In [20]:
```
import seaborn as sns
import numpy as np
```

In [2]:
```
df=sns.load_dataset('iris')
```

In [3]:
```
df
```

Out[3]:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	virginica
146	6.3	2.5	5.0	1.9	virginica
147	6.5	3.0	5.2	2.0	virginica
148	6.2	3.4	5.4	2.3	virginica
149	5.9	3.0	5.1	1.8	virginica

150 rows × 5 columns

In [4]:
```
df.shape
```

Out[4]:
```
(150, 5)
```

In [29]:
```
df.describe()
```

Out[29]:

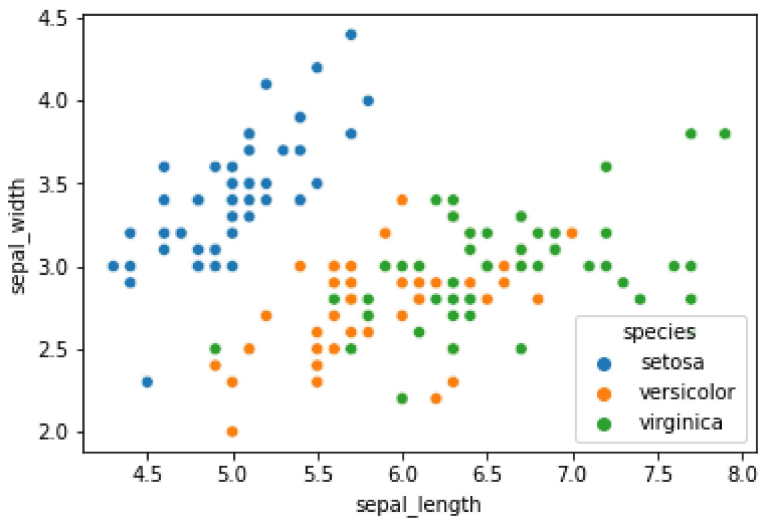
	sepal_length	sepal_width	petal_length	petal_width
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.057333	3.758000	1.199333
std	0.828066	0.435866	1.765298	0.762238
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.350000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

In [5]:
```
df.isnull().sum()
```

Out[5]:
```
sepal_length    0
sepal_width     0
petal_length    0
petal_width     0
species         0
dtype: int64
```

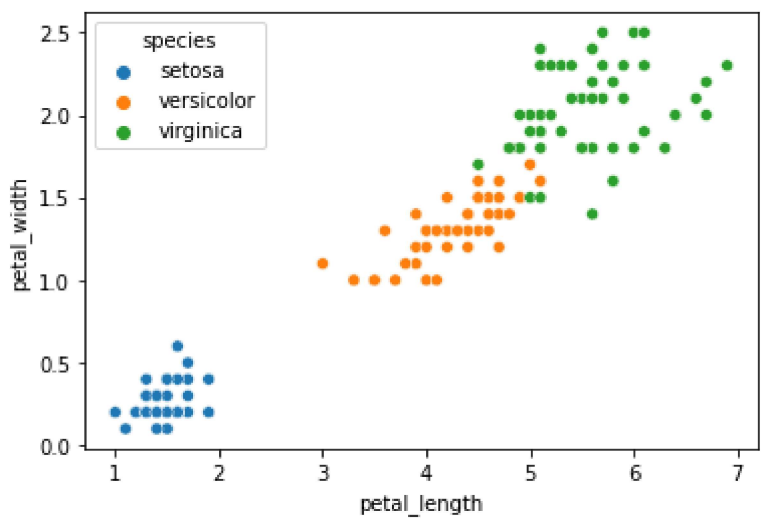
In [6]:
```
sns.scatterplot(x='sepal_length',y='sepal_width',hue='species',data=df)
```

Out[6]:
```
<AxesSubplot:xlabel='sepal_length', ylabel='sepal_width'>
```



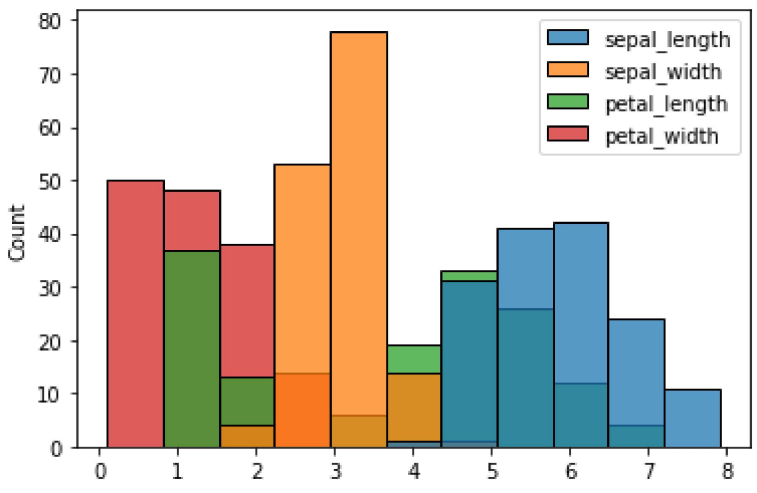
In [7]:
```
sns.scatterplot(x='petal_length',y='petal_width',hue='species',data=df)
```

Out[7]:
```
<AxesSubplot:xlabel='petal_length', ylabel='petal_width'>
```



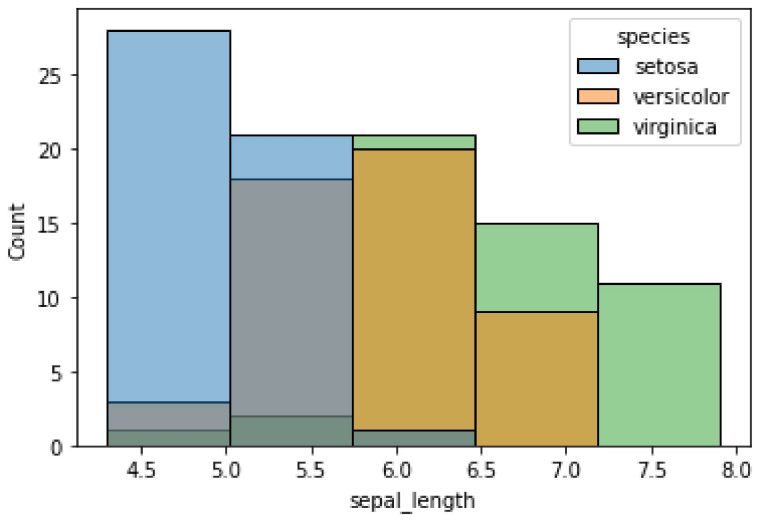
In [8]:
```
sns.histplot(df)
```

Out[8]: <AxesSubplot:ylabel='Count'>



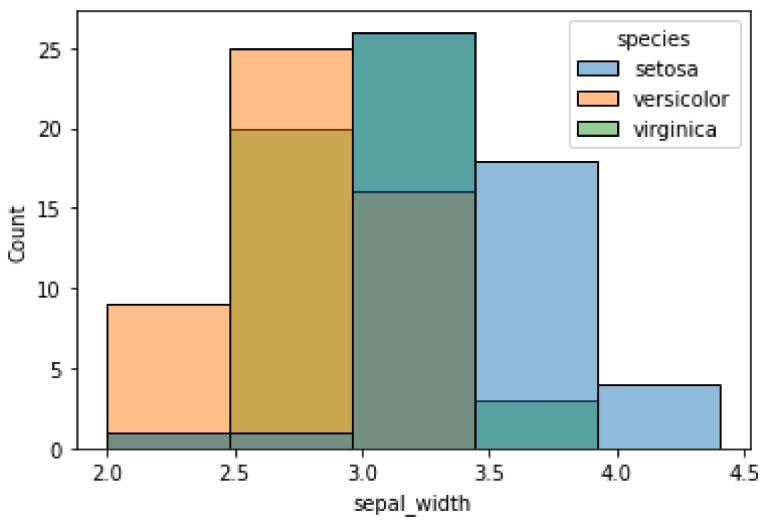
In [9]:
```
sns.histplot(df,x='sepal_length',hue='species',bins=5)
```

Out[9]: <AxesSubplot:xlabel='sepal_length', ylabel='Count'>



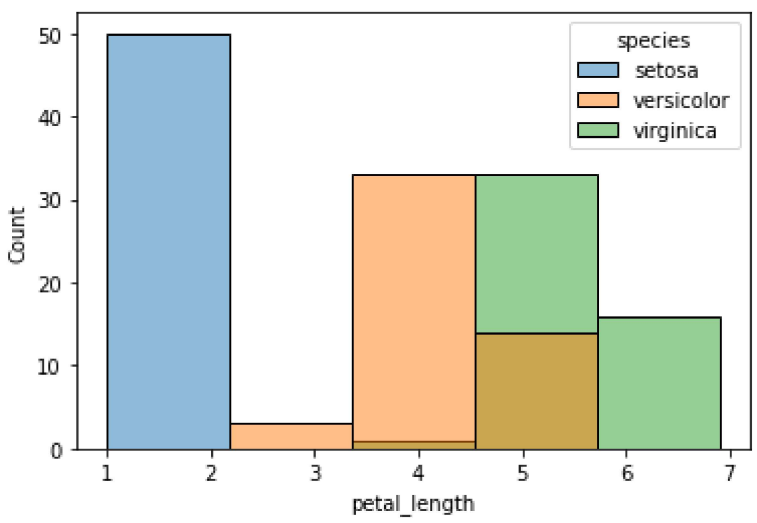
In [10]:
```
sns.histplot(df,x='sepal_width',hue='species',bins=5)
```

Out[10]: <AxesSubplot:xlabel='sepal_width', ylabel='Count'>



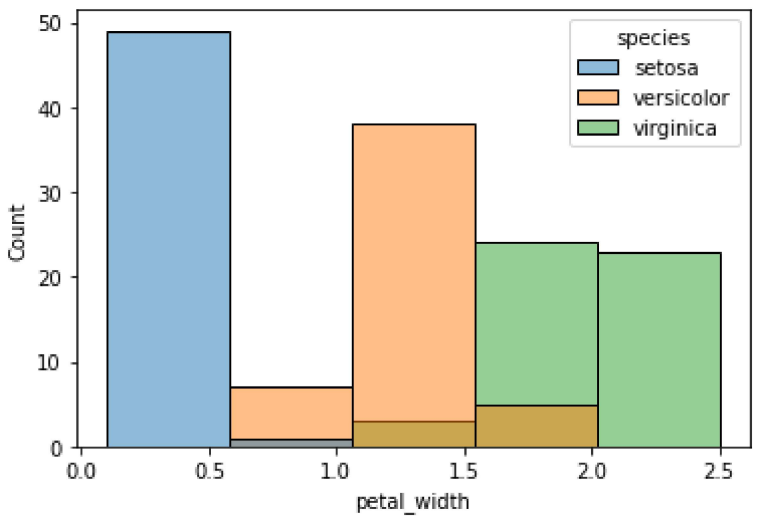
In [11]:
```
sns.histplot(df,x='petal_length',hue='species',bins=5)
```

Out[11]: <AxesSubplot:xlabel='petal_length', ylabel='Count'>



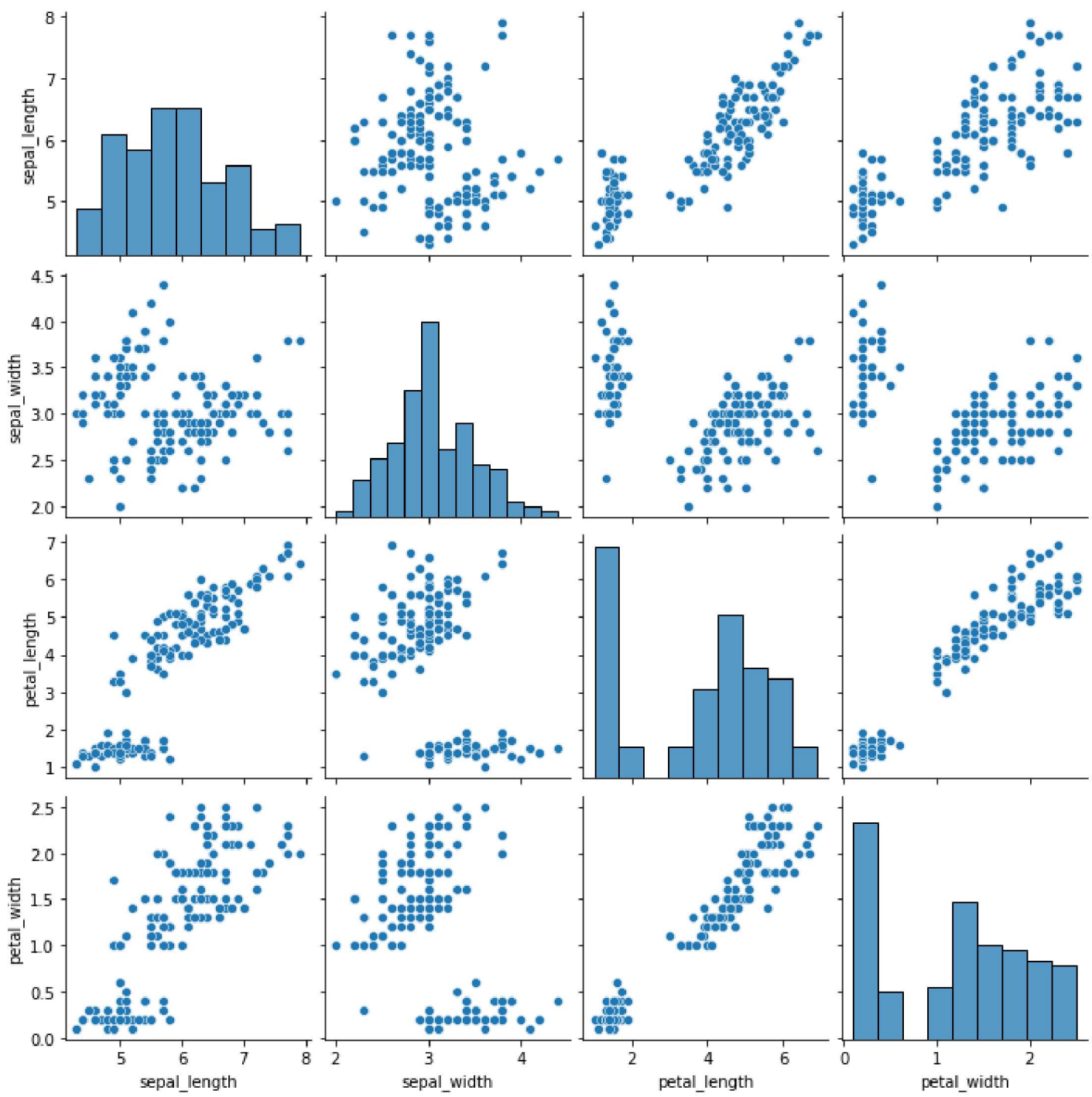
In [12]:
```
sns.histplot(df,x='petal_width',hue='species',bins=5)
```

Out[12]: <AxesSubplot:xlabel='petal_width', ylabel='Count'>



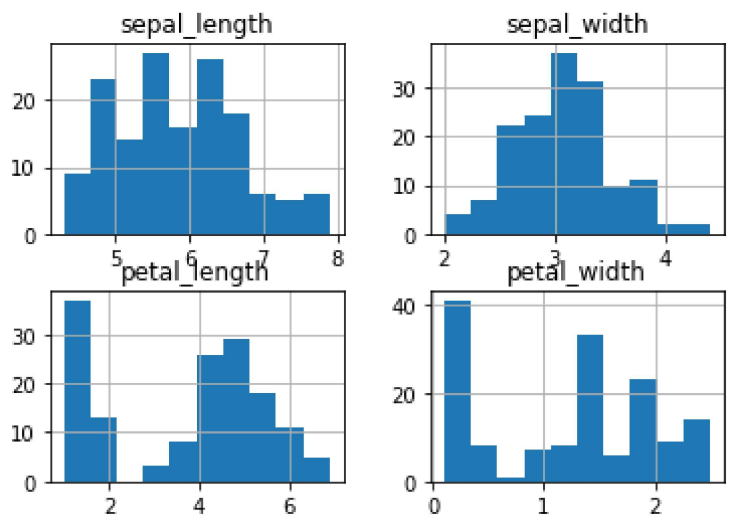
In [13]:
```
sns.pairplot(df)
```

Out[13]: <seaborn.axisgrid.PairGrid at 0x7fd9600f8fd0>



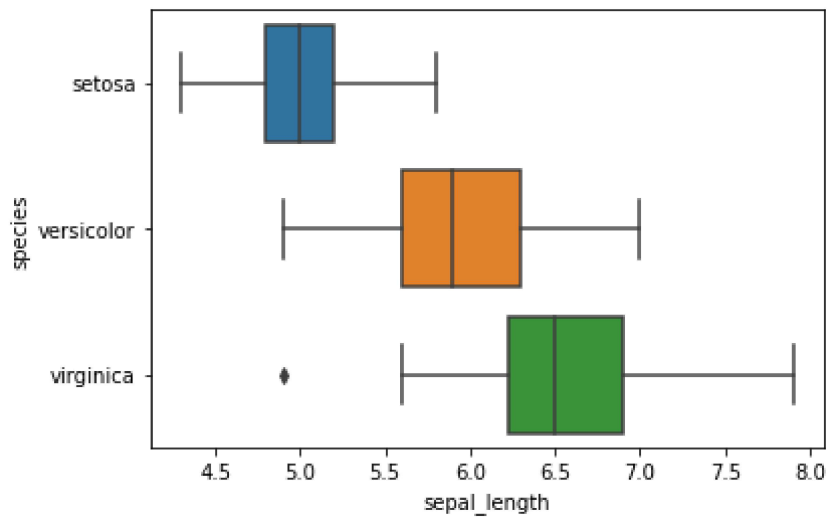
In [14]:
```
df.hist()
```

Out[14]:
```
array([[<AxesSubplot:title={'center':'sepal_length'}>,
        <AxesSubplot:title={'center':'sepal_width'}>],
       [<AxesSubplot:title={'center':'petal_length'}>,
        <AxesSubplot:title={'center':'petal_width'}>]], dtype=object)
```



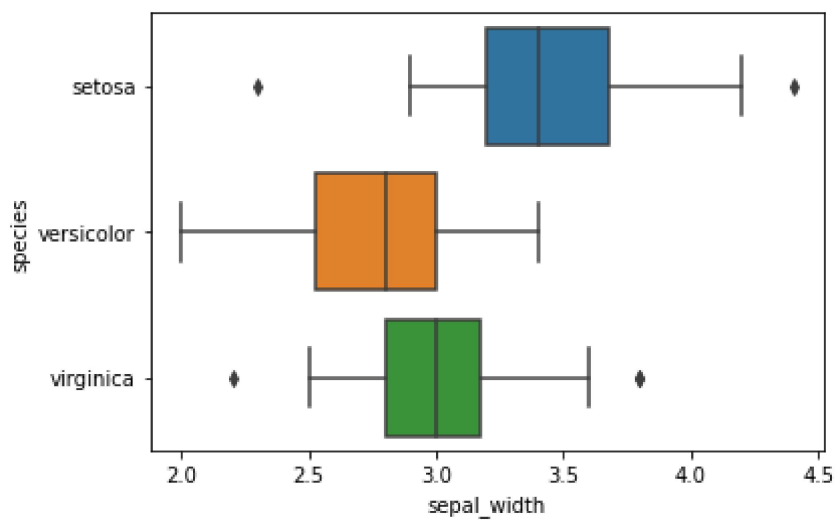
In [15]:
```
sns.boxplot(x='sepal_length',y='species',data=df)
```

Out[15]: <AxesSubplot:xlabel='sepal_length', ylabel='species'>



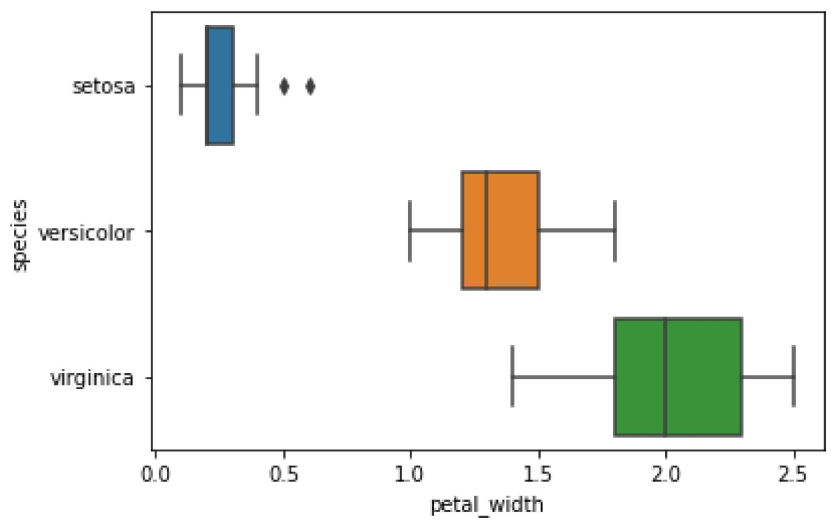
In [16]:
```
sns.boxplot(x='sepal_width',y='species',data=df)
```

Out[16]: <AxesSubplot:xlabel='sepal_width', ylabel='species'>



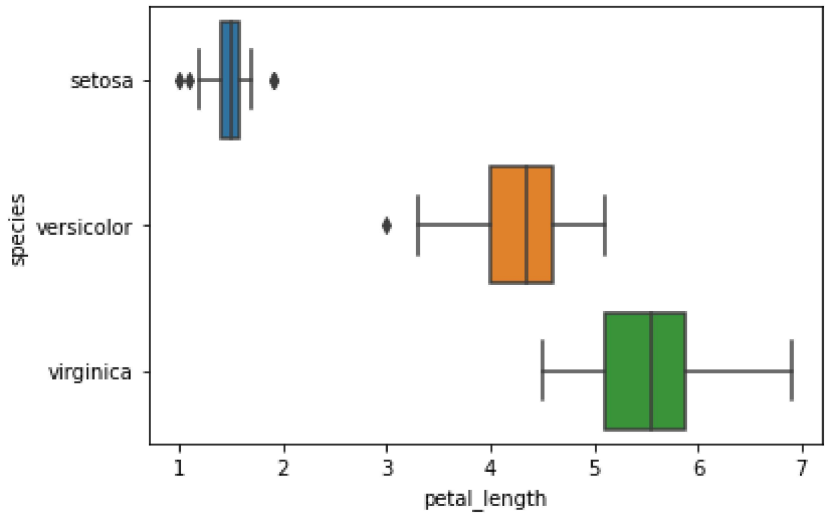
In [17]:
```
sns.boxplot(x='petal_width',y='species',data=df)
```

Out[17]: <AxesSubplot:xlabel='petal_width', ylabel='species'>



In [31]:
```
sns.boxplot(x='petal_length',y='species',data=df)
```

Out[31]: <AxesSubplot:xlabel='petal_length', ylabel='species'>



In [25]:
```
outliers = []
new=df['sepal_width']

def detect_outliers_iqr(data):
    data = sorted(data)
    q1 = np.percentile(data, 25)
    q3 = np.percentile(data, 75)

    IQR = q3-q1
    lwr_bound = q1-(1.5*IQR)
    upr_bound = q3+(1.5*IQR)

    for i in data:
        if (i<lwr_bound or i>upr_bound):
            outliers.append(i)
    return outliers
new_outliers = detect_outliers_iqr(new)
print("Outliers from IQR method: ", new_outliers)
```

```
Outliers from IQR method:  [2.0, 4.1, 4.2, 4.4]
```

In [ ]: