Exploritory Analysis I

Exploritory Analysis of the Datasets to be used for
Machine Learning Predictions of Exoplanet candidates

The analyis will look briefly at:

- The Confirmed Planets datset being used.
- The main features of the dataset
- If any of the features will be needed for the Classification Machine Learning

In [1]:
```
#imports
import pandas as pd
import numpy as np
import lightkurve as lk
import matplotlib.pyplot as plt
import seaborn as sn
```

Kepler Confirmed Exoplanets

Kepler confirmed exoplanets are downloaded from through the MAST API and give details on the exoplanet such as the star name it orbits, the star mass, brightness, and the orbital period of the confirmed exoplanet

In [2]:
```
# data downloaded in csv format from API query string given by MAST
confirmed_file=("https://exoplanetarchive.ipac.caltech.edu/TAP/sync?query=s


# dataframe object of the csv
confirmed_df = pd.read_csv(confirmed_file, low_memory=False)
```

In [3]:
```
# basic details on the dataframe
confirmed_df.shape
```

Out[3]:
```
(4884, 373)
```

The dataframe contains 4,884 rows and 373 columns

In [4]:
```
# print first 5 rows of dataset
confirmed_df.head()
```

Out[4]:

	pl_name	pl_letter	hostname	hd_name	hip_name	tic_id	disc_pubdate	disc_year	di
0	OGLE-2016-BLG-1227L b	b	OGLE-2016-BLG-1227L	NaN	NaN	NaN	2020-03	2020	
1	GJ 480 b	b	GJ 480	NaN	HIP 61706	TIC 399119319	2020-08	2020	

	pl_name	pl_letter	hostname	hd_name	hip_name	tic_id	disc_pubdate	disc_year	di
2	Kepler-276 c	c	Kepler-276	NaN	NaN	TIC 138213510	2014-02	2013	
3	Kepler-829 b	b	Kepler-829	NaN	NaN	TIC 123451768	2016-05	2016	
						TIC			

In [5]:

```python
# basic details on the dataframe
confirmed_df.info(verbose=True)
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4884 entries, 0 to 4883
Data columns (total 373 columns):
 #    Column                 Dtype  
---   ------                 -----  
 0    pl_name                object 
 1    pl_letter              object 
 2    hostname               object 
 3    hd_name                object 
 4    hip_name               object 
 5    tic_id                 object 
 6    disc_pubdate           object 
 7    disc_year              int64  
 8    discoverymethod        object 
 9    disc_locale            object 
 10   disc_facility          object 
 11   disc_instrument        object 
 12   disc_telescope         object 
 13   disc_refname           object 
 14   ra                     float64
 15   rastr                  object 
 16   dec                    float64
 17   decstr                 object 
 18   glon                   float64
 19   glat                   float64
 20   elon                   float64
 21   elat                   float64
 22   ra_reflink             object 
 23   pl_orbper              float64
 24   pl_orbpererr1          float64
 25   pl_orbpererr2          float64
 26   pl_orbperlim           float64
 27   pl_orbperstr           object 
 28   pl_orbper_reflink      object 
 29   pl_orblpererr1         float64
 30   pl_orblper             float64
 31   pl_orblpererr2         float64
 32   pl_orblperlim          float64
 33   pl_orblperstr          object 
 34   pl_orblper_reflink     object 
 35   pl_orbsmax             float64
 36   pl_orbsmaxerr1         float64
 37   pl_orbsmaxerr2         float64
 38   pl_orbsmaxlim          float64
 39   pl_orbsmaxstr          object 
 40   pl_orbsmax_reflink     object 
 41   pl_orbincl             float64
 42   pl_orbinclerr1         float64
 43   pl_orbinclerr2         float64
 44   pl_orbincllim          float64
 45   pl_orbinclstr          object 
 46   pl_orbincl_reflink     object 
```

47	pl_orbtper	float64
48	pl_orbtpererr1	float64
49	pl_orbtpererr2	float64
50	pl_orbtperlim	float64
51	pl_orbtperstr	object
52	pl_orbtper_reflink	object
53	pl_orbeccen	float64
54	pl_orbeccenerr1	float64
55	pl_orbeccenerr2	float64
56	pl_orbeccenlim	float64
57	pl_orbeccenstr	object
58	pl_orbeccen_reflink	object
59	pl_eqt	float64
60	pl_eqterr1	float64
61	pl_eqterr2	float64
62	pl_eqtlim	float64
63	pl_eqtstr	object
64	pl_eqt_reflink	object
65	pl_occdep	float64
66	pl_occdeperr1	float64
67	pl_occdeperr2	float64
68	pl_occdeplim	float64
69	pl_occdepstr	object
70	pl_occdep_reflink	object
71	pl_insol	float64
72	pl_insolerr1	float64
73	pl_insolerr2	float64
74	pl_insollim	float64
75	pl_insolstr	object
76	pl_insol_reflink	object
77	pl_dens	float64
78	pl_denserr1	float64
79	pl_denserr2	float64
80	pl_denslim	float64
81	pl_densstr	object
82	pl_dens_reflink	object
83	pl_trandep	float64
84	pl_trandeperr1	float64
85	pl_trandeperr2	float64
86	pl_trandeplim	float64
87	pl_trandepstr	object
88	pl_trandep_reflink	object
89	pl_tranmid	float64
90	pl_tranmiderr1	float64
91	pl_tranmiderr2	float64
92	pl_tranmidlim	float64
93	pl_tranmidstr	object
94	sy_pmdec	float64
95	sy_pmdecerr1	float64
96	sy_pmdecerr2	float64
97	sy_pmdecstr	object
98	sy_plx	float64
99	sy_plxerr1	float64
100	sy_plxerr2	float64
101	sy_plxstr	object
102	sy_plx_reflink	object
103	sy_dist	float64
104	sy_disterr1	float64
105	sy_disterr2	float64
106	sy_diststr	object
107	sy_dist_reflink	object
108	sy_bmag	float64
109	sy_bmagerr1	float64
110	sy_bmagerr2	float64
111	sy_bmagstr	object
112	sy_bmag_reflink	object
113	sy_vmag	float64
114	sy_vmagerr1	float64
115	sy_vmagerr2	float64
116	sy_vmagstr	object

117	sy_vmag_reflink	object
118	sy_jmag	float64
119	sy_jmagerr1	float64
120	sy_jmagerr2	float64
121	sy_jmagstr	object
122	sy_jmag_reflink	object
123	sy_hmag	float64
124	sy_hmagerr1	float64
125	sy_hmagerr2	float64
126	sy_hmagstr	object
127	sy_hmag_reflink	object
128	sy_kmag	float64
129	sy_kmagerr1	float64
130	sy_kmagerr2	float64
131	sy_kmagstr	object
132	sy_kmag_reflink	object
133	sy_umag	float64
134	sy_umagerr1	float64
135	sy_umagerr2	float64
136	sy_umagstr	object
137	sy_umag_reflink	object
138	sy_rmag	float64
139	sy_rmagerr1	float64
140	sy_rmagerr2	float64
141	sy_rmagstr	object
142	sy_rmag_reflink	object
143	sy_imag	float64
144	sy_imagerr1	float64
145	sy_imagerr2	float64
146	sy_imagstr	object
147	sy_imag_reflink	object
148	sy_zmag	float64
149	sy_zmagerr1	float64
150	sy_zmagerr2	float64
151	sy_zmagstr	object
152	sy_zmag_reflink	object
153	sy_w1mag	float64
154	sy_w1magerr1	float64
155	sy_w1magerr2	float64
156	sy_w1magstr	object
157	sy_w1mag_reflink	object
158	sy_w2mag	float64
159	sy_w2magerr1	float64
160	sy_w2magerr2	float64
161	sy_w2magstr	object
162	sy_w2mag_reflink	object
163	sy_w3mag	float64
164	sy_w3magerr1	float64
165	sy_w3magerr2	float64
166	sy_w3magstr	object
167	sy_w3mag_reflink	object
168	sy_w4mag	float64
169	sy_w4magerr1	float64
170	sy_w4magerr2	float64
171	sy_w4magstr	object
172	sy_w4mag_reflink	object
173	sy_gmag	float64
174	sy_gmagerr1	float64
175	sy_gmagerr2	float64
176	sy_gmagstr	object
177	sy_gmag_reflink	object
178	sy_gaiamag	float64
179	sy_gaiamagerr1	float64
180	sy_gaiamagerr2	float64
181	sy_gaiamagstr	object
182	sy_gaiamag_reflink	object
183	sy_tmag	float64
184	sy_tmagerr1	float64
185	sy_tmagerr2	float64
186	sy_tmagstr	object

187	sy_tmag_reflink	object
188	pl_controv_flag	int64
189	pl_orbtper_systemref	object
190	pl_tranmid_systemref	object
191	st_metratio	object
192	st_spectype	object
193	st_spectype_reflink	object
194	sy_kepmag	float64
195	sy_kepmagerr1	float64
196	sy_kepmagerr2	float64
197	sy_kepmagstr	float64
198	sy_kepmag_reflink	object
199	st_rotp	float64
200	st_rotperr1	float64
201	st_rotperr2	float64
202	st_rotplim	float64
203	st_rotpstr	object
204	st_rotp_reflink	object
205	pl_projobliq	float64
206	pl_projobliqerr1	float64
207	pl_projobliqerr2	float64
208	pl_projobliqlim	float64
209	pl_projobliqstr	object
210	pl_projobliq_reflink	object
211	gaia_id	object
212	cb_flag	int64
213	pl_tranmid_reflink	object
214	pl_trandur	float64
215	pl_trandurerr1	float64
216	pl_trandurerr2	float64
217	pl_trandurlim	float64
218	pl_trandurstr	object
219	pl_trandur_reflink	object
220	pl_rvamp	float64
221	pl_rvamperr1	float64
222	pl_rvamperr2	float64
223	pl_rvamplim	float64
224	pl_rvampstr	object
225	pl_rvamp_reflink	object
226	pl_radj	float64
227	pl_radjerr1	float64
228	pl_radjerr2	float64
229	pl_radjlim	float64
230	pl_radjstr	object
231	pl_radj_reflink	object
232	pl_rade	float64
233	pl_radeerr1	float64
234	pl_radeerr2	float64
235	pl_radelim	float64
236	pl_radestr	object
237	pl_rade_reflink	object
238	pl_ratror	float64
239	pl_ratrorerr1	float64
240	pl_ratrorerr2	float64
241	pl_ratrorlim	float64
242	pl_ratrorstr	object
243	pl_ratror_reflink	object
244	pl_ratdor	float64
245	pl_trueobliq	float64
246	pl_trueobliqerr1	float64
247	pl_trueobliqerr2	float64
248	pl_trueobliqlim	float64
249	pl_trueobliqstr	object
250	pl_trueobliq_reflink	object
251	sy_icmag	float64
252	sy_icmagerr1	float64
253	sy_icmagerr2	float64
254	sy_icmagstr	object
255	sy_icmag_reflink	object
256	dkin_flag	int64

257	pl_ratdorerr1	float64
258	pl_ratdorerr2	float64
259	pl_ratdorlim	float64
260	pl_ratdorstr	object
261	pl_ratdor_reflink	object
262	pl_imppar	float64
263	pl_impparerr1	float64
264	pl_impparerr2	float64
265	pl_impparlim	float64
266	pl_impparstr	object
267	pl_imppar_reflink	object
268	pl_bmassj	float64
269	pl_bmassjerr1	float64
270	pl_bmassjerr2	float64
271	pl_bmassjlim	float64
272	pl_bmassjstr	object
273	pl_bmassj_reflink	object
274	pl_bmasse	float64
275	pl_bmasseerr1	float64
276	pl_bmasseerr2	float64
277	pl_bmasselim	float64
278	pl_bmassestr	object
279	pl_bmasse_reflink	object
280	pl_bmassprov	object
281	st_teff	float64
282	st_tefferr1	float64
283	st_tefferr2	float64
284	st_tefflim	float64
285	st_teffstr	object
286	st_teff_reflink	object
287	st_met	float64
288	st_meterr1	float64
289	st_meterr2	float64
290	st_metlim	float64
291	st_metstr	object
292	st_met_reflink	object
293	st_radv	float64
294	st_radverr1	float64
295	st_radverr2	float64
296	st_radvlim	float64
297	st_radvstr	object
298	st_radv_reflink	object
299	st_vsin	float64
300	st_vsinerr1	float64
301	st_vsinerr2	float64
302	st_vsinlim	float64
303	st_vsinstr	object
304	st_vsin_reflink	object
305	st_lum	float64
306	st_lumerr1	float64
307	st_lumerr2	float64
308	st_lumlim	float64
309	st_lumstr	object
310	st_lum_reflink	object
311	st_logg	float64
312	st_loggerr1	float64
313	st_loggerr2	float64
314	st_logglim	float64
315	st_loggstr	object
316	st_logg_reflink	object
317	st_age	float64
318	st_ageerr1	float64
319	st_ageerr2	float64
320	st_agelim	float64
321	st_agestr	object
322	st_age_reflink	object
323	st_mass	float64
324	st_masserr1	float64
325	st_masserr2	float64
326	st_masslim	float64

```
 327   st_massstr              object
 328   st_mass_reflink         object
 329   st_dens                 float64
 330   st_denserr1             float64
 331   st_denserr2             float64
 332   st_denslim              float64
 333   st_densstr              object
 334   st_dens_reflink         object
 335   st_rad                  float64
 336   st_raderr1              float64
 337   st_raderr2              float64
 338   st_radlim               float64
 339   st_radstr               object
 340   st_rad_reflink          object
 341   ttv_flag                int64
 342   ptv_flag                int64
 343   tran_flag               int64
 344   rv_flag                 int64
 345   ast_flag                int64
 346   obm_flag                int64
 347   micro_flag              int64
 348   etv_flag                int64
 349   ima_flag                int64
 350   pul_flag                int64
 351   sy_snum                 int64
 352   sy_pnum                 int64
 353   sy_mnum                 int64
 354   st_nphot                int64
 355   st_nrvc                 int64
 356   st_nspec                int64
 357   pl_nespec               int64
 358   pl_ntranspec            int64
 359   pl_nnotes               float64
 360   sy_pm                   float64
 361   sy_pmerr1               float64
 362   sy_pmerr2               float64
 363   sy_pmstr                object
 364   sy_pm_reflink           object
 365   sy_pmra                 float64
 366   sy_pmraerr1             float64
 367   sy_pmraerr2             float64
 368   sy_pmrastr              object
 369   x                       float64
 370   y                       float64
 371   z                       float64
 372   htm20                   int64
dtypes: float64(216), int64(23), object(134)
```

Columns from the dataset that are of use to this project
From the documentation at: https://exoplanetarchive.ipac.caltech.edu/TAP/tables and https://exoplanetarchive.ipac.caltech.edu/docs/API_kepcandidate_columns.html are:

pl_name : Char
The planet name

pl_letter : Char
The planet lettering system (single aplphabetical digit)

hostname : Char
The host(Star) name for the planet discovered

tic_id : Char
The target identification number, for use downloading the specific Pixel Image data

disc_pubdate : Char
When the discovered planet was published

disc_year : int
The year the planet was discovered

discoverymethod : Char
The method used to discover the planet, Transit Microlensing etc,,

disc_locale : Char
From where the discovery was made, from the ground or from space

disc_facility : Char
The facility that was incharge of the instrument that made the discovery

disc_instrument : Char
The type of instrument used, For Kepler it is the Kepler CCD Array

disc_telescope : Char
The name of the telescope, or camera that made the discovery

pl_orbper : double
Is the orbital period of the planet discovered

pl_eqt : double
The equilibrium temperature (Kelvin) of the planet discovered

pl_dens : double
The density (g/cm3) of the planet discovered

pl_trandur : double
The tranist duration in days

pl_radj : double
The discovered planets Juptier Radius

pl_rade : double
The discovered planets Earth Radius

pl_bmasse : double
The discovered planets Earth mass (Planet Mass.sin(i)/sin(i))

st_age : double
The stellar age in Gyr

st_mass : double
The Stellar Mass (Solar Mass)

tran_flag : int
Detected by Transits

sy_dist : double
The system distance in (pc) Parsec approx 3.26 light-years

From the 373 initial columns the above columns will give some additional insights into the discovery of exoplanets by during the Kepler mission.

Although another data set containing the Kepler ID (KIC) will be used to download the pixel images
needed for the flux readings, this dataset will be used to lable and verify the positive confirmations.

In [6]:

```
# Reindex only the columns of possible interest to the initial exploritory
confirmed_df = confirmed_df.reindex(columns=['pl_name','pl_letter','hostnam
                                    'discoverymethod','disc_locale','disc_facilit
                                    'disc_telescope','pl_orbper','pl_eqt','pl_der
                                    'pl_rade','pl_bmasse','st_age','st_mass','tra

confirmed_df
```

Out[6]:

	pl_name	pl_letter	hostname	tic_id	disc_pubdate	disc_year	discoverymethod	d
0	OGLE-2016-BLG-1227L b	b	OGLE-2016-BLG-1227L	NaN	2020-03	2020	Microlensing	
1	GJ 480 b	b	GJ 480	TIC 399119319	2020-08	2020	Radial Velocity	
2	Kepler-276 c	c	Kepler-276	TIC 138213510	2014-02	2013	Transit	
3	Kepler-829 b	b	Kepler-829	TIC 123451768	2016-05	2016	Transit	
4	K2-283 b	b	K2-283	TIC 266017624	2018-12	2018	Transit	
...	...	...	...	...	...	...	...	
4879	2M0437 b	b	2M0437	TIC 125843782	2021-10	2021	Imaging	
4880	HATS-74 A b	b	HATS-74 A	TIC 219189765	2021-12	2021	Transit	
4881	HATS-75 b	b	HATS-75	TIC 44737596	2021-12	2021	Transit	
4882	HATS-76 b	b	HATS-76	TIC 170849515	2021-12	2021	Transit	

	pl_name	pl_letter	hostname	tic_id	disc_pubdate	disc_year	discoverymethod	d
4883	HATS-77 b	b	HATS-77	TIC 11561667	2021-12	2021	Transit	

In [7]:

```
# simple desciptive statistics of the new dataframe
confirmed_df.describe()
```

Out[7]:

	disc_year	pl_orbper	pl_eqt	pl_dens	pl_trandur	pl_radj	pl_rad
count	4884.000000	4.726000e+03	3710.000000	4779.000000	3638.000000	4868.000000	4870.00000
mean	2015.251229	8.863196e+04	909.615374	4.278715	3.929477	0.492018	5.51216
std	4.114448	5.849702e+06	456.408995	22.959142	2.647875	0.472840	5.29870
min	1989.000000	9.070629e-02	50.000000	0.030000	0.112700	0.026000	0.29600
25%	2014.000000	4.488917e+00	571.250000	1.460000	2.292250	0.155000	1.73100
50%	2016.000000	1.161783e+01	817.500000	2.590000	3.270500	0.241000	2.70000
75%	2018.000000	3.978237e+01	1141.750000	4.570000	4.797750	0.995750	11.14750
max	2021.000000	4.020000e+08	4050.000000	1290.000000	53.600000	6.900000	77.34200

In [8]:

```
# simple desciptive statistics of the new dataframe
confirmed_df.dtypes
```

Out[8]:

```
pl_name              object
pl_letter            object
hostname             object
tic_id               object
disc_pubdate         object
disc_year             int64
discoverymethod      object
disc_locale          object
disc_facility        object
disc_instrument      object
disc_telescope       object
pl_orbper           float64
pl_eqt              float64
pl_dens             float64
pl_trandur          float64
pl_radj             float64
pl_rade             float64
pl_bmasse           float64
st_age              float64
st_mass             float64
tran_flag             int64
sy_dist             float64
dtype: object
```

In [9]:

```
# simple desciptive statistics of the new dataframe
confirmed_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4884 entries, 0 to 4883
Data columns (total 22 columns):
 #   Column           Non-Null Count  Dtype  
---  ------           --------------  -----  
 0   pl_name          4884 non-null   object 
 1   pl_letter        4884 non-null   object 
```

```
 2   hostname         4884 non-null   object
 3   tic_id           4751 non-null   object
 4   disc_pubdate     4884 non-null   object
 5   disc_year        4884 non-null   int64
 6   discoverymethod  4884 non-null   object
 7   disc_locale      4884 non-null   object
 8   disc_facility    4884 non-null   object
 9   disc_instrument  4884 non-null   object
 10  disc_telescope   4884 non-null   object
 11  pl_orbper        4726 non-null   float64
 12  pl_eqt           3710 non-null   float64
 13  pl_dens          4779 non-null   float64
 14  pl_trandur       3638 non-null   float64
 15  pl_radj          4868 non-null   float64
 16  pl_rade          4870 non-null   float64
 17  pl_bmasse        4862 non-null   float64
 18  st_age           3971 non-null   float64
 19  st_mass          4880 non-null   float64
 20  tran_flag        4884 non-null   int64
 21  sy_dist          4869 non-null   float64
dtypes: float64(10), int64(2), object(10)
```

Discovery Methods & Discovery Telescopes

Checking the discovery methods for expolanets. The Machine learning aspect of the project will focus on the Transit method discovered by the Kepler Telescope.

In [10]:

```python
# group and counting planets by discovery method
disc_method = confirmed_df.copy()
disc_method = disc_method.groupby(['discoverymethod'])['pl_name'].count()
disc_method = disc_method.to_frame()
disc_method.reset_index(inplace=True)
disc_method.rename(columns={'pl_name':'count'}, inplace=True)
disc_method['%'] = (disc_method['count'] / disc_method['count'].sum())*100
disc_method
```

Out[10]:

	discoverymethod	count	%
0	Astrometry	1	0.020475
1	Disk Kinematics	1	0.020475
2	Eclipse Timing Variations	16	0.327600
3	Imaging	55	1.126126
4	Microlensing	120	2.457002
5	Orbital Brightness Modulation	9	0.184275
6	Pulsar Timing	7	0.143325
7	Pulsation Timing Variations	2	0.040950
8	Radial Velocity	899	18.407043
9	Transit	3752	76.822277
10	Transit Timing Variations	22	0.450450

Checking to see which instrunments have discovered how many exoplanets

In [11]:

```
# group and counting planets by discovery instrument
disc_telescope = confirmed_df.copy()
disc_telescope = disc_telescope.groupby(['disc_telescope'])['pl_name'].cour
disc_telescope = disc_telescope.to_frame()
disc_telescope.reset_index(inplace=True)
disc_telescope.rename(columns={'pl_name':'count'}, inplace=True)
disc_telescope['%'] = (disc_telescope['count'] / disc_telescope['count'].su
disc_telescope = disc_telescope.sort_values(by='%', ascending=False)
disc_telescope.reset_index(drop= True, inplace=True)
disc_telescope
```

Out[11]:

	disc_telescope	count	%
0	0.95 m Kepler Telescope	3180	65.110565
1	Canon 200mm f/1.8L	222	4.545455
2	Multiple Telescopes	217	4.443079
3	3.6 m ESO Telescope	200	4.095004
4	0.1 m TESS Telescope	175	3.583129
...	...	...	...
70	6.5 m Magellan I Baade Telescope	1	0.020475
71	1.55 m Wyeth Telescope	1	0.020475
72	4.20 m William Herschel Telescope	1	0.020475
73	4 m ESO Vista Telescope	1	0.020475
74	2.7m Harlan J. Smith Telescope	1	0.020475

75 rows × 3 columns

In [58]:

```
table_1 = disc_telescope.copy()
table_1 = table_1.iloc[:5]


# plot a table
fig, ax = plt.subplots()

# hide the axis
fig.patch.set_visible(False)
ax.axis('off')
ax.axis('tight')

#create the table
table = ax.table(cellText=table_1.values, colLabels = table_1.columns, loc=

# disply and save the table
name = "telescope %"
plt.title(name, y=1.0, pad=-80)
fig.tight_layout()
plt.savefig('./graphs/disc_telescope_percent.jpg', bbox_inches="tight", dpi
plt.show()
```

telescope %

disc_telescope	count	%
0.95 m Kepler Telescope	3180	65.1105651105651
Canon 200mm f/1.8L	222	4.545454545454546
Multiple Telescopes	217	4.443079443079443
3.6 m ESO Telescope	200	4.095004095004095
0.1 m TESS Telescope	175	3.583128583128583

Checking the combination of instrument and detection types

In [13]:

```
# group and counting planets by discovery instrument and method
disc_combination = confirmed_df.copy()
disc_combination = disc_combination.groupby(['disc_telescope', 'discoveryme
disc_combination = disc_combination.to_frame()
disc_combination.reset_index(inplace=True)
disc_combination.rename(columns={'pl_name':'count'}, inplace=True)
disc_combination['%'] = (disc_combination['count'] / disc_combination['cour
disc_combination = disc_combination.sort_values(by='%', ascending=False)
disc_combination.reset_index(drop= True, inplace=True)
disc_combination
```

Out[13]:

	disc_telescope	discoverymethod	count	%
0	0.95 m Kepler Telescope	Transit	3150	64.496314
1	Canon 200mm f/1.8L	Transit	220	4.504505
2	3.6 m ESO Telescope	Radial Velocity	200	4.095004
3	Multiple Telescopes	Radial Velocity	174	3.562654
4	0.1 m TESS Telescope	Transit	172	3.521704
...	...	...	...	...
98	3.9 m Anglo-Australian Telescope	Transit	1	0.020475
99	4 m ESO Vista Telescope	Imaging	1	0.020475
100	4.20 m William Herschel Telescope	Imaging	1	0.020475
101	6.5 m Magellan I Baade Telescope	Imaging	1	0.020475
102	2.7m Harlan J. Smith Telescope	Radial Velocity	1	0.020475

103 rows × 4 columns

In [14]:

```
table_2 = disc_combination.copy()
table_2 = table_2.iloc[:5]
# plot a table of the stances count and percentages
fig, ax = plt.subplots()

# hide the axis
fig.patch.set_visible(False)
ax.axis('off')
ax.axis('tight')

#create the table
table = ax.table(cellText=table_2.values, colLabels = table_2.columns, loc=

# disply and save the table
name = "telescope & method %"
plt.title(name, y=1.0, pad=-80)
fig.tight_layout()
plt.savefig('./graphs/disc_telescope_method_percent.jpg', bbox_inches="tigh
plt.show()
```

telescope & method %

disc_telescope	discoverymethod	count	%
0.95 m Kepler Telescope	Transit	3150	64.4963144963145
Canon 200mm f/1.8L	Transit	220	4.504504504504505
3.6 m ESO Telescope	Radial Velocity	200	4.095004095004095
Multiple Telescopes	Radial Velocity	174	3.562653562653563
0.1 m TESS Telescope	Transit	172	3.5217035217035217

Checking the combination of instrument, facility and method

In [15]:

```
# group and counting planets by discovery instrument and method
disc_combination_1 = confirmed_df.copy()
disc_combination_1 = disc_combination_1.groupby(['disc_facility','disc_tele
disc_combination_1 = disc_combination_1.to_frame()
disc_combination_1.reset_index(inplace=True)
disc_combination_1.rename(columns={'pl_name':'count'}, inplace=True)
disc_combination_1['%'] = (disc_combination_1['count'] / disc_combination_1
disc_combination_1 = disc_combination_1.sort_values(by='%', ascending=False
disc_combination_1.reset_index(drop= True, inplace=True)
disc_combination_1
```

Out[15]:

	disc_facility	disc_telescope	discoverymethod	count	%
0	Kepler	0.95 m Kepler Telescope	Transit	2673	54.729730
1	K2	0.95 m Kepler Telescope	Transit	477	9.766585
2	La Silla Observatory	3.6 m ESO Telescope	Radial Velocity	200	4.095004
3	Transiting Exoplanet Survey Satellite (TESS)	0.1 m TESS Telescope	Transit	172	3.521704

	disc_facility	disc_telescope	discoverymethod	count	%
4	W. M. Keck Observatory	10 m Keck I Telescope	Radial Velocity	171	3.501229
...	...	...	...	...	...
116	Infrared Survey Facility	1.4 m IRSF Telescope	Imaging	1	0.020475
117	Las Campanas Observatory	6.5 m Magellan I Baade Telescope	Imaging	1	0.020475
118	Paranal Observatory	8.2 m ESO VLT UT1 Antu Telescope	Astrometry	1	0.020475
119	Paranal Observatory	8.2 m ESO VLT UT1 Antu Telescope	Transit	1	0.020475
120	Acton Sky Portal Observatory	279mm RASA-11 wide-field telescope	Transit	1	0.020475

In [16]:

```
table_3 = disc_combination.copy()
table_3 = table_3.iloc[:5]

# plot a table
fig, ax = plt.subplots()

# hide the axis
fig.patch.set_visible(False)
ax.axis('off')
ax.axis('tight')

#create the table
table = ax.table(cellText=table_3.values, colLabels = table_3.columns, loc=

# disply and save the table
name = "facility, telescope & method %"
plt.title(name, y=1.0, pad=-80)
fig.tight_layout()
plt.savefig('./graphs/disc_facil_telescope_method_percent.jpg', bbox_inches
plt.show()
```

facility, telescope & method %

disc_telescope	discoverymethod	count	%
0.95 m Kepler Telescope	Transit	3150	64.4963144963145
Canon 200mm f/1.8L	Transit	220	4.504504504504505
3.6 m ESO Telescope	Radial Velocity	200	4.095004095004095
Multiple Telescopes	Radial Velocity	174	3.562653562653563
0.1 m TESS Telescope	Transit	172	3.5217035217035217

The Kepler Mission and Telescope combination has discovered 54.73% of all exoplanets in the MAST archives
and will be the focus of the rest of the exploritory analysis

In [17]:

```
# filtering out just Kepler discovered exoplanets
kep_df = confirmed_df.copy()
kep_df = kep_df[kep_df['disc_facility'].str.contains("Kepler", na=False) &
                kep_df['discoverymethod'].str.contains("Transit", na=False)
               ~kep_df['discoverymethod'].str.contains("Variations", na=Fal


kep_df.drop_duplicates(subset=['pl_name'])
kep_df
```

Out[17]:

	pl_name	pl_letter	hostname	tic_id	disc_pubdate	disc_year	discoverymethod	d
2	Kepler-276 c	c	Kepler-276	TIC 138213510	2014-02	2013	Transit	
3	Kepler-829 b	b	Kepler-829	TIC 123451768	2016-05	2016	Transit	
5	Kepler-477 b	b	Kepler-477	TIC 158633329	2016-05	2016	Transit	
20	Kepler-1390 b	b	Kepler-1390	TIC 264508835	2016-05	2016	Transit	
21	Kepler-393 c	c	Kepler-393	TIC 159580535	2014-03	2014	Transit	
...	...	...	...	...	...	...	...	
4872	Kepler-1896 b	b	Kepler-1896	TIC 394175334	2021-11	2021	Transit	
4873	Kepler-1897 b	b	Kepler-1897	TIC 138645719	2021-11	2021	Transit	
4874	Kepler-1899 b	b	Kepler-1899	TIC 137348756	2021-11	2021	Transit	
4875	Kepler-1904 b	b	Kepler-1904	TIC 299089587	2021-11	2021	Transit	
4876	Kepler-1905 b	b	Kepler-1905	TIC 63362238	2021-11	2021	Transit	

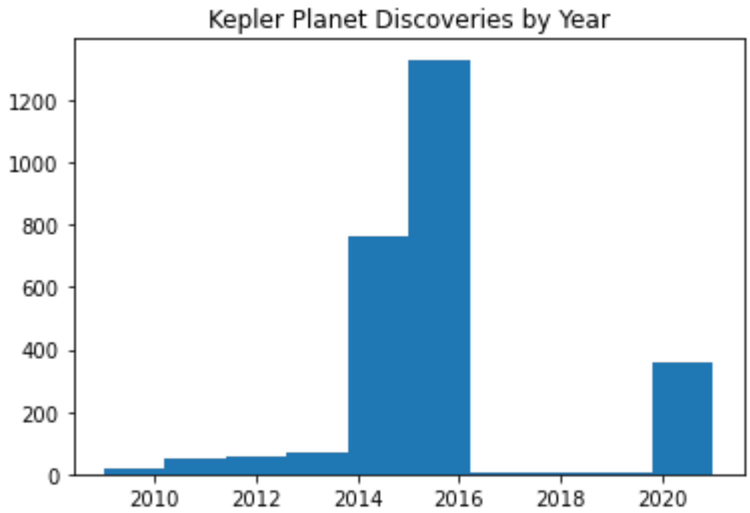
2673 rows × 22 columns

Discovery Years

Plotting the years of discovered planets by Kepler Mission, Kepler launched on March 7, 2009 and ran until October 30, 2018

In [18]:

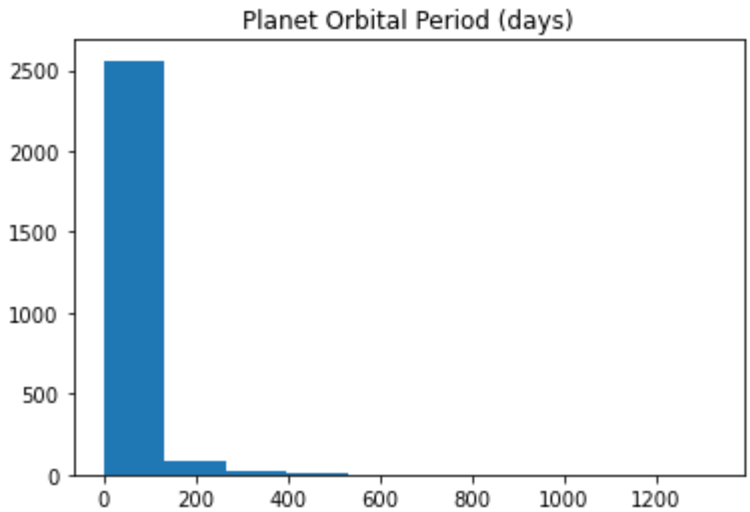
```
plt.subplot()
plt.hist(kep_df['disc_year'])
plt.title('Kepler Planet Discoveries by Year')


plt.savefig('./graphs/kep_disc_by_year.jpg', bbox_inches="tight", dpi=450)
plt.show()
```



Exoplanet Orbital Period

Plotting the distributions of Orbital Periods

In [19]:
```
plt.subplot()
plt.hist(kep_df['pl_orbper'])
plt.title('Planet Orbital Period (days)')


plt.savefig('./graphs/planet_orbital_period.jpg', bbox_inches="tight", dpi=
plt.show()
```



In [20]:
```
orbital_period_stats = kep_df['pl_orbper'].describe()
orbital_period_stats = orbital_period_stats.to_frame()
orbital_period_stats.reset_index(inplace=True)
orbital_period_stats.rename(columns={'index':'Stat', 'pl_orbper':'Value'},
orbital_period_stats
```

Out[20]:

	Stat	Value
0	count	2673.000000
1	mean	30.753326
2	std	75.882054
3	min	0.355007

	Stat	Value
4	25%	5.194922
5	50%	11.555300
6	75%	27.082511

In [21]:

```
# plot a table
fig, ax = plt.subplots()

# hide the axis
fig.patch.set_visible(False)
ax.axis('off')
ax.axis('tight')

#create the table
table = ax.table(cellText=orbital_period_stats.values, colLabels = orbital_

# disply and save the table
name = "Orbital Period Stats"
plt.title(name, y=1.0, pad=-60)
fig.tight_layout()
plt.savefig('./graphs/orbital_period_stats.jpg', bbox_inches="tight", dpi=3
plt.show()
```

Orbital Period Stats

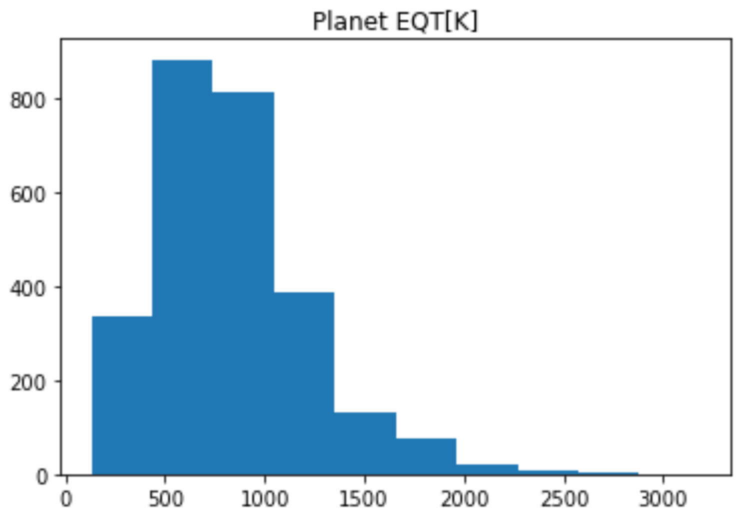
Stat	Value
count	2673.0
mean	30.753326332424272
std	75.88205397827873
min	0.35500744
25%	5.19492202
50%	11.55530021
75%	27.082511
max	1322.3

Exoplanet Equilibrium Temperatures

Plotting the distribution of temperatures in Kelvin

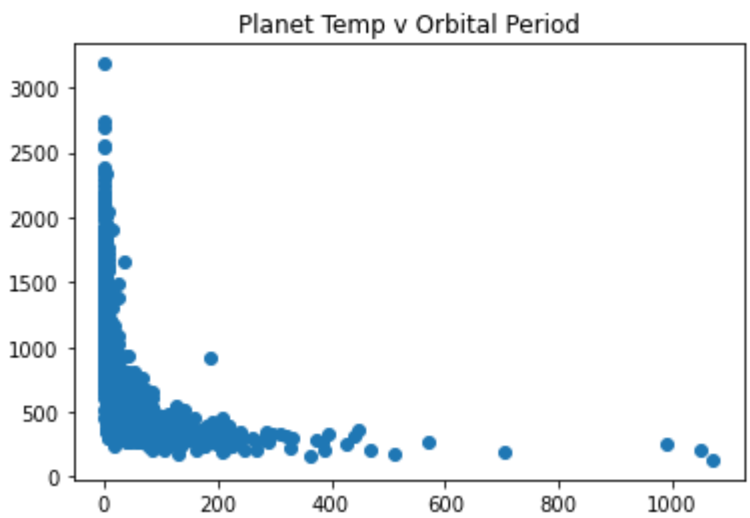
In [22]:

```
plt.subplot()
plt.hist(kep_df['pl_eqt'])
plt.title('Planet EQT[K]')


plt.savefig('./graphs/planet_temps.jpg', bbox_inches="tight", dpi=450)
plt.show()
```



In [23]:

```
# scatter plot of orbital period and temp
x = kep_df['pl_orbper']
y = kep_df['pl_eqt']

plt.scatter(x, y)
plt.title('Planet Temp v Orbital Period')
plt.savefig('./graphs/planet_temp_orbital_period_scatter.jpg', bbox_inches=
plt.show()
```

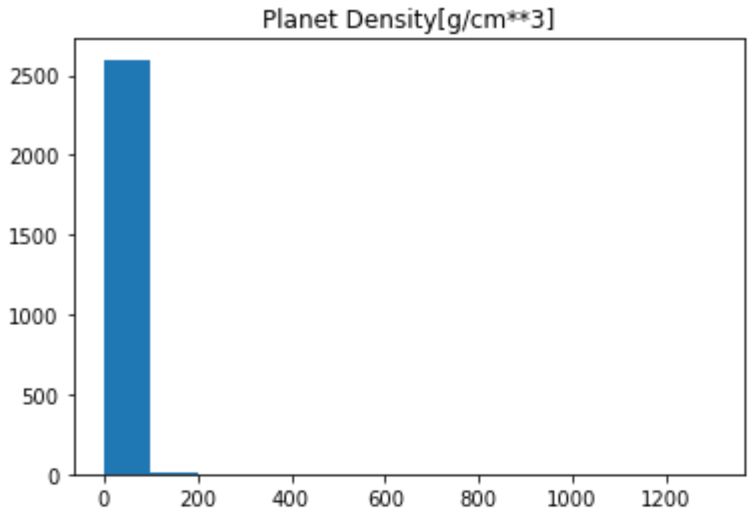


Exoplanet Density

Plotting the density of exoplanets in (g/cm**3)

In [24]:

```
# hostogram of planet desnities
plt.subplot()
data = kep_df['pl_dens']
plt.hist(data, bins=np.arange(min(data), max(data) + 100, 100))
plt.title('Planet Density[g/cm**3]')


plt.savefig('./graphs/planet_density.jpg', bbox_inches="tight", dpi=450)
plt.show()
```



In [25]:

```
max_density = kep_df['pl_dens'].max()
max_density
```

Out[25]: 1290.0

In [26]:

```
min_density = kep_df['pl_dens'].min()
min_density
```

Out[26]: 0.03

In [27]:

```
mode_density = kep_df['pl_dens'].mode()
mode_density
```

Out[27]:

```
0    4.99
1    5.71
dtype: float64
```

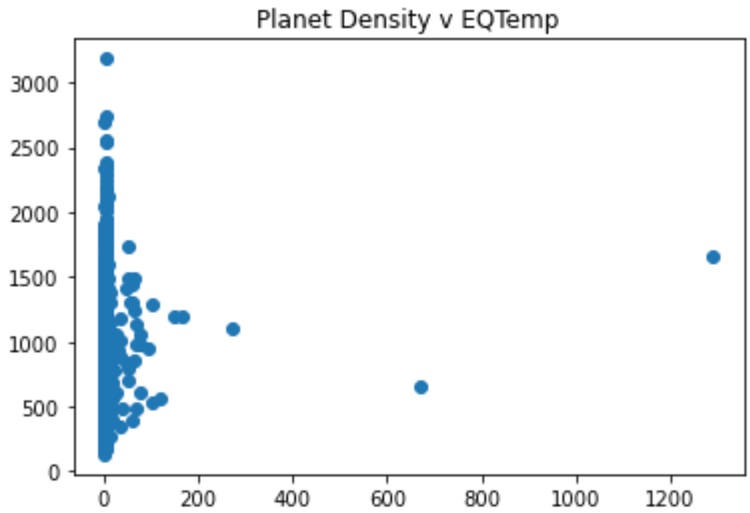
In [28]:

```
median_density = kep_df['pl_dens'].median()
median_density
```

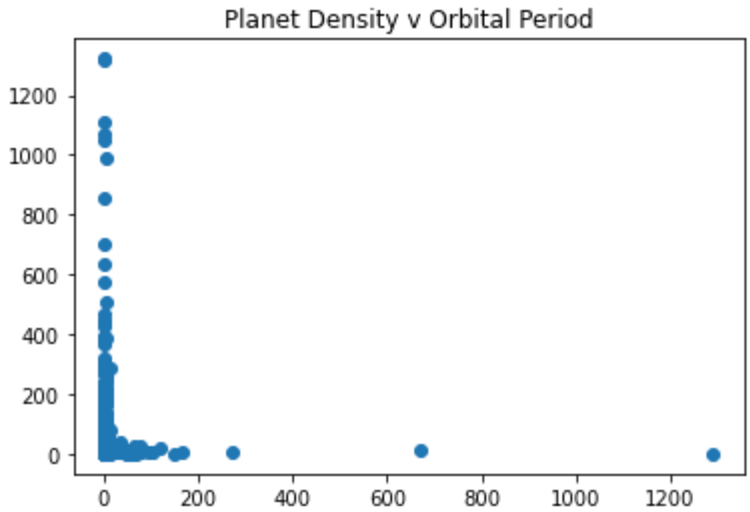
Out[28]: 3.065

In [29]:

```
# scatter plot of density and temp
x = kep_df['pl_dens']
y = kep_df['pl_eqt']

plt.scatter(x, y)
plt.title('Planet Density v EQTemp')
plt.savefig('./graphs/planet_density_scatter.jpg', bbox_inches="tight", dpi
plt.show()
```



In [30]:
```
# scatter plot of density and orbital period
x = kep_df['pl_dens']
y = kep_df['pl_orbper']

plt.scatter(x, y)
plt.title('Planet Density v Orbital Period')
plt.savefig('./graphs/planet_density_orb_period_scatter.jpg', bbox_inches='
plt.show()
```



In [31]:
```
earth_density_stats = kep_df['pl_dens'].describe()
earth_density_stats = earth_density_stats.to_frame()
earth_density_stats.reset_index(inplace=True)
earth_density_stats.rename(columns={'index':'Stat', 'pl_dens':'Value'}, inp
earth_density_stats
```

Out[31]:

	Stat	Value
0	count	2606.000000
1	mean	5.092675
2	std	29.954707
3	min	0.030000
4	25%	2.050000
5	50%	3.065000

	Stat	Value
6	75%	4.770000

In [32]:

```
# plot a table
fig, ax = plt.subplots()

# hide the axis
fig.patch.set_visible(False)
ax.axis('off')
ax.axis('tight')

#create the table
table = ax.table(cellText=earth_density_stats.values, colLabels = earth_der

# disply and save the table
name = "Planet Density Stats"
plt.title(name, y=1.0, pad=-60)
fig.tight_layout()
plt.savefig('./graphs/planet_density_stats.jpg', bbox_inches="tight", dpi=3
plt.show()
```

Planet Density Stats

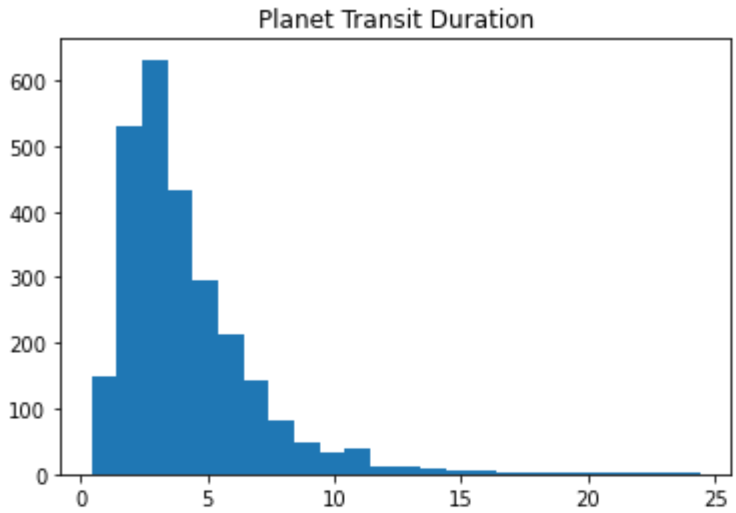
Stat	Value
count	2606.0
mean	5.0926749808134995
std	29.954707152509364
min	0.03
25%	2.05
50%	3.065
75%	4.77
max	1290.0

Exoplanet Transit Durations

Transit duration is set in days and is the duration for the transit across the star.

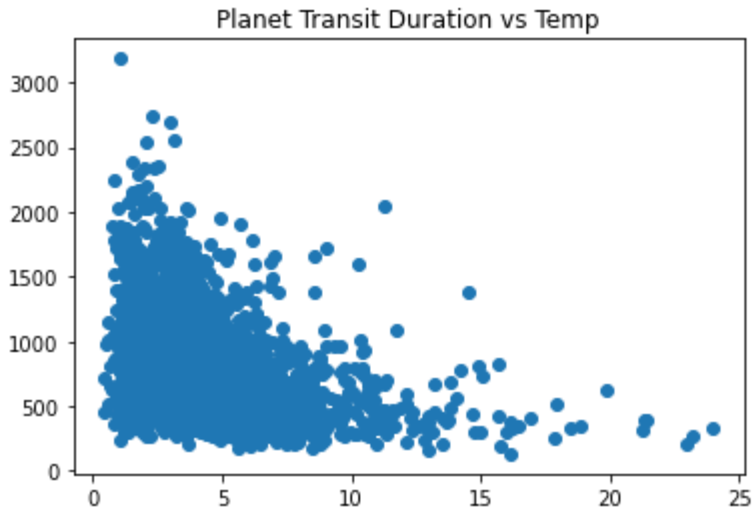
In [33]:

```
plt.subplot()
data = kep_df['pl_trandur']
plt.hist(data, bins=np.arange(min(data), max(data) + 1, 1))
plt.title('Planet Transit Duration')


plt.savefig('./graphs/kep_transit_durations.jpg', bbox_inches="tight", dpi=
plt.show()
```



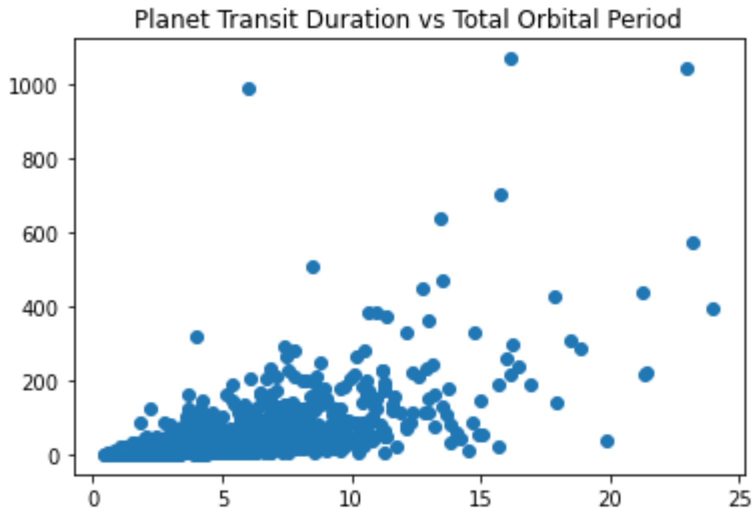
In [34]:

```
# checking for any relationship between the duration and the temperature
x = kep_df['pl_trandur']
y = kep_df['pl_eqt']

plt.scatter(x, y)
plt.title('Planet Transit Duration vs Temp')
plt.savefig('./graphs/planet_trans_and_temp_scatter.jpg', bbox_inches="tigh
plt.show()
```



In [35]:

```
# checking for any relationship between the duration and the period
x = kep_df['pl_trandur']
y = kep_df['pl_orbper']

plt.scatter(x, y)
plt.title('Planet Transit Duration vs Total Orbital Period')
plt.savefig('./graphs/planet_trans_and_orbper_scatter.jpg', bbox_inches="ti
plt.show()
```

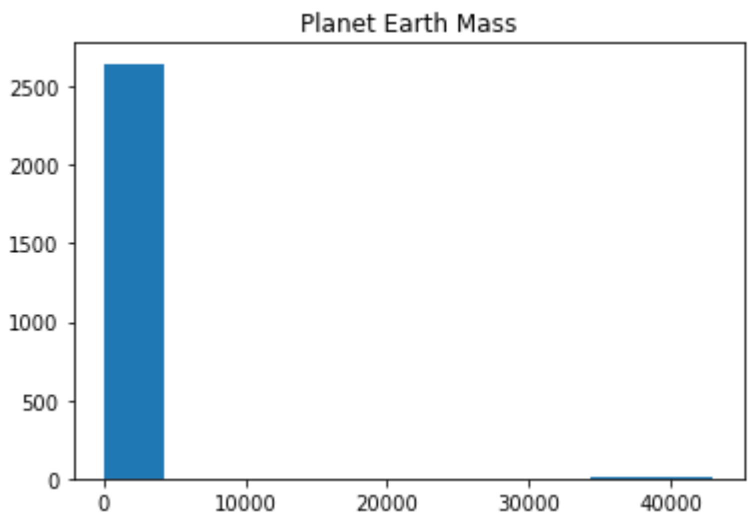


Exoplanet Earth Mass

The Earth Mass equivalent of the exoplanet found

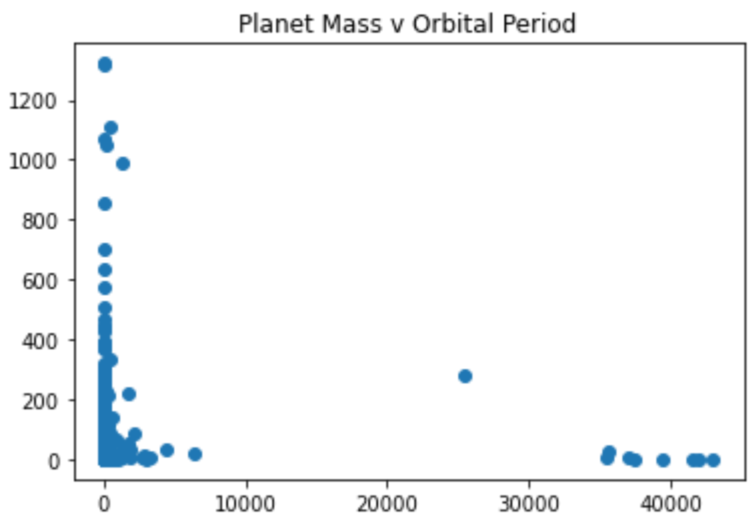
In [36]:

```
plt.subplot()
data = kep_df['pl_bmasse']
plt.hist(data)
plt.title('Planet Earth Mass')


plt.savefig('./graphs/kep_earth_mass.jpg', bbox_inches="tight", dpi=450)
plt.show()
```



In [37]:

```
# scatter plot of earth mass and orbital period
x = kep_df['pl_bmasse']
y = kep_df['pl_orbper']

plt.scatter(x, y)
plt.title('Planet Mass v Orbital Period')
plt.savefig('./graphs/planet_mass_orbper_scatter.jpg', bbox_inches="tight",
plt.show()
```



In [38]:
```
earth_mass_mode = kep_df['pl_bmasse'].mode()
earth_mass_mode
```

Out[38]:
```
0    3.33
dtype: float64
```

In [39]:
```
earth_mass_min = kep_df['pl_bmasse'].min()
earth_mass_min
```

Out[39]: 0.0374

In [40]:
```
earth_mass_max = kep_df['pl_bmasse'].max()
earth_mass_max
```

Out[40]: 43000.0

In [41]:
```
earth_mass_mean = kep_df['pl_bmasse'].mean()
earth_mass_mean
```

Out[41]: 156.51208530673722

In [42]:
```
earth_mass_median = kep_df['pl_bmasse'].median()
earth_mass_median
```

Out[42]: 5.39

In [43]:
```
earth_mass_stats = kep_df['pl_bmasse'].describe()
earth_mass_stats = earth_mass_stats.to_frame()
earth_mass_stats.reset_index(inplace=True)
earth_mass_stats.rename(columns={'index':'Stat', 'pl_bmasse':'Value'}, inpl
earth_mass_stats
```

Out[43]:

	Stat	Value
0	count	2657.000000
1	mean	156.512085
2	std	2204.692704

	Stat	Value
3	min	0.037400
4	25%	2.890000
5	50%	5.390000
6	75%	9.010000

In [44]:

```
# plot a table
fig, ax = plt.subplots()

# hide the axis
fig.patch.set_visible(False)
ax.axis('off')
ax.axis('tight')

#create the table
table = ax.table(cellText=earth_mass_stats.values, colLabels = earth_mass_s

# disply and save the table
name = "Earth Mass Stats"
plt.title(name, y=1.0, pad=-60)
fig.tight_layout()
plt.savefig('./graphs/earth_mass_stats.jpg', bbox_inches="tight", dpi=300)
plt.show()
```

Earth Mass Stats

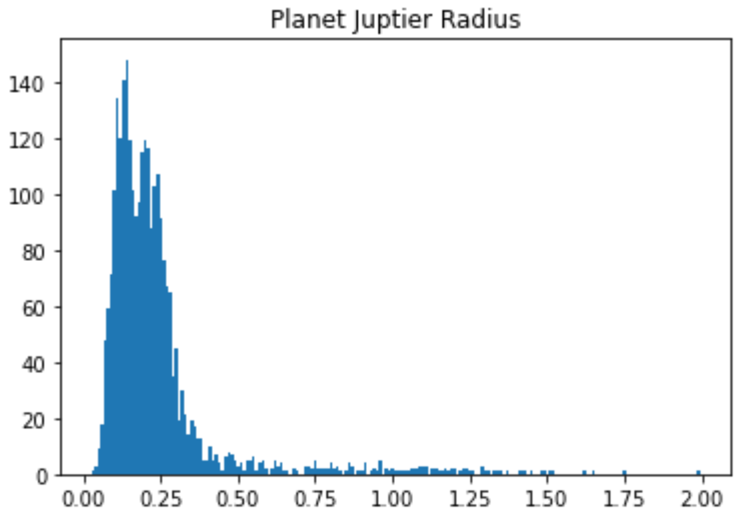
Stat	Value
count	2657.0
mean	156.51208530673722
std	2204.6927036880425
min	0.0374
25%	2.89
50%	5.39
75%	9.01
max	43000.0

Exoplanet in Jupiter Radius

Plot the Jupiter radius distributions between the exoplanets discovered

In [45]:

```
plt.subplot()
data = kep_df['pl_radj']
plt.hist(data, bins=np.arange(min(data), max(data) + 0.01, 0.01))
plt.title('Planet Juptier Radius')


plt.savefig('./graphs/kep_jupiter_radius.jpg', bbox_inches="tight", dpi=450
plt.show()
```



In [46]:

```
jupiter_radius_stats = kep_df['pl_radj'].describe()
jupiter_radius_stats = jupiter_radius_stats.to_frame()
jupiter_radius_stats.reset_index(inplace=True)
jupiter_radius_stats.rename(columns={'index':'Stat', 'pl_radj':'Value'}, ir
jupiter_radius_stats
```

Out[46]:

	Stat	Value
0	count	2673.000000
1	mean	0.238951
2	std	0.206891
3	min	0.026000
4	25%	0.133000
5	50%	0.192000
6	75%	0.256000
7	max	1.990000

In [47]:

```
# plot a table
fig, ax = plt.subplots()

# hide the axis
fig.patch.set_visible(False)
ax.axis('off')
ax.axis('tight')

#create the table
table = ax.table(cellText=jupiter_radius_stats.values, colLabels = jupiter_

# disply and save the table
name = "Juptier Radius Stats"
plt.title(name, y=1.0, pad=-60)
fig.tight_layout()
plt.savefig('./graphs/jupiter_radius_stats.jpg', bbox_inches="tight", dpi=3
plt.show()
```

Juptier Radius Stats

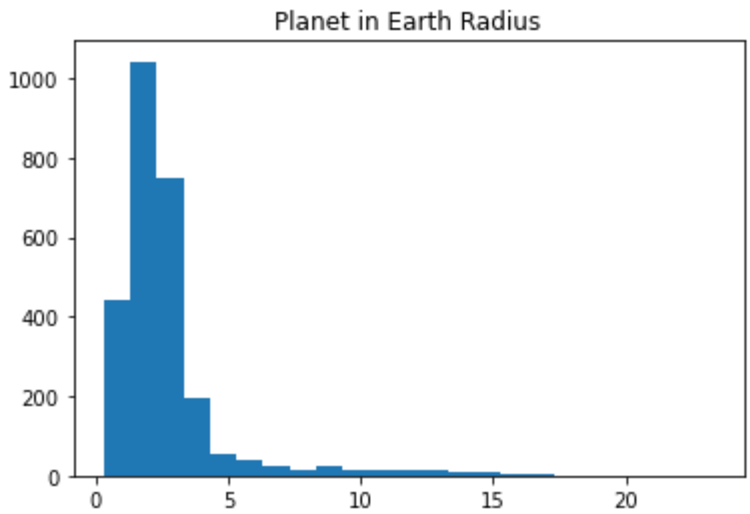
Stat	Value
count	2673.0
mean	0.23895099139543613
std	0.20689140294422023
min	0.026
25%	0.133
50%	0.192
75%	0.256
max	1.99

Exoplanet in Earth Radius

Plot the Earth radius distributions between the exoplanets discovered

In [48]:

```
plt.subplot()
data = kep_df['pl_rade']
plt.hist(data, bins=np.arange(min(data), max(data) + 1, 1))
plt.title('Planet in Earth Radius')


plt.savefig('./graphs/kep_earth_radius.jpg', bbox_inches="tight", dpi=450)
plt.show()
```



In [49]:

```
earth_radius_stats = kep_df['pl_rade'].describe()
earth_radius_stats = earth_radius_stats.to_frame()
earth_radius_stats.reset_index(inplace=True)
earth_radius_stats.rename(columns={'index':'Stat', 'pl_rade':'Value'}, inpl
earth_radius_stats
```

Out[49]:

	Stat	Value
0	count	2673.000000
1	mean	2.678313
2	std	2.319057

	Stat	Value
3	min	0.296000
4	25%	1.490000
5	50%	2.150000
6	75%	2.870000

In [50]:

```
# plot a table
fig, ax = plt.subplots()

# hide the axis
fig.patch.set_visible(False)
ax.axis('off')
ax.axis('tight')

#create the table
table = ax.table(cellText=earth_radius_stats.values, colLabels = earth_radi

# disply and save the table
name = "Earth Radius Stats"
plt.title(name, y=1.0, pad=-60)
fig.tight_layout()
plt.savefig('./graphs/earth_radius_stats.jpg', bbox_inches="tight", dpi=300
plt.show()
```

Earth Radius Stats

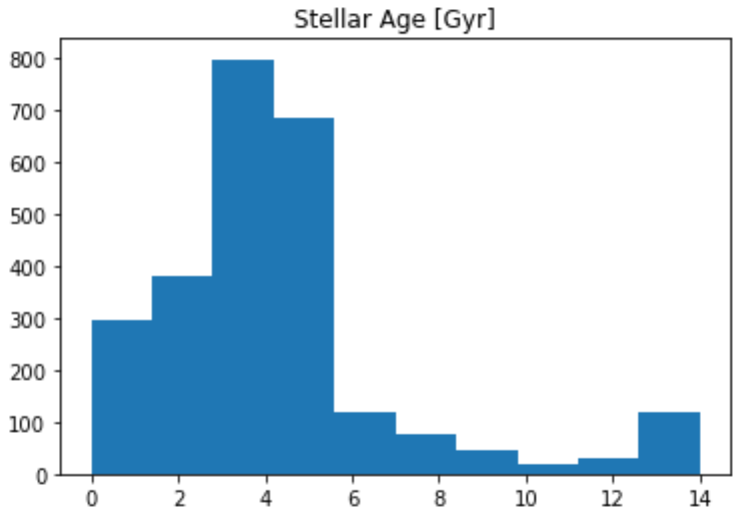
Stat	Value
count	2673.0
mean	2.6783127572016494
std	2.3190566893338156
min	0.296
25%	1.49
50%	2.15
75%	2.87
max	22.31

Stellar Age

The stellar ages of stars hosting known exoplanets

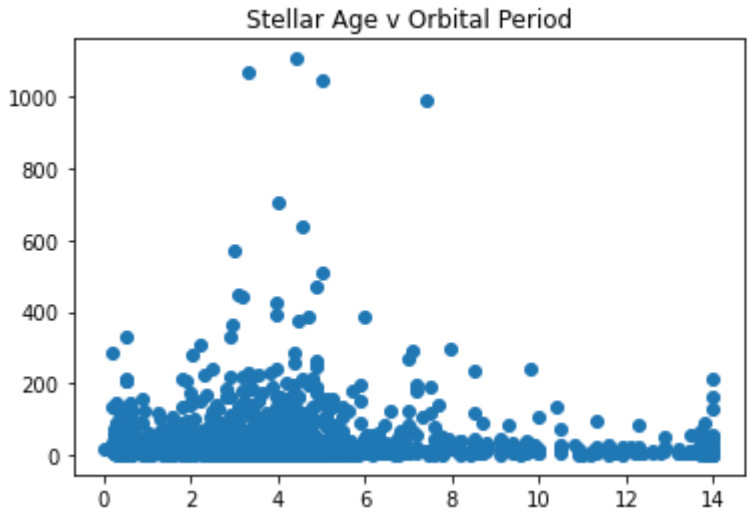
In [51]:

```
# plot a histogram to visualise distribution
plt.subplot()
data = kep_df['st_age']
plt.hist(data)
plt.title('Stellar Age [Gyr]')


plt.savefig('./graphs/stellar_age.jpg', bbox_inches="tight", dpi=450)
plt.show()
```



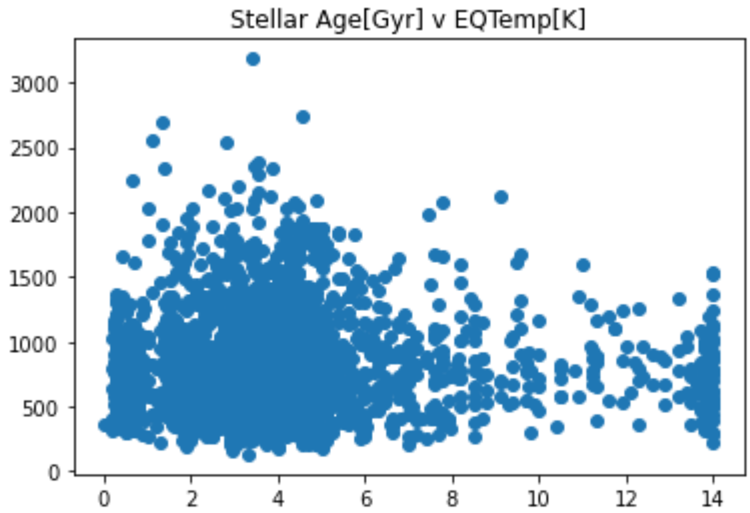
In [52]:

```
# scatter plot of stellar age and orbital period
x = kep_df['st_age']
y = kep_df['pl_orbper']

plt.scatter(x, y)
plt.title('Stellar Age v Orbital Period')
plt.savefig('./graphs/stellar_age_orbper_scatter.jpg', bbox_inches="tight",
plt.show()
```



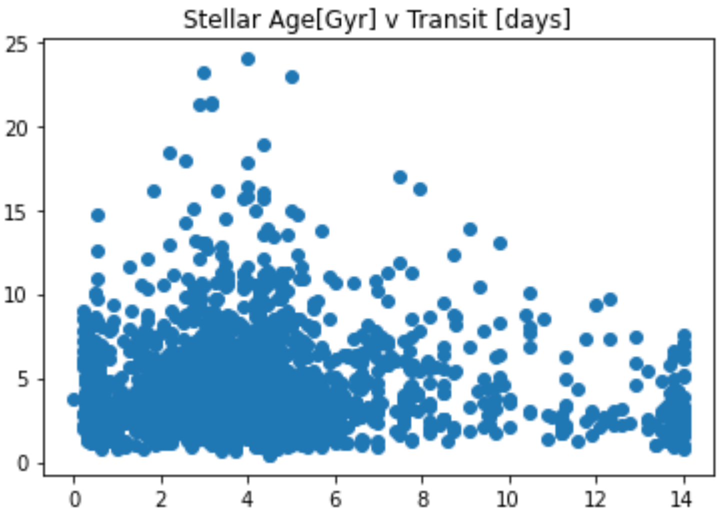
In [53]:

```
# scatter plot of stellar age and temp
x = kep_df['st_age']
y = kep_df['pl_eqt']

plt.scatter(x, y)
plt.title('Stellar Age[Gyr] v EQTemp[K]')
plt.savefig('./graphs/stellar_age_temp_scatter.jpg', bbox_inches="tight", c
plt.show()
```



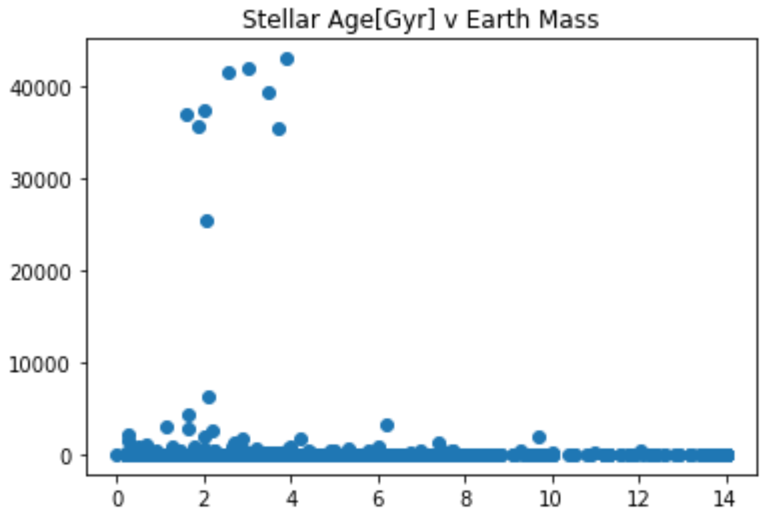
In [54]:

```
# scatter plot of stellar age and transit duration
x = kep_df['st_age']
y = kep_df['pl_trandur']

plt.scatter(x, y)
plt.title('Stellar Age[Gyr] v Transit [days]')
plt.savefig('./graphs/stellar_age_transit_scatter.jpg', bbox_inches="tight'
plt.show()
```



In [55]:

```
# scatter plot of stellar age and transit duration
x = kep_df['st_age']
y = kep_df['pl_bmasse']

plt.scatter(x, y)
plt.title('Stellar Age[Gyr] v Earth Mass')
plt.savefig('./graphs/stellar_age_earthMass_scatter.jpg', bbox_inches="tigh
plt.show()
```



In [56]:

```
stellar_age_stats = kep_df['st_age'].describe()
stellar_age_stats = stellar_age_stats.to_frame()
stellar_age_stats.reset_index(inplace=True)
stellar_age_stats.rename(columns={'index':'Stat', 'st_age':'Value'}, inplac
stellar_age_stats
```

Out[56]:

	Stat	Value
0	count	2568.000000
1	mean	4.233917
2	std	2.932827
3	min	0.000000
4	25%	2.690000
5	50%	3.890000
6	75%	4.680000
7	max	14.000000

In [57]:

```
# plot a table
fig, ax = plt.subplots()

# hide the axis
fig.patch.set_visible(False)
ax.axis('off')
ax.axis('tight')

#create the table
table = ax.table(cellText=stellar_age_stats.values, colLabels = stellar_age

# disply and save the table
name = "Stellar Age[Gyr] Stats"
plt.title(name, y=1.0, pad=-60)
fig.tight_layout()
plt.savefig('./graphs/stellar_age_stats.jpg', bbox_inches="tight", dpi=300)
plt.show()
```

Stellar Age[Gyr] Stats

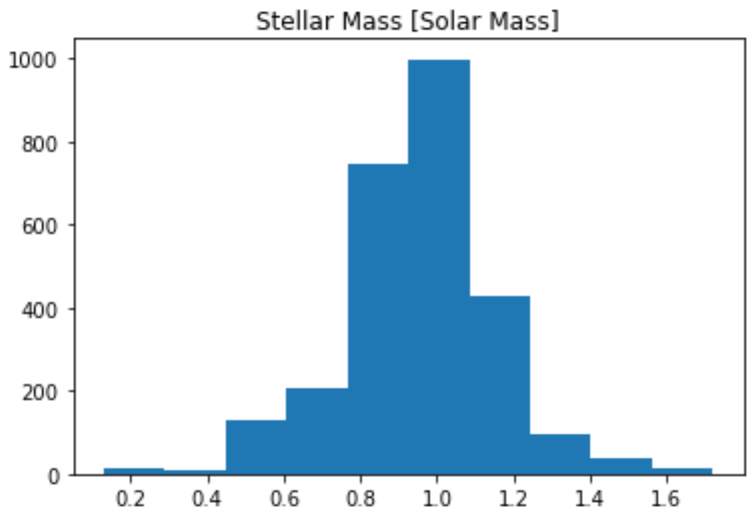
Stat	Value
count	2568.0
mean	4.233917445482864
std	2.9328274812653143
min	0.0
25%	2.69
50%	3.89
75%	4.68
max	14.0

Stellar Mass

The stellar mass (also known as Solar Mass) of stars hosting known exoplanets

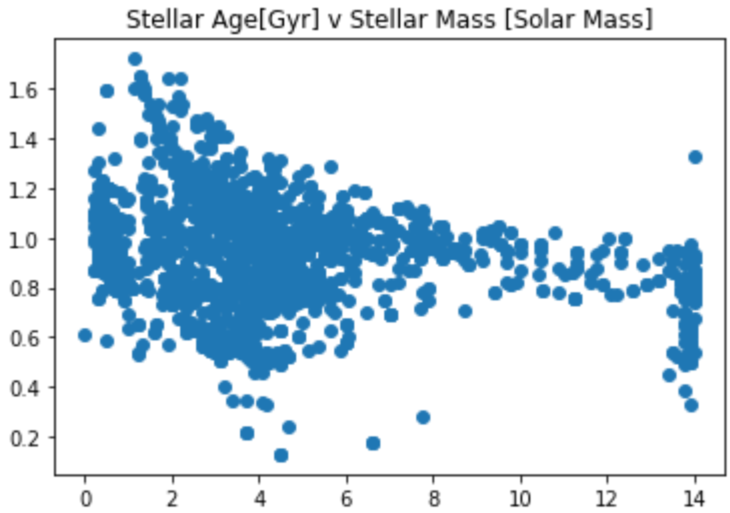
In [59]:

```
# plot a histogram to visualise distribution
plt.subplot()
data = kep_df['st_mass']
plt.hist(data)
plt.title('Stellar Mass [Solar Mass]')


plt.savefig('./graphs/stellar_mass.jpg', bbox_inches="tight", dpi=450)
plt.show()
```



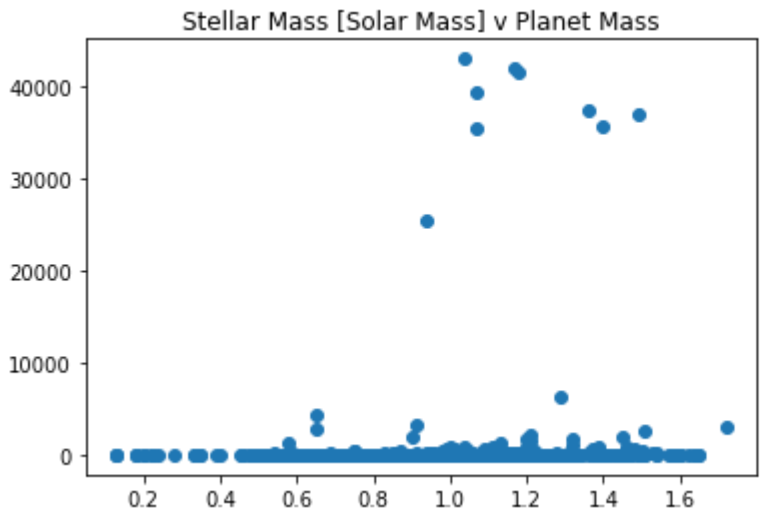
In [60]:

```
# scatter plot of stellar age and stellar mass
x = kep_df['st_age']
y = kep_df['st_mass']

plt.scatter(x, y)
plt.title('Stellar Age[Gyr] v Stellar Mass [Solar Mass]')
plt.savefig('./graphs/stellar_age_stealler_mass_scatter.jpg', bbox_inches='
plt.show()
```



In [63]:

```
# scatter plot of planet mass and stellar mass
x = kep_df['st_mass']
y = kep_df['pl_bmasse']

plt.scatter(x, y)
plt.title('Stellar Mass [Solar Mass] v Planet Mass')
plt.savefig('./graphs/stealler_mass__planet_mass_scatter.jpg', bbox_inches=
plt.show()
```



In [64]:

```
stellar_mass_stats = kep_df['st_mass'].describe()
stellar_mass_stats = stellar_mass_stats.to_frame()
stellar_mass_stats.reset_index(inplace=True)
stellar_mass_stats.rename(columns={'index':'Stat', 'st_mass':'Value'}, inpl
stellar_mass_stats
```

Out[64]:

	Stat	Value
0	count	2673.000000
1	mean	0.951104
2	std	0.196974
3	min	0.130000
4	25%	0.840000
5	50%	0.960000

	Stat	Value
6	75%	1.070000

In [65]:

```
# plot a table
fig, ax = plt.subplots()

# hide the axis
fig.patch.set_visible(False)
ax.axis('off')
ax.axis('tight')

#create the table
table = ax.table(cellText=stellar_mass_stats.values, colLabels = stellar_ma

# disply and save the table
name = "Stellar Mass [Solar Mass] Stats"
plt.title(name, y=1.0, pad=-60)
fig.tight_layout()
plt.savefig('./graphs/stellar_mass_stats.jpg', bbox_inches="tight", dpi=300
plt.show()
```

Stellar Mass [Solar Mass] Stats

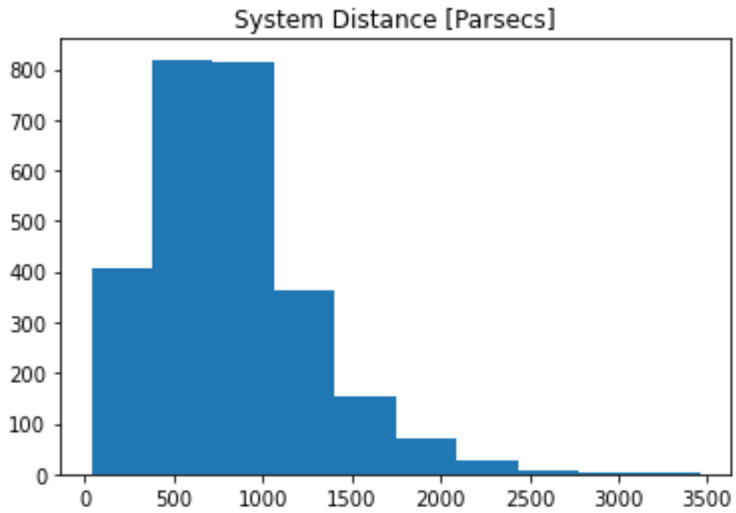
Stat	Value
count	2673.0
mean	0.9511036288814071
std	0.19697358813228316
min	0.13
25%	0.84
50%	0.96
75%	1.07
max	1.72

System Distance

The Parsec distance to the host of the exoplanet

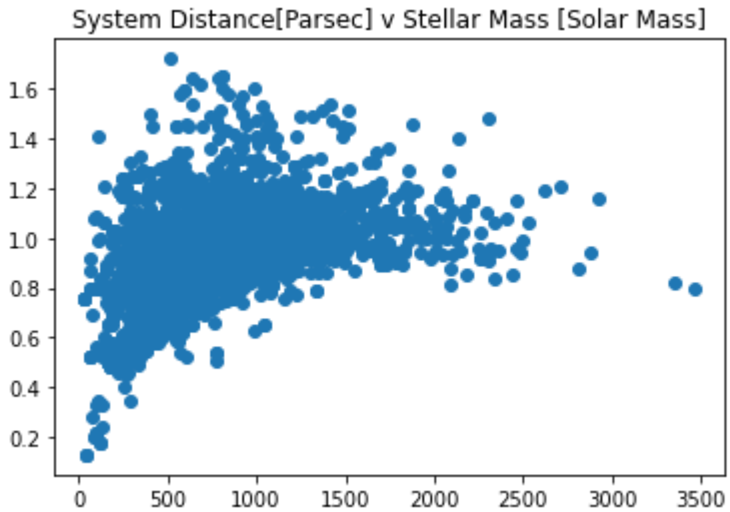
In [66]:

```
# plot a histogram to visualise distribution
plt.subplot()
data = kep_df['sy_dist']
plt.hist(data)
plt.title('System Distance [Parsecs]')


plt.savefig('./graphs/system_distance.jpg', bbox_inches="tight", dpi=450)
plt.show()
```



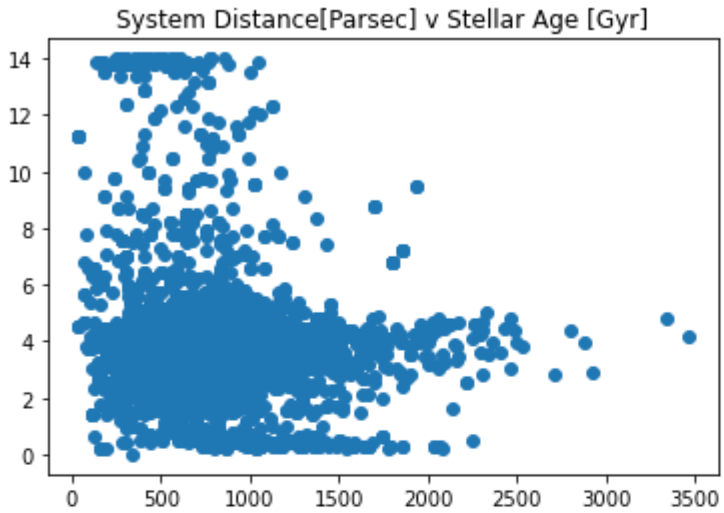
In [67]:

```
# scatter plot of system distance and stellar mass
x = kep_df['sy_dist']
y = kep_df['st_mass']

plt.scatter(x, y)
plt.title('System Distance[Parsec] v Stellar Mass [Solar Mass]')
plt.savefig('./graphs/system_distance_steller_mass_scatter.jpg', bbox_inche
plt.show()
```



In [68]:

```
# scatter plot of system distance and stellar age
x = kep_df['sy_dist']
y = kep_df['st_age']

plt.scatter(x, y)
plt.title('System Distance[Parsec] v Stellar Age [Gyr]')
plt.savefig('./graphs/system_distance_steller_age_scatter.jpg', bbox_inches
plt.show()
```



In [69]:

```
stellar_distance_stats = kep_df['sy_dist'].describe()
stellar_distance_stats = stellar_distance_stats.to_frame()
stellar_distance_stats.reset_index(inplace=True)
stellar_distance_stats.rename(columns={'index':'Stat', 'sy_dist':'Value'},
stellar_distance_stats
```

Out[69]:

	Stat	Value
0	count	2663.000000
1	mean	818.424865
2	std	449.332459
3	min	36.439600
4	25%	486.360000
5	50%	769.096000
6	75%	1047.030000
7	max	3460.510000

In [70]:

```
# plot a table
fig, ax = plt.subplots()

# hide the axis
fig.patch.set_visible(False)
ax.axis('off')
ax.axis('tight')

#create the table
table = ax.table(cellText=stellar_distance_stats.values, colLabels = stella

# disply and save the table
name = "System Distance [Parsecs] Stats"
plt.title(name, y=1.0, pad=-60)
fig.tight_layout()
plt.savefig('./graphs/system_distance_stats.jpg', bbox_inches="tight", dpi=
plt.show()
```

System Distance [Parsecs] Stats

Stat	Value
count	2663.0
mean	818.4248652612065
std	449.332458762824
min	36.4396
25%	486.36
50%	769.096
75%	1047.03
max	3460.51

In [ ]: