In [1]:

```
import pandas as pd
import matplotlib.pyplot as plt
```

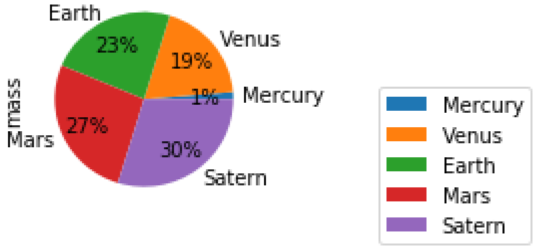
Using pandas df

In [2]:

```
df = pd.DataFrame({'mass': [0.330, 4.87 , 5.97, 6.8, 7.6]},
index=['Mercury', 'Venus', 'Earth', 'Mars', 'Satern'])
plot = df.plot.pie(y='mass', figsize=(2, 2), autopct='%1.0f%%', pctdistance=0.7, labeldistance=1.1) # pct will show percentage
plot.legend(loc='center left', bbox_to_anchor=(1.5, 0.2)) # This is put the legend information at right side
```

Out[2]:

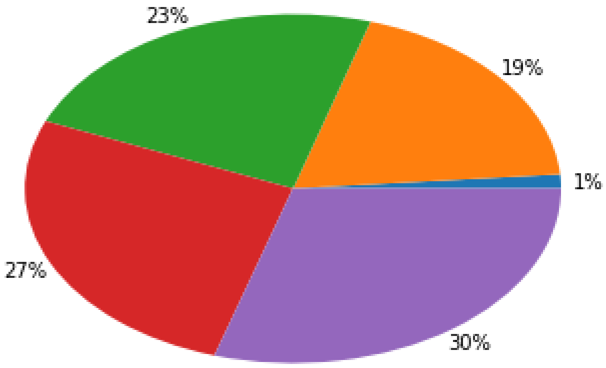
```
<matplotlib.legend.Legend at 0x25c84b0f630>
```



With plt and without label

In [40]:

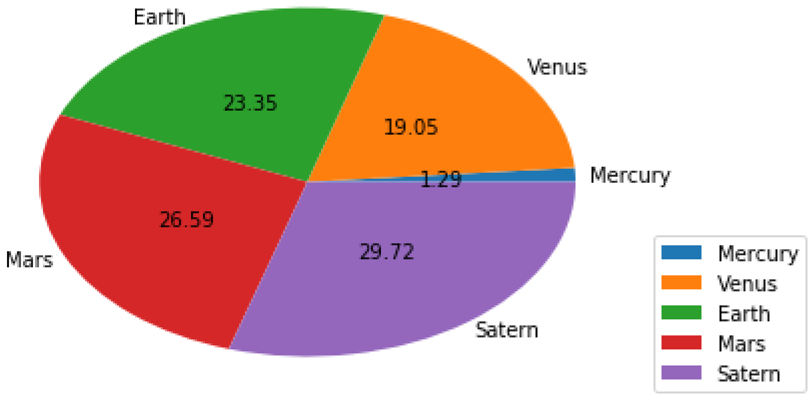
```
plt.close()
labels_info = ['Mercury', 'Venus', 'Earth', 'Mars', 'Satern']
data = [0.330, 4.87 , 5.97, 6.8, 7.6]
plt.pie(data, autopct='%1.0f%%', pctdistance=1.1, labeldistance=1.2)
plt.show()
```



With label and modification in pctdistance and labeldistance

In [20]:

```
plt.close()
labels_info = ['Mercury', 'Venus', 'Earth', 'Mars', 'Satern']
data = [0.330, 4.87 , 5.97, 6.8, 7.6]
plt.pie(data, labels= labels_info, autopct='%.2f', pctdistance=.5, labeldistance=1.05)
plt.legend(loc='center left', bbox_to_anchor=(1, 0.2))
plt.show()
```

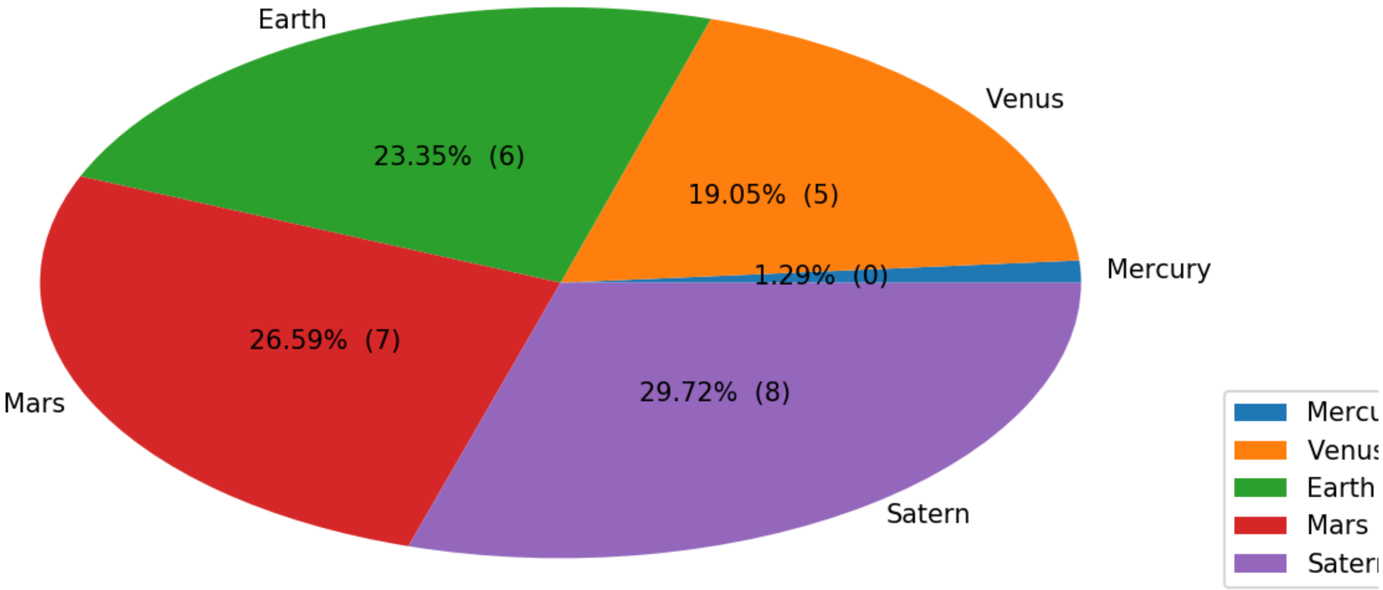


Adding actualy values along with percentage

In [21]:

```
def make_autopct(values):
    def my_autopct(pct):
        total = sum(values)
        val = int(round(pct*total/100.0))
        return '{p:.2f}%  ({v:d})'.format(p=pct,v=val)
    return my_autopct
```

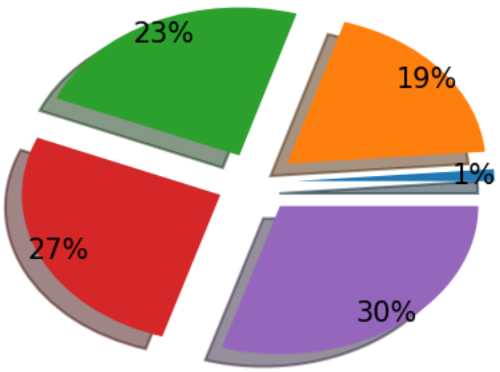
```
In [29]: plt.close()
         labels_info = ['Mercury', 'Venus', 'Earth', 'Mars', 'Satern']
         data = [0.330, 4.87 , 5.97, 6.8, 7.6]
         plt.pie(data, labels= labels_info, autopct=make_autopct(data), pctdistance=.5, labeldistance=1.05)
         plt.legend(loc='center left', bbox_to_anchor=(1, 0.2))
         plt.show()
```



Show the label when hover to the resepective region

In [41]:

```
plt.close()
labels_info = ['Mercury', 'Venus', 'Earth', 'Mars', 'Satern']
explode = [0.1, 0.1, 0.1, 0.1,0.1]
data = [0.330, 4.87 , 5.97, 6.8, 7.6]
plt.pie(data, explode = explode, autopct='%1.0f%%', pctdistance=0.9, labeldistance=1.2,radius=0.5, shadow=True)
plt.show()
```



In [24]:

```
%matplotlib notebook
```

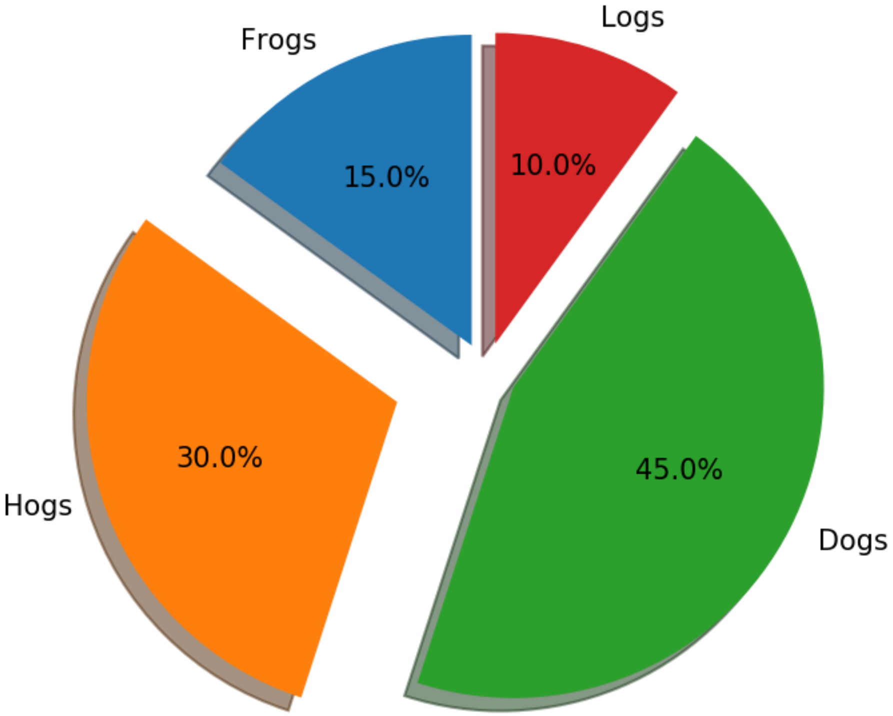
In [35]:

```
labels = 'Frogs', 'Hogs', 'Dogs', 'Logs'
sizes = [15, 30, 45, 10]

explode = [0.1, 0.3, 0.1, 0.1]  # only "explode" the 2nd slice (i.e. 'Hogs')

fig1, ax1 = plt.subplots()
ax1.pie(sizes, explode=explode, labels=labels, autopct='%1.1f%%',
        shadow=True, startangle=90)
ax1.axis('equal')  # Equal aspect ratio ensures that pie is drawn as a circle.

plt.show()
```



In [37]:
```
ex = [0] * 10
ex[1] = 1
ex
```

Out[37]:
```
[0, 1, 0, 0, 0, 0, 0, 0, 0, 0]
```

In [43]:
```
def calculate_rank(vector):
      a={}
      rank=1
      for num in sorted(vector):
          if num not in a:
              a[num]=rank
              rank=rank+1
      return[a[i] for i in vector]
```

In [47]:
```
r = calculate_rank([1,3,4,8,7,5,4,6])
# r*2
r
```

Out[47]:
```
[1, 2, 3, 7, 6, 4, 3, 5]
```

In [52]:
```
a = 0.01*40
a
```

Out[52]:
```
0.4
```

In [42]:
```
import scipy.stats as ss
ss.rankdata([3, 1, 4, 15, 92])
array([ 2.,  1.,  3.,  4.,  5.])
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-42-3bba91cd53d2> in <module>()
      1 import scipy.stats as ss
      2 ss.rankdata([3, 1, 4, 15, 92])
----> 3 array([ 2.,  1.,  3.,  4.,  5.])

NameError: name 'array' is not defined
```