In [47]:

```
# Dependencies and Setup
%matplotlib inline
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
from scipy.stats import sem
# Hide warning messages in notebook
import warnings
warnings.filterwarnings('ignore')
```

In [48]:

```
# File to Load (Remember to Change These)
mouse_drug_data_to_load = "../data/mouse_drug_data.csv"
clinical_trial_data_to_load = "../data/clinicaltrial_data.csv"

# Read the Mouse and Drug Data and the Clinical Trial Data
drug_data=pd.read_csv(mouse_drug_data_to_load)
clinical_data=pd.read_csv(clinical_trial_data_to_load)

# Combine the data into a single dataset

trial_data=pd.merge(drug_data,clinical_data,on = "Mouse ID")
# Display the data table for preview
trial_data.head()
```

Out[48]:

	Mouse ID	Drug	Timepoint	Tumor Volume (mm3)	Metastatic Sites
0	f234	Stelasyn	0	45.000000	0
1	f234	Stelasyn	5	47.313491	0
2	f234	Stelasyn	10	47.904324	0
3	f234	Stelasyn	15	48.735197	1
4	f234	Stelasyn	20	51.112713	2

Tumor Response to Treatment

In [49]:

```
# Store the Mean Tumor Volume Data Grouped by Drug and Timepoint
tumor_response =  trial_data.groupby(["Drug","Timepoint"]).agg({"Tumor Volume (mm3)":"mean"}).reset_index()
# # Preview DataFrame
tumor_response.head()
```

Out[49]:

	Drug	Timepoint	Tumor Volume (mm3)
0	Capomulin	0	45.000000
1	Capomulin	5	44.266086
2	Capomulin	10	43.084291
3	Capomulin	15	42.064317
4	Capomulin	20	40.716325

In [50]:

```
# Store the Standard Error of Tumor Volumes Grouped by Drug and Timepoint
tumor_vol_sem= trial_data.groupby(["Drug","Timepoint"]).agg({"Tumor Volume (mm3)":"sem"})
#print(tumor_vol_sem)
# Convert to DataFrame
tumor_vol_df = pd.DataFrame(tumor_vol_sem).reset_index()
# Preview DataFrame
tumor_vol_df.head()
```

Out[50]:

	Drug	Timepoint	Tumor Volume (mm3)
0	Capomulin	0	0.000000
1	Capomulin	5	0.448593
2	Capomulin	10	0.702684
3	Capomulin	15	0.838617
4	Capomulin	20	0.909731

In [51]:
```
# Minor Data Munging to Re-Format the Data Frames


tumor_reformat= trial_data.pivot_table(index='Timepoint',
                             columns='Drug',
                             values='Tumor Volume (mm3)'

                             )
# Preview that Reformatting worked

tumor_reformat
```

Out[51]:

Drug	Capomulin	Ceftamin	Infubinol	Ketapril	Naftisol	Placebo	Propriva	Ramica
Timepoint								
0	45.000000	45.000000	45.000000	45.000000	45.000000	45.000000	45.000000	45.0000
5	44.266086	46.503051	47.062001	47.389175	46.796098	47.125589	47.248967	43.9448
10	43.084291	48.285125	49.403909	49.582269	48.694210	49.423329	49.101541	42.5319
15	42.064317	50.094055	51.296397	52.399974	50.933018	51.359742	51.067318	41.4950
20	40.716325	52.157049	53.197691	54.920935	53.644087	54.364417	53.346737	40.2383
25	39.939528	54.287674	55.715252	57.678982	56.731968	57.482574	55.504138	38.9743
30	38.769339	56.769517	58.299397	60.994507	59.559509	59.809063	58.196374	38.7031
35	37.816839	58.827548	60.742461	63.371686	62.685087	62.420615	60.350199	37.4519
40	36.958001	61.467895	63.162824	66.068580	65.600754	65.052675	63.045537	36.5740
45	36.236114	64.132421	65.755562	70.662958	69.265506	68.084082	66.258529	34.9555

In [52]:
```
capomulin1_data= trial_data.loc[trial_data["Drug"]=="Capomulin",["Timepoint",
"Tumor Volume (mm3)"]]
capomulin1_grouped=capomulin1_data.groupby("Timepoint").mean().reset_index()
```

In [53]:
```
infubinol1_data= trial_data.loc[trial_data["Drug"]=="Infubinol",["Timepoint",
"Tumor Volume (mm3)"]]
infubinol1_grouped=infubinol1_data.groupby("Timepoint").mean().reset_index()
```

In [54]:
```
ketapril1_data= trial_data.loc[trial_data["Drug"]=="Ketapril",["Timepoint","Tu
mor Volume (mm3)"]]
ketapril1_grouped=ketapril1_data.groupby("Timepoint").mean().reset_index()
```

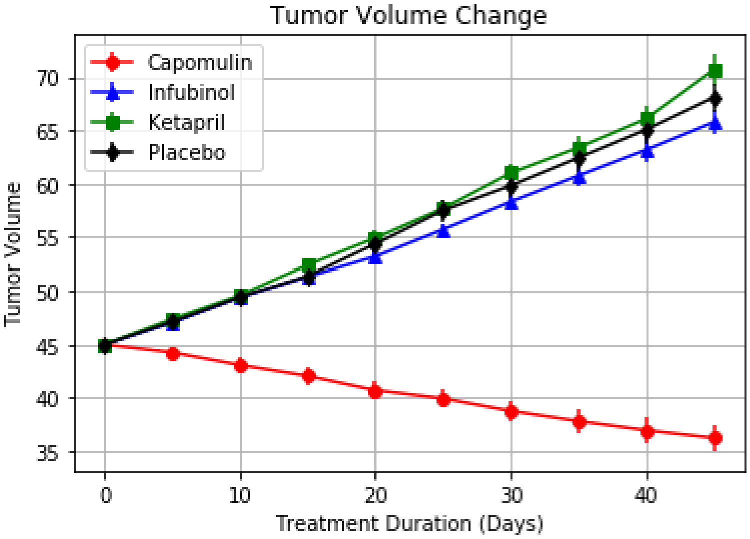
In [55]:
```
placebo1_data= trial_data.loc[trial_data["Drug"]=="Placebo",["Timepoint","Tumo
r Volume (mm3)"]]
placebo1_grouped=placebo1_data.groupby("Timepoint").mean().reset_index()
```

In [56]:

```
#assign values to x,y to use out plot
c1_x= capomulin1_grouped["Timepoint"]
c1_y=capomulin1_grouped["Tumor Volume (mm3)"]
i1_x= infubinol1_grouped["Timepoint"]
i1_y=infubinol1_grouped["Tumor Volume (mm3)"]
k1_x= ketapril1_grouped["Timepoint"]
k1_y=ketapril1_grouped["Tumor Volume (mm3)"]
p1_x= placebo1_grouped["Timepoint"]
p1_y=placebo1_grouped["Tumor Volume (mm3)"]
```

In [57]:

```
# Generate the Plot (with Error Bars)

capo1_handle=plt.errorbar(c1_x, c1_y,yerr= tumor_vol_sem.loc['Capomulin']['Tumor Volume (mm3)'], marker='o', color='red',label= "Capomulin")
infu1_handle=plt.errorbar(i1_x,i1_y, yerr= tumor_vol_sem.loc['Infubinol']['Tumor Volume (mm3)'], marker='^', color='blue',label= "Infubinol")
keta1_handle=plt.errorbar(k1_x,k1_y,yerr= tumor_vol_sem.loc['Ketapril']['Tumor Volume (mm3)'],marker='s', color='green',label= "Ketapril")
plac1_handle=plt.errorbar(p1_x,p1_y,yerr= tumor_vol_sem.loc['Placebo']['Tumor Volume (mm3)'], marker='d', color='black',label= "Placebo")

# Create a legend for our chart
plt.title("Tumor Volume Change")
plt.xlabel("Treatment Duration (Days)")
plt.ylabel("Tumor Volume")
plt.legend(loc="best")
# Show the chart
plt.grid()
# Save the Figure
plt.savefig("TumorVolumeChange.png")
plt.show()
```



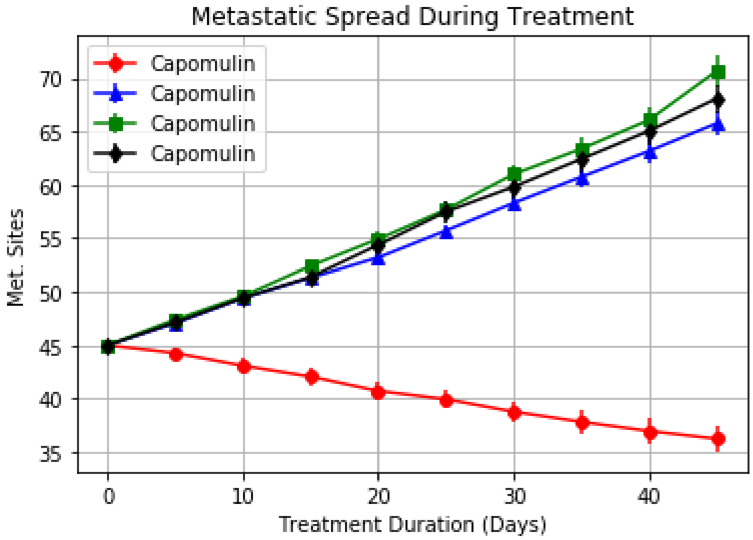
I tried the above method first and figured out how to the same another way. So including both.

In [58]:

```
#Finds the sem for the Tumor Volume
tumor_response_pivot=tumor_response.pivot(index="Timepoint", columns="Drug")[
"Tumor Volume (mm3)"]
tumor_vol_sem_pivot= tumor_vol_sem.pivot_table(index='Timepoint',
                             columns='Drug',
                             values='Tumor Volume (mm3)'

                             )
```

In [59]:

```
#plot graph using the values from the pivot fuction
plt.errorbar(tumor_response_pivot.index,tumor_response_pivot["Capomulin"],yerr
=tumor_vol_sem_pivot["Capomulin"], marker='o', color='red',label= "Capomulin")
plt.errorbar(tumor_response_pivot.index,tumor_response_pivot["Infubinol"],yerr
=tumor_vol_sem_pivot["Infubinol"], marker='^', color='blue',label= "Capomulin"
)
plt.errorbar(tumor_response_pivot.index,tumor_response_pivot["Ketapril"],yerr=
tumor_vol_sem_pivot["Ketapril"], marker='s', color='green',label= "Capomulin")
plt.errorbar(tumor_response_pivot.index,tumor_response_pivot["Placebo"],yerr=t
umor_vol_sem_pivot["Placebo"], marker='d', color='black',label= "Capomulin")
plt.title("Metastatic Spread During Treatment")
plt.xlabel("Treatment Duration (Days)")
plt.ylabel("Met. Sites")
plt.legend(loc="best")
# Show the chart
plt.grid()
plt.show()
# Save the Figure
plt.savefig("MetastaticSpreaAlt.png")
```



```
<Figure size 432x288 with 0 Axes>
```

Metastatic Response to Treatment

In [60]:

```
# Store the Mean Met. Site Data Grouped by Drug and Timepoint
ms_data= trial_data.groupby(["Drug","Timepoint"]).agg({"Metastatic Sites":"mean"})


# Preview DataFrame
ms_data.head()
```

Out[60]:

		Metastatic Sites
Drug	**Timepoint**	
Capomulin	**0**	0.000000
	5	0.160000
	10	0.320000
	15	0.375000
	20	0.652174

In [61]:

```
# Store the Standard Error associated with Met. Sites Grouped by Drug and Timepoint
metas_sem= trial_data.groupby(["Drug","Timepoint"]).agg({"Metastatic Sites":"sem"})
# Convert to DataFrame
metas_df = pd.DataFrame(metas_sem)
# Preview DataFrame
metas_df.head()
```

Out[61]:

		Metastatic Sites
Drug	**Timepoint**	
Capomulin	**0**	0.000000
	5	0.074833
	10	0.125433
	15	0.132048
	20	0.161621

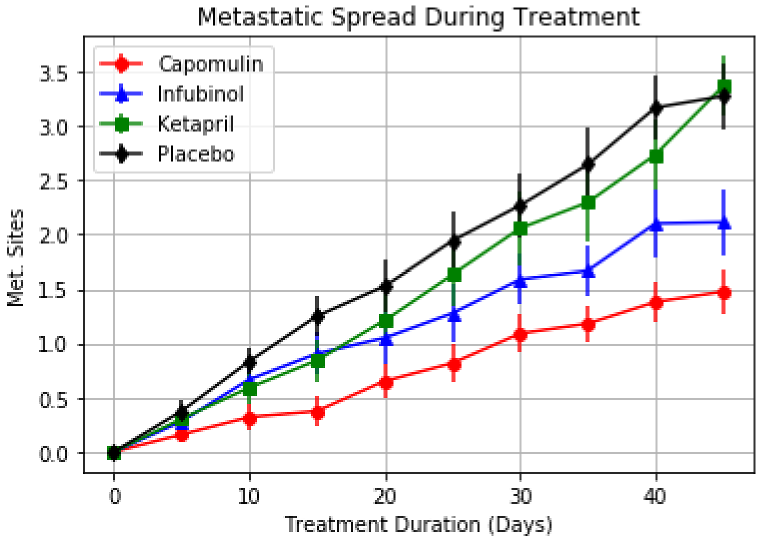
In [62]:
```
# Minor Data Munging to Re-Format the Data Frames
meta_reformat= trial_data.pivot_table(index='Timepoint',
                             columns='Drug',
                             values='Tumor Volume (mm3)',
                             )
# Preview that Reformatting worked
meta_reformat
```

Out[62]:
```
     Drug  Capomulin   Ceftamin  Infubinol   Ketapril   Naftisol    Placebo   Propriva   Ramica
Timepoint
        0  45.000000  45.000000  45.000000  45.000000  45.000000  45.000000  45.000000  45.0000
        5  44.266086  46.503051  47.062001  47.389175  46.796098  47.125589  47.248967  43.9448
       10  43.084291  48.285125  49.403909  49.582269  48.694210  49.423329  49.101541  42.5319
       15  42.064317  50.094055  51.296397  52.399974  50.933018  51.359742  51.067318  41.4950
       20  40.716325  52.157049  53.197691  54.920935  53.644087  54.364417  53.346737  40.2383
       25  39.939528  54.287674  55.715252  57.678982  56.731968  57.482574  55.504138  38.9743
       30  38.769339  56.769517  58.299397  60.994507  59.559509  59.809063  58.196374  38.7031
       35  37.816839  58.827548  60.742461  63.371686  62.685087  62.420615  60.350199  37.4519
       40  36.958001  61.467895  63.162824  66.068580  65.600754  65.052675  63.045537  36.5740
       45  36.236114  64.132421  65.755562  70.662958  69.265506  68.084082  66.258529  34.9555
```

In [63]:
```
capomulin2_data= trial_data.loc[trial_data["Drug"]=="Capomulin",["Timepoint",
"Metastatic Sites"]]
capomulin2_grouped=capomulin2_data.groupby("Timepoint").mean().reset_index()
infubinol2_data= trial_data.loc[trial_data["Drug"]=="Infubinol",["Timepoint",
"Metastatic Sites"]]
infubinol2_grouped=infubinol2_data.groupby("Timepoint").mean().reset_index()
ketapril2_data= trial_data.loc[trial_data["Drug"]=="Ketapril",["Timepoint","Me
tastatic Sites"]]
ketapril2_grouped=ketapril2_data.groupby("Timepoint").mean().reset_index()
placebo2_data= trial_data.loc[trial_data["Drug"]=="Placebo",["Timepoint","Meta
static Sites"]]
placebo2_grouped=placebo2_data.groupby("Timepoint").mean().reset_index()
```

In [64]:
```
#assign values to x,y to use out plot
c2_x= capomulin2_grouped["Timepoint"]
c2_y=capomulin2_grouped["Metastatic Sites"]
i2_x= infubinol2_grouped["Timepoint"]
i2_y=infubinol2_grouped["Metastatic Sites"]
k2_x= ketapril2_grouped["Timepoint"]
k2_y=ketapril2_grouped["Metastatic Sites"]
p2_x= placebo2_grouped["Timepoint"]
p2_y=placebo2_grouped["Metastatic Sites"]
# Save the Figure

# Show the Figure
```

In [65]:
```
# Generate the Plot (with Error Bars)
capo2_handle=plt.errorbar(c2_x, c2_y,yerr= metas_sem.loc['Capomulin']['Metastatic Sites'], marker='o', color='red',label= "Capomulin")
infu2_handle=plt.errorbar(i2_x, i2_y,yerr= metas_sem.loc['Infubinol']['Metastatic Sites'], marker='^', color='blue',label= "Infubinol")
keta2_handle=plt.errorbar(k2_x,k2_y,yerr= metas_sem.loc['Ketapril']['Metastatic Sites'],marker='s', color='green',label= "Ketapril")
plac2_handle=plt.errorbar(p2_x,p2_y,yerr= metas_sem.loc['Placebo']['Metastatic Sites'], marker='d', color='black',label= "Placebo")

# Create a legend for our chart
plt.title("Metastatic Spread During Treatment")
plt.xlabel("Treatment Duration (Days)")
plt.ylabel("Met. Sites")
plt.legend(loc="best")
# Show the chart
plt.grid()
plt.show()
# Save the Figure
plt.savefig("MetastaticSpread.png")
```



```
<Figure size 432x288 with 0 Axes>
```

Survival Rates

In [66]:

```
# Store the Count of Mice Grouped by Drug and Timepoint (W can pass any metri
c)
survival_data= trial_data.groupby(["Drug","Timepoint"]).agg({"Mouse ID":"coun
t"})

# Preview DataFrame
survival_data.head()
```

Out[66]:

		Mouse ID
Drug	**Timepoint**	
Capomulin	**0**	25
	5	25
	10	25
	15	24
	20	23

In [67]:

```
# Minor Data Munging to Re-Format the Data Frames
survival_reformat= trial_data.pivot_table(index='Timepoint',
                             columns='Drug',
                             values='Mouse ID',
                             aggfunc='count'

                             )
# Preview the Data Frame
survival_reformat
```

Out[67]:

Drug	Capomulin	Ceftamin	Infubinol	Ketapril	Naftisol	Placebo	Propriva	Ramicane	Stel
Timepoint									
0	25	25	25	25	25	25	26	25	
5	25	21	25	23	23	24	25	25	
10	25	20	21	22	21	24	23	24	
15	24	19	21	19	21	20	17	24	
20	23	18	20	19	20	19	17	23	
25	22	18	18	19	18	17	14	23	
30	22	16	17	18	15	15	13	23	
35	22	14	12	17	15	14	10	21	
40	21	14	10	15	15	12	9	20	
45	21	13	9	11	13	11	7	20	

In [68]:

```
# Generate the Plot (Accounting for percentages)
drug_df=trial_data[trial_data['Drug'].isin(['Capomulin','Infubinol','Ketapril','Placebo'])]
drug_data= drug_df.groupby('Drug')
drug_data.head()
```

Out[68]:

	Mouse ID	Drug	Timepoint	Tumor Volume (mm3)	Metastatic Sites
393	q119	Ketapril	0	45.000000	0
394	q119	Ketapril	5	47.864440	0
395	q119	Ketapril	10	51.236606	0
396	n923	Ketapril	0	45.000000	0
397	n923	Ketapril	5	45.824881	0
581	b128	Capomulin	0	45.000000	0
582	b128	Capomulin	5	45.651331	0
583	b128	Capomulin	10	43.270852	0
584	b128	Capomulin	15	43.784893	0
585	b128	Capomulin	20	42.731552	0
811	q132	Infubinol	0	45.000000	0
812	q132	Infubinol	5	46.716399	1
813	q132	Infubinol	10	47.953844	2
814	q132	Infubinol	15	49.159749	3
815	q132	Infubinol	20	51.909025	4
1725	a897	Placebo	0	45.000000	0
1726	a897	Placebo	5	45.691124	1
1727	a897	Placebo	10	47.972927	1
1728	a897	Placebo	15	52.343085	2
1729	a897	Placebo	20	54.595705	3

In [69]:

```
capo3_surv= trial_data.loc[trial_data["Drug"]=="Capomulin",["Timepoint","Mouse ID"]]
capo3_surv_grouped=capo3_surv.groupby("Timepoint").count()["Mouse ID"].reset_index()
#Calculated % by multiplying by 4 instead of 100/25 -25 being count of Mouse ID
capo3_surv_grouped['Percent']=capo3_surv_grouped["Mouse ID"]*4
#assign values to x,y to use out plot
c3_x=capo3_surv_grouped["Timepoint"]
c3_y= capo3_surv_grouped["Percent"]
```

In [70]:

```
infu3_surv= trial_data.loc[trial_data["Drug"]=="Infubinol",["Timepoint","Mouse ID"]]
infu3_surv_grouped=infu3_surv.groupby("Timepoint").count()["Mouse ID"].reset_index()
#Calculated % by multiplying by 4 instead of 100/25 -25 being count of Mouse ID
infu3_surv_grouped['Percent']=infu3_surv_grouped["Mouse ID"]*4
#assign values to x,y to use out plot
i3_x=infu3_surv_grouped["Timepoint"]
i3_y= infu3_surv_grouped["Percent"]
```

In [71]:

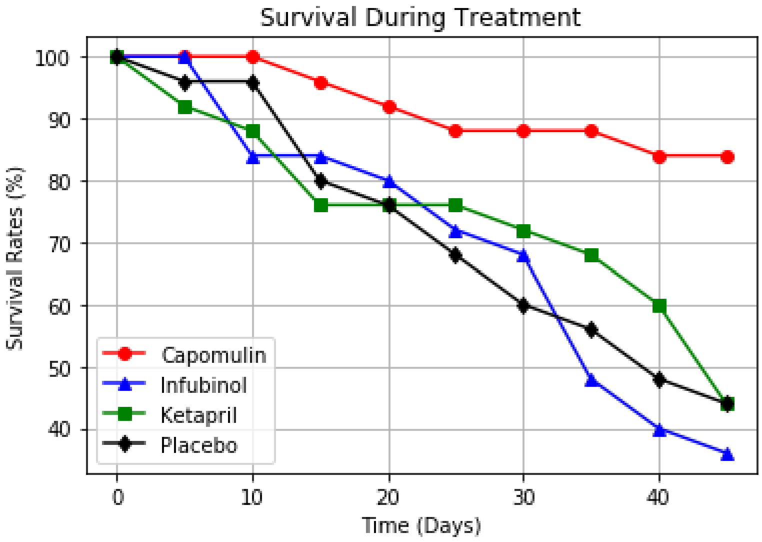
```
keta3_surv= trial_data.loc[trial_data["Drug"]=="Ketapril",["Timepoint","Mouse ID"]]
keta3_surv_grouped=keta3_surv.groupby("Timepoint").count()["Mouse ID"].reset_index()
#Calculated % by multiplying by 4 instead of 100/25 -25 being count of Mouse ID
keta3_surv_grouped['Percent']=keta3_surv_grouped["Mouse ID"]*4
#assign values to x,y to use out plot
k3_x=keta3_surv_grouped["Timepoint"]
k3_y= keta3_surv_grouped["Percent"]
```

In [72]:

```
plac3_surv= trial_data.loc[trial_data["Drug"]=="Placebo",["Timepoint","Mouse ID"]]
plac3_surv_grouped=plac3_surv.groupby("Timepoint").count()["Mouse ID"].reset_index()
#Calculated % by multiplying by 4 instead of 100/25 -25 being count of Mouse ID
plac3_surv_grouped['Percent']=plac3_surv_grouped["Mouse ID"]*4
#assign values to x,y to use out plot
p3_x=plac3_surv_grouped["Timepoint"]
p3_y= plac3_surv_grouped["Percent"]
```

In [73]:

```
#plots a line graph for the 4 drugs
capo3_handle=plt.plot(c3_x, c3_y ,marker='o', color='red',label= "Capomulin")
infu3_handle=plt.plot(i3_x,i3_y,marker='^', color='blue',label= "Infubinol")
keta3_handle=plt.plot(k3_x,k3_y,marker='s', color='green',label= "Ketapril")
plac3_handle=plt.plot(p3_x,p3_y,marker='d', color='black',label= "Placebo")

# Creates a legend for our chart
plt.title("Survival During Treatment")
plt.xlabel("Time (Days)")
plt.ylabel("Survival Rates (%)")
plt.legend(loc="best")
# Show the chart
plt.grid()
# Save the Figure
plt.savefig("MetastaticSpread.png")
plt.show()
```



Summary Bar Graph

In [74]:

```
# Calculate the percent changes for each drug
size_change_all = ((tumor_reformat.loc[45, :] - tumor_reformat.loc[0, :])/tumo
r_reformat.loc[0, :] * 100)
size_change_all['Tum Vol Change Pcent'] = ((tumor_reformat.loc[45, :] - tumor_
reformat.loc[0, :])/tumor_reformat.loc[0, :] * 100)
```

In [75]:

```
# Display the data to confirm
size_change_all['Tum Vol Change Pcent']
```

Out[75]:

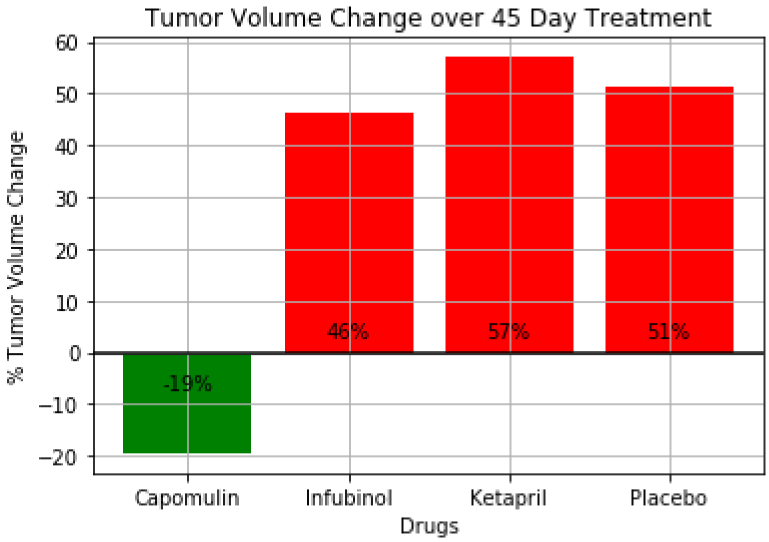
```
Drug
Capomulin   -19.475303
Ceftamin     42.516492
Infubinol    46.123472
Ketapril     57.028795
Naftisol     53.923347
Placebo      51.297960
Propriva     47.241175
Ramicane    -22.320900
Stelasyn     52.085134
Zoniferol    46.579751
dtype: float64
```

In [76]:

```
# Store all Relevant Percent Changes into a Tuple
c4_y= ((tumor_reformat.loc[45, :] - tumor_reformat.loc[0, :])/tumor_reformat.l
oc[0, :] * 100)[['Capomulin']]
i4_y= ((tumor_reformat.loc[45, :] - tumor_reformat.loc[0, :])/tumor_reformat.l
oc[0, :] * 100)[['Infubinol']]
k4_y= ((tumor_reformat.loc[45, :] - tumor_reformat.loc[0, :])/tumor_reformat.l
oc[0, :] * 100)[['Ketapril']]
p4_y= ((tumor_reformat.loc[45, :] - tumor_reformat.loc[0, :])/tumor_reformat.l
oc[0, :] * 100)[['Placebo']]
```

In [77]:

```
#creates a list of labels and assign them to the bars
labels= ['Capomulin','Infubinol','Ketapril','Placebo']
capo4_handle= plt.bar(labels[0],c4_y,color='green')
infu4_handle= plt.bar(labels[1],i4_y,color='red')
keta4_handle=plt.bar(labels[2],k4_y,color='red')
plac4_handle=plt.bar(labels[3],p4_y,color='red')

#adds a horizontal line at zero
plt.axhline(y=0, color = 'black')


# Add title,labels, tick marks, etc.
plt.title('Tumor Volume Change over 45 Day Treatment')
plt.ylabel('% Tumor Volume Change')
plt.xlabel('Drugs')

# Use functions to label the percentages of changes
# Code Credit: http://composition.al/blog/2015/11/29/a-better-way-to-add-label
s-to-bar-charts-with-matplotlib/

def autolabel(rects):
    for rect in rects:
        height = rect.get_height()
        if height > 0:
            label_position = 2
        else:
            label_position = -8
        plt.text(rect.get_x() + rect.get_width()/2., label_position,
                '%d' % int(height)+'%',color='black',
                ha='center', va='bottom')
# Calls functions to implement the function calls
autolabel(capo4_handle)
autolabel(infu4_handle)
autolabel(keta4_handle)
autolabel(plac4_handle)
plt.grid()
# Saves the Figure
plt.savefig("SummaryTumorChange.png")

# Shows the Figure
plt.show()
```



Observations

1)Ramicane has been the best performing drug followed by Capomulin when measured against Tumor volume reduction and Number of metastatic sites.

2)Ketapril has been the worst performing drug in terms of tumor volume change and metastatic sites. It has performed worse than the Placebo which is essentially the "do nothing" option.

3)propriva and Infubinol have the worst survival rate where as Capomulin followed by Infubinol have the best survival rate.