data=titanic);
```



In [93]:

```
sns.pointplot(x="sex", y="survived", hue="class",data=titanic);
```



In [103]:

```
sns.pointplot(x="sex", y="survived", hue="class",data=titanic,
             palette={"male":"g", "female":"m"},
             markers=["^", "o"], linestyles=["-", "--"]);
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-103-a92c175e67ee> in <module>
      1 sns.pointplot(x="sex", y="survived", hue="class",data=titanic,
----> 2              palette={"male":"g", "female":"m"});

D:\Anaconda\ancaonda\lib\site-packages\seaborn\categorical.py in pointplot(x,
 y, hue, data, order, hue_order, estimator, ci, n_boot, units, markers, linesty
les, dodge, join, scale, orient, color, palette, errwidth, capsize, ax, **kwarg
s)
   3333                             estimator, ci, n_boot, units,
   3334                             markers, linestyles, dodge, join, scale,
-> 3335                             orient, color, palette, errwidth, capsize)
   3336 
   3337     if ax is None:

D:\Anaconda\ancaonda\lib\site-packages\seaborn\categorical.py in __init__(self,
x, y, hue, data, order, hue_order, estimator, ci, n_boot, units, markers, lines
tyles, dodge, join, scale, orient, color, palette, errwidth, capsize)
```

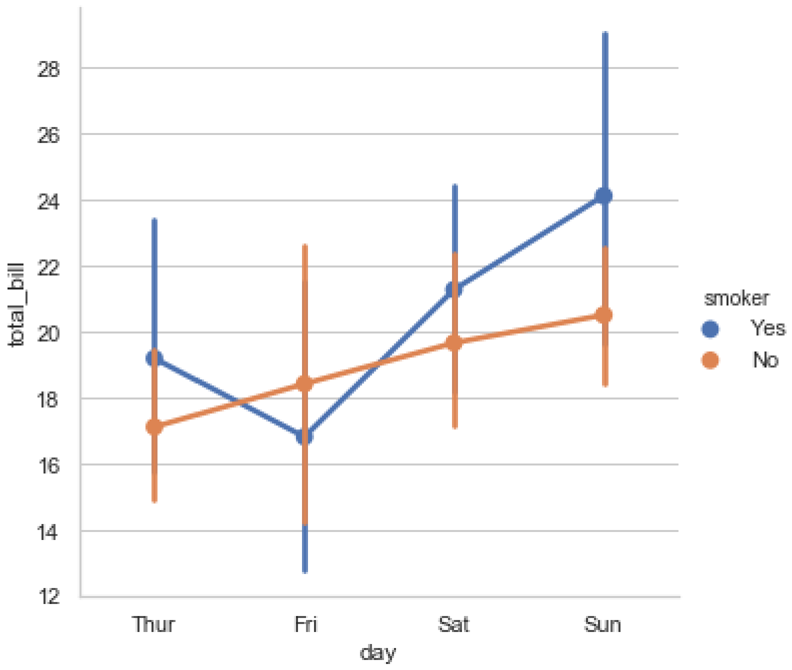
In [111]:

```
sns.factorplot(x="day", y="total_bill", hue="smoker", data=tips)
```

```
D:\Anaconda\ancaonda\lib\site-packages\seaborn\categorical.py:3666: UserWarning: The
`factorplot` function has been renamed to `catplot`. The original name will be remov
ed in a future release. Please update your code. Note that the default `kind` in `fa
ctorplot` (`'point'`) has changed `'strip'` in `catplot`.
  warnings.warn(msg)
```

Out[111]:

```
<seaborn.axisgrid.FacetGrid at 0x147de2c6940>
```



In [112]:

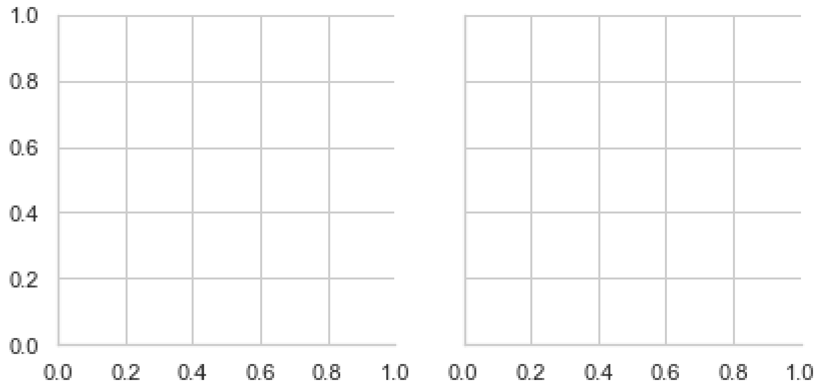
```
tips = sns.load_dataset("tips")
tips.head()
```

Out[112]:

	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

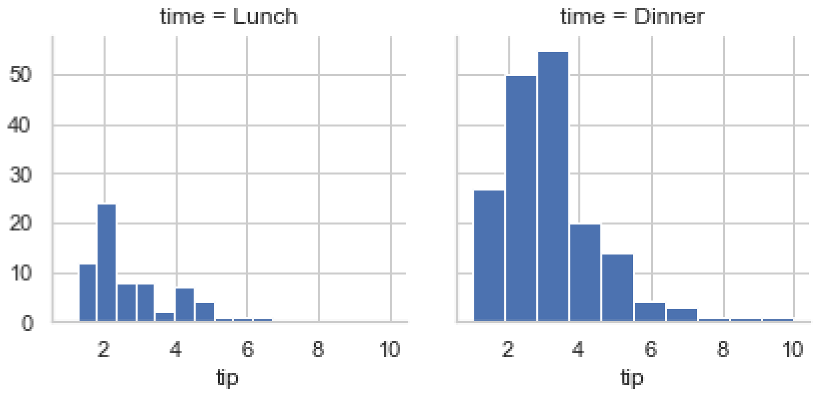
In [114]:

```
g = sns.FacetGrid(tips, col="time")
```



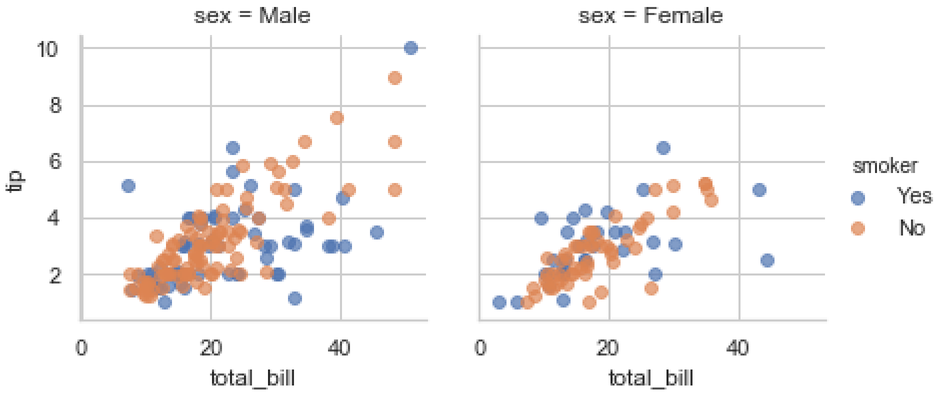
In [116]:

```
g = sns.FacetGrid(tips, col="time")
g.map(plt.hist, "tip");
```



In [118]:

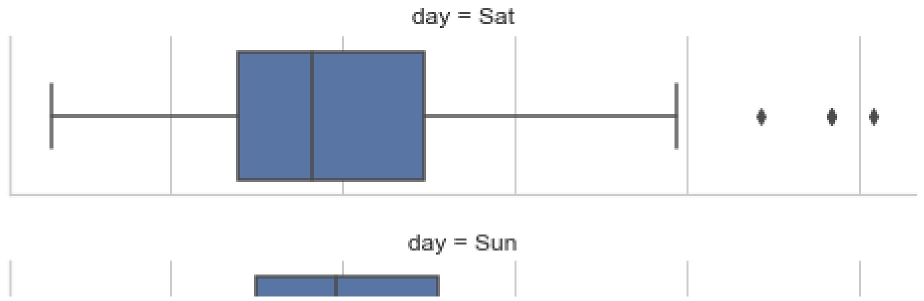
```
g = sns.FacetGrid(tips, col="sex", hue="smoker")
g.map(plt.scatter, "total_bill", "tip", alpha=.7)
g.add_legend();
```



In [125]:

```
from pandas import Categorical
ordered_days = tips.day.value_counts().index
print (ordered_days)

g = sns.FacetGrid(tips, row="day", row_order=ordered_days,
                  size=1.7, aspect=4,)
g.map(sns.boxplot, "total_bill");
```

```
CategoricalIndex(['Sat', 'Sun', 'Thur', 'Fri'], categories=['Thur', 'Fri', 'Sat',
'Sun'], ordered=False, dtype='category')

D:\Anaconda\ancaonda\lib\site-packages\seaborn\axisgrid.py:230: UserWarning: The `
size` paramter has been renamed to `height`; please update your code.
  warnings.warn(msg, UserWarning)
D:\Anaconda\ancaonda\lib\site-packages\seaborn\axisgrid.py:715: UserWarning: Using
the boxplot function without specifying `order` is likely to produce an incorrect
plot.
  warnings.warn(warning)
```



In [129]:

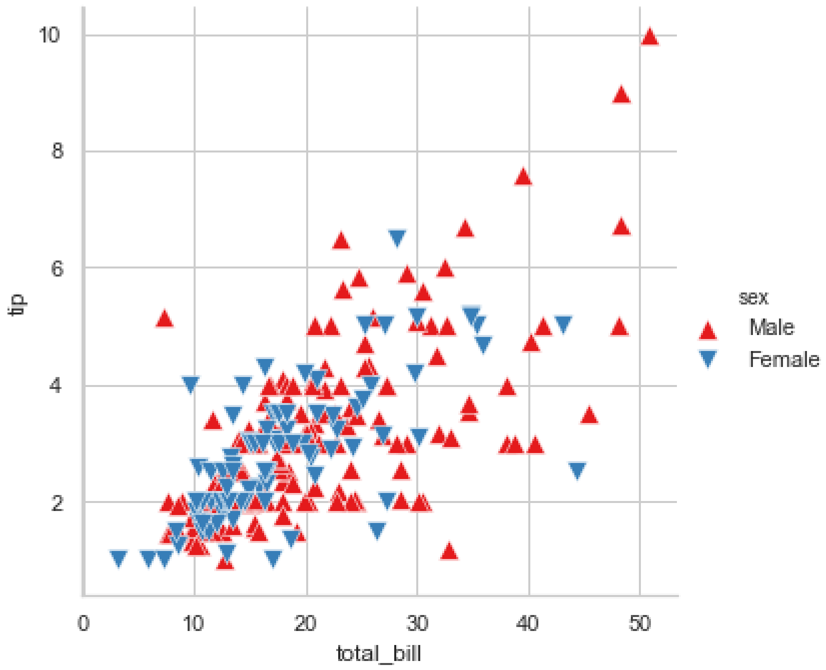
```
pal = dict(Lunch="seagreen", Dinner="gray")
g = sns.FacetGrid(tips, hue="time", palette=pal, size=5)
g.map(plt.scatter, "total_bill", "tip", s=50,alpha=.7,linewidth=.5,edgecolor="white")
g.add_legend();
```

```
  warnings.warn(msg, UserWarning)
```

In [132]:

```
g = sns.FacetGrid(tips, hue="sex", palette="Set1", size=5,hue_kws={"marker": ["^", "v"]})
g.map(plt.scatter, "total_bill", "tip", s=100,linewidth=.5,edgecolor="white")
g.add_legend();
```

```
D:\Anaconda\ancaonda\lib\site-packages\seaborn\axisgrid.py:230: UserWarning: The `si
ze` paramter has been renamed to `height`; please update your code.
  warnings.warn(msg, UserWarning)
```



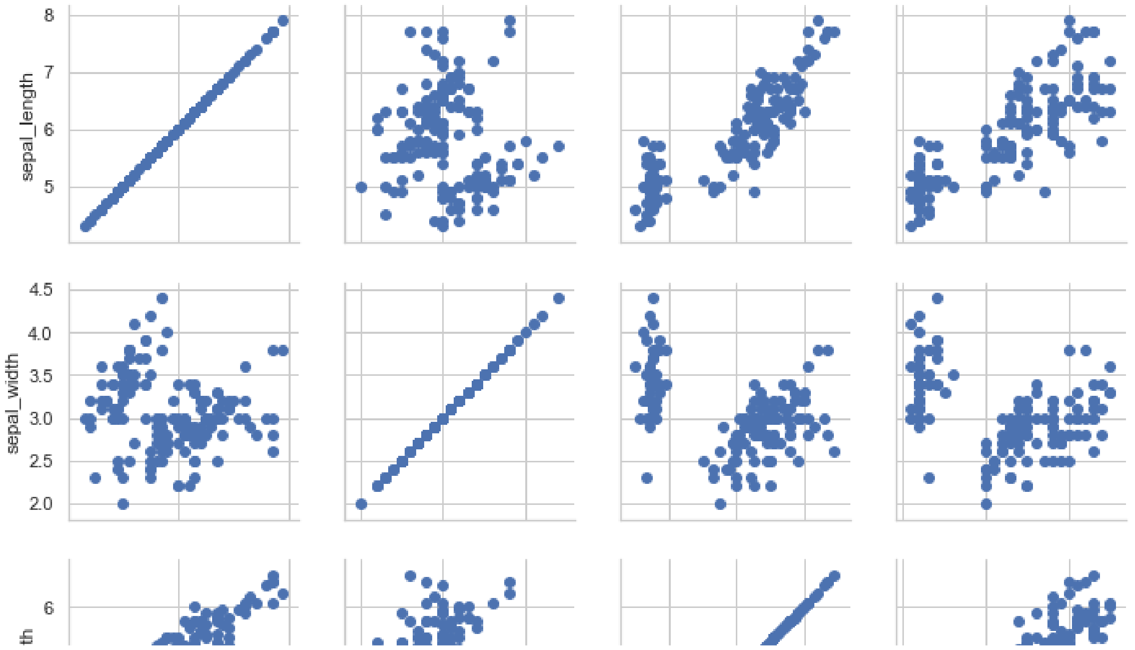
In [140]:

```
with sns.axes_style("white"):
    g = sns.FacetGrid(tips, row="sex", col="smoker",margin_titles=True,size=2.5)
g.map(plt.scatter, "total_bill", "tip", color="#334488", edgecolor="white",lw=1.5)
g.set_axis_labels("Total bill (US Dollars)", "Tip");
g.set(xticks=[10, 30, 50], ytick=[2,6,10]);
g.fig.subplots_adjust(wspace=0.2, hspace=.52);
```

```
D:\Anaconda\ancaonda\lib\site-packages\seaborn\axisgrid.py:230: UserWarning: The `
size` paramter has been renamed to `height`; please update your code.
  warnings.warn(msg, UserWarning)
```

In [141]:

```
iris = sns.load_dataset("iris")
g = sns.PairGrid(iris)
g.map(plt.scatter);
```

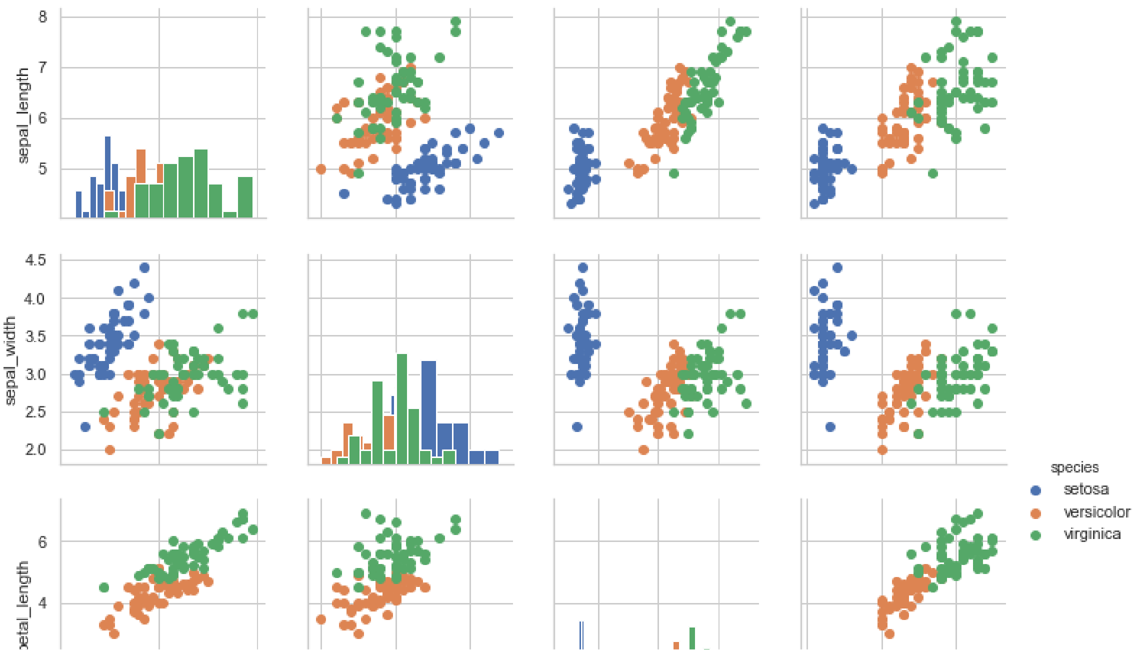


In [142]:

```
g = sns.PairGrid(iris)
g.map_diag(plt.hist)
g.map_offdiag(plt.scatter)
```

In [143]:

```
g = sns.PairGrid(iris, hue="species")
g.map_diag(plt.hist)
g.map_offdiag(plt.scatter)
g.add_legend();
```



In [144]:

```
g = sns.PairGrid(tips, hue="size", palette="GnBu_d")
g.map(plt.scatter, s=50, edgecolor="white")
g.add_legend();
```

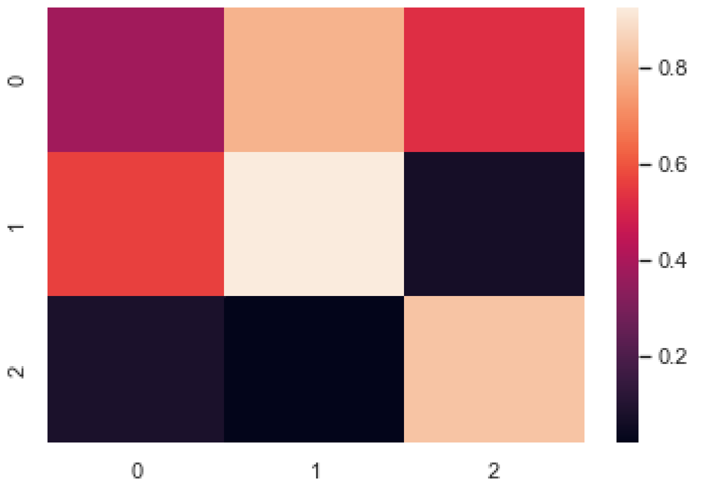
In [145]:

```
%matplotlib inline
import matplotlib.pyplot as plt
import numpy as np
np.random.seed(0)
import seaborn as sns;
sns.set()
```

In [147]:

```
uniform_data = np.random.rand(3,3)
print(uniform_data)
heatmap = sns.heatmap(uniform_data)
```

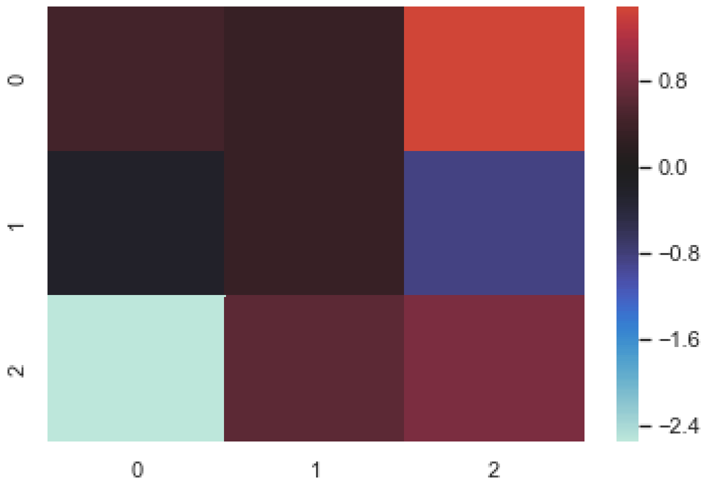
```
[[0.38344152 0.79172504 0.52889492]
 [0.56804456 0.92559664 0.07103606]
 [0.0871293  0.0202184  0.83261985]]
```



In [148]:

```
normal_data = np.random.randn(3,3)
print (normal_data)
ax = sns.heatmap(normal_data, center=0)
```

```
[[ 0.44386323  0.33367433  1.49407907]
 [-0.20515826  0.3130677  -0.85409574]
 [-2.55298982  0.6536186   0.8644362 ]]
```



In [152]:

```
flights = sns.load_dataset("flights")
flights.head()
```

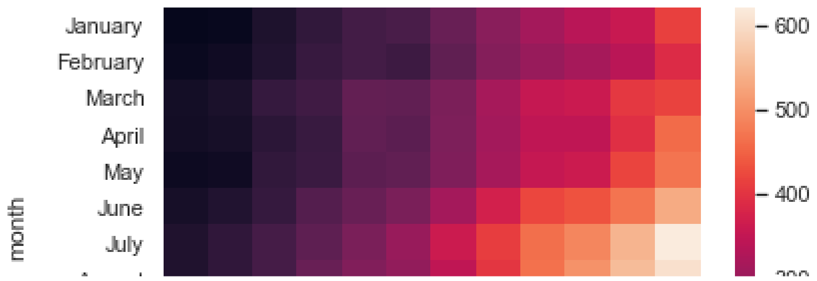
Out[152]:

	year	month	passengers
0	1949	January	112
1	1949	February	118
2	1949	March	132
3	1949	April	129
4	1949	May	121

In [153]:

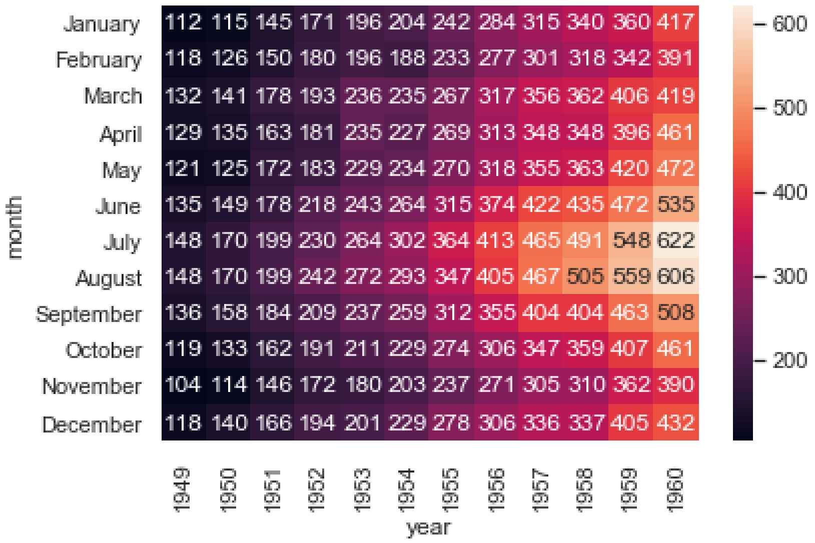
```
flights = flights.pivot("month", "year","passengers")
print (flights)
ax = sns.heatmap(flights)
```

```
November     390
December     432
```



In [154]:

```
ax = sns.heatmap(flights, annot=True, fmt="d")
```


January 112 115 145 171 196 204 242 284 315 340 360 417
February 118 126 150 180 196 188 233 277 301 318 342 391
March 132 141 178 193 236 235 267 317 356 362 406 419
April 129 135 163 181 235 227 269 313 348 348 396 461
May 121 125 172 183 229 234 270 318 355 363 420 472
June 135 149 178 218 243 264 315 374 422 435 472 535
July 148 170 199 230 264 302 364 413 465 491 548 622
August 148 170 199 242 272 293 347 405 467 505 559 606
September 136 158 184 209 237 259 312 355 404 404 463 508
October 119 133 162 191 211 229 274 306 347 359 407 461
November 104 114 146 172 180 203 237 271 305 310 362 390
December 118 140 166 194 201 229 278 306 336 337 405 432
month
year