In [1]:
```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd

%matplotlib inline
```

In [2]:
```
planets = pd.read_csv(r'C:\Users\mikel\Documents\planets.csv', low_memory = False)
```

In [83]:
```
planets.head()
```

Out[83]:

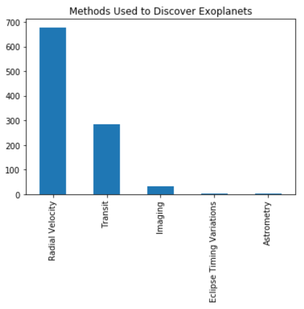
	34	pl_hostname	pl_letter	pl_name	pl_discmethod	pl_pnum	pl_orbper	pl_orbsmax	pl_orbincl	pl_bmassj	...	dec	st_dist	gaia_dist	st_optmag
0	35	11 Com	b	11 Com b	Radial Velocity	1	326.03000	1.29	NaN	19.40	...	17.792868	93.37	93.37	4.740
1	36	11 UMi	b	11 UMi b	Radial Velocity	1	516.21997	1.53	NaN	14.74	...	71.823898	125.72	125.72	5.016
2	37	14 And	b	14 And b	Radial Velocity	1	185.84000	0.83	NaN	4.80	...	39.236198	75.59	75.59	5.227
3	38	14 Her	b	14 Her b	Radial Velocity	1	1773.40002	2.93	NaN	4.66	...	43.817646	17.94	17.94	6.610
4	39	16 Cyg B	b	16 Cyg B b	Radial Velocity	1	798.50000	1.66	NaN	1.78	...	50.517525	21.15	21.15	6.250

5 rows × 28 columns

The dataset I have chosen to examine is a compilation of various exoplanets and their properties, gathered together by NASA.

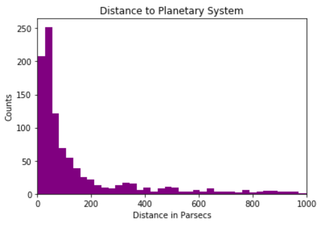
In [111]:
```
planets['pl_discmethod'].value_counts().plot.bar(title = "Methods Used to Discover Exoplanets")
```

Out[111]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x2bac91eccf8>
```



This simple bar chart shows different methods used to discover exoplanets. From the chart we see that the radial velocity method is by far the most common method used, more than double the next closest, the transit method. The other three methods are used fairly infrequently.

In [106]:
```
plt.hist(planets['st_dist'], color = 'purple', bins = 100)
plt.xlim(0, 1000)
plt.xticks()
plt.xlabel('Distance in Parsecs')
plt.ylabel('Counts')
plt.title('Distance to Planetary System')
plt.show()
```



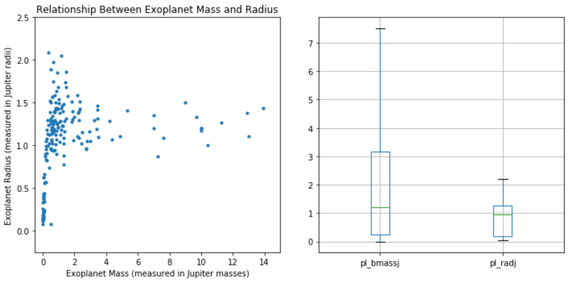
This histogram shows the distribution of exoplanets grouped together by their distance from Earth. This shows that the majority of exoplanets in the dataset are within 200 parsecs of our system. As the distance increases, the number of exoplanets decreases.

In [135]:
```
plt.figure(figsize = (10, 5))

plt.subplot(1, 2, 1)
plt.scatter(planets['pl_bmassj'], planets['pl_radj'], s = 10)
plt.xlim(-0.5, 15)
plt.ylim(-0.25, 2.5)
plt.xlabel('Exoplanet Mass (measured in Jupiter masses)')
plt.ylabel('Exoplanet Radius (measured in Jupiter radii)')
plt.title('Relationship Between Exoplanet Mass and Radius')

plt.subplot(1, 2, 2)
planets.boxplot(column = ['pl_bmassj', 'pl_radj'], showfliers = False)

plt.tight_layout()
plt.show()
```



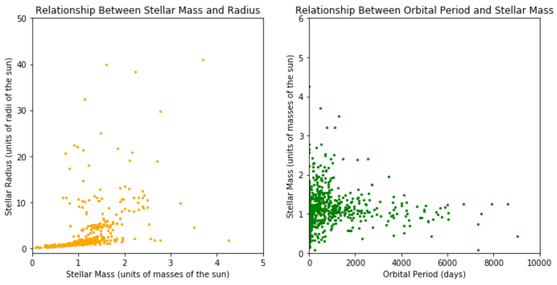
These two plots show the relationship between exoplanet mass and radius. The boxplots show the range of both these values, however, extreme outliers were removed to get a clear representation of the plots. The scatter plot shows what looks to be a slight positive correlation between mass and radius, but further testing would be warranted.

In [134]:
```
plt.figure(figsize = (10, 5))

plt.subplot(1, 2, 1)
plt.scatter(planets['st_mass'], planets['st_rad'], color = "orange", s = 5)
plt.xlim(0, 5)
plt.ylim(-1, 50)
plt.xlabel('Stellar Mass (units of masses of the sun)')
plt.ylabel('Stellar Radius (units of radii of the sun)')
plt.title('Relationship Between Stellar Mass and Radius')

plt.subplot(1, 2, 2)
plt.scatter(planets['pl_orbper'], planets['st_mass'], color = "green", s = 5)
plt.xlim(0, 10000)
plt.ylim(0, 6)
plt.xlabel('Orbital Period (days)')
plt.ylabel('Stellar Mass (units of masses of the sun)')
plt.title('Relationship Between Orbital Period and Stellar Mass')

plt.tight_layout()
plt.show()
```



The final plot show the relationship between the stellar mass and radius of the exoplanet's star and the relationship between the stellar mass and orbital period of the exoplanet. The scatter plot of stellar mass and stellar radius seems to show a slightly positive correlation between the variable, similar to exoplanet mass and radius. The relationship between orbital period and stellar mass on the other hand seems to be fairly random indicating little correlation.