In [31]:
```
import pandas as pd
```

In [33]:
```
df = pd.read_csv("C:\\Users\\Atwongire Vianney\\Desktop\\AI_PRAC\\Iris_Data.csv
df.head()
```

Out[33]:

	sepal_length	sepal_width	petal_length	petal_width
0	5.1	3.5	1.4	0.2
1	4.9	3.0	1.4	0.2
2	4.7	3.2	1.3	0.2
3	4.6	3.1	1.5	0.2
4	5.0	3.6	1.4	0.2

In [34]:
```
df.head(20)
```

Out[34]:

	sepal_length	sepal_width	petal_length	petal_width
0	5.1	3.5	1.4	0.2
1	4.9	3.0	1.4	0.2
2	4.7	3.2	1.3	0.2
3	4.6	3.1	1.5	0.2
4	5.0	3.6	1.4	0.2
5	5.4	3.9	1.7	0.4
6	4.6	3.4	1.4	0.3
7	5.0	3.4	1.5	0.2
8	4.4	2.9	1.4	0.2
9	4.9	3.1	1.5	0.1
10	5.4	3.7	1.5	0.2
11	4.8	3.4	1.6	0.2
12	4.8	3.0	1.4	0.1
13	4.3	3.0	1.1	0.1
14	5.8	4.0	1.2	0.2
15	5.7	4.4	1.5	0.4
16	5.4	3.9	1.3	0.4
17	5.1	3.5	1.4	0.3
18	5.7	3.8	1.7	0.3
19	5.1	3.8	1.5	0.3

In [35]:
```
df.tail()
```

Out[35]:

	sepal_length	sepal_width	petal_length	petal_width
145	6.7	3.0	5.2	2.3
146	6.3	2.5	5.0	1.9
147	6.5	3.0	5.2	2.0
148	6.2	3.4	5.4	2.3
149	5.9	3.0	5.1	1.8

In [37]:
```
df.tail(10)
```

Out[37]:

	sepal_length	sepal_width	petal_length	petal_width
140	6.7	3.1	5.6	2.4
141	6.9	3.1	5.1	2.3
142	5.8	2.7	5.1	1.9
143	6.8	3.2	5.9	2.3
144	6.7	3.3	5.7	2.5
145	6.7	3.0	5.2	2.3
146	6.3	2.5	5.0	1.9
147	6.5	3.0	5.2	2.0
148	6.2	3.4	5.4	2.3
149	5.9	3.0	5.1	1.8

In [38]:
```
df
```

Out[38]:

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

150 rows × 4 columns

In [39]:
```
#appropriate 200 data sample
sepal_length_df = df['sepal_length']
sepal_length_df
```

Out[39]:
```
0      5.1
1      4.9
2      4.7
3      4.6
4      5.0
      ...
145    6.7
146    6.3
147    6.5
148    6.2
149    5.9
Name: sepal_length, Length: 150, dtype: float64
```

In [40]:
```
petal_length_df = df['petal_length']
petal_length_df
```

Out[40]:
```
0      1.4
1      1.4
2      1.3
3      1.5
4      1.4
      ...
145    5.2
146    5.0
147    5.2
148    5.4
149    5.1
Name: petal_length, Length: 150, dtype: float64
```

In [41]:
```
petal_length_list = petal_length_df.tolist()
sepal_length_list = sepal_length_df.tolist()
```

In [42]:
```
petal_length_list
```

Out[42]:
```
[1.4,
 1.4,
 1.3,
 1.5,
 1.4,
 1.7,
 1.4,
 1.5,
 1.4,
 1.5,
 1.5,
 1.6,
 1.4,
 1.1,
 1.2,
 1.5,
 1.3,
 1.4,
 1.7,
```

In [43]:
```
sepal_length_list
```

Out[43]:
```
[5.1,
 4.9,
 4.7,
 4.6,
 5.0,
 5.4,
 4.6,
 5.0,
 4.4,
 4.9,
 5.4,
 4.8,
 4.8,
 4.3,
 5.8,
 5.7,
 5.4,
 5.1,
 5.7,
```

In [50]:
```
#plotting a graph of petal_length_list against sepal_length_list
def line_of_the_best_fit(xs,ys):
    slope=((mean(xs)*mean(ys))-(mean(xs*ys))/(mean(xs)*mean(xs)-mean(xs*xs)))
    y_intercept=mean(ys)-slope*mean(xs)
    return slope, y_intercept
```

In [51]:
```
#introducing numpy library
import numpy as np
xs=np.array(petal_length_list, dtype = np.float64)
ys=np.array(sepal_length_list, dtype = np.float64)
```

In [53]:
```
xs
```

Out[53]:
```
array([1.4, 1.4, 1.3, 1.5, 1.4, 1.7, 1.4, 1.5, 1.4, 1.5, 1.5, 1.6, 1.4,
       1.1, 1.2, 1.5, 1.3, 1.4, 1.7, 1.5, 1.7, 1.5, 1. , 1.7, 1.9, 1.6,
       1.6, 1.5, 1.4, 1.6, 1.6, 1.5, 1.5, 1.4, 1.5, 1.2, 1.3, 1.5, 1.3,
       1.5, 1.3, 1.3, 1.3, 1.6, 1.9, 1.4, 1.6, 1.4, 1.5, 1.4, 4.7, 4.5,
       4.9, 4. , 4.6, 4.5, 4.7, 3.3, 4.6, 3.9, 3.5, 4.2, 4. , 4.7, 3.6,
       4.4, 4.5, 4.1, 4.5, 3.9, 4.8, 4. , 4.9, 4.7, 4.3, 4.4, 4.8, 5. ,
       4.5, 3.5, 3.8, 3.7, 3.9, 5.1, 4.5, 4.5, 4.7, 4.4, 4.1, 4. , 4.4,
       4.6, 4. , 3.3, 4.2, 4.2, 4.2, 4.3, 3. , 4.1, 6. , 5.1, 5.9, 5.6,
       5.8, 6.6, 4.5, 6.3, 5.8, 6.1, 5.1, 5.3, 5.5, 5. , 5.1, 5.3, 5.5,
       6.7, 6.9, 5. , 5.7, 4.9, 6.7, 4.9, 5.7, 6. , 4.8, 4.9, 5.6, 5.8,
       6.1, 6.4, 5.6, 5.1, 5.6, 6.1, 5.6, 5.5, 4.8, 5.4, 5.6, 5.1, 5.1,
       5.9, 5.7, 5.2, 5. , 5.2, 5.4, 5.1])
```

In [46]:
```
ys
```

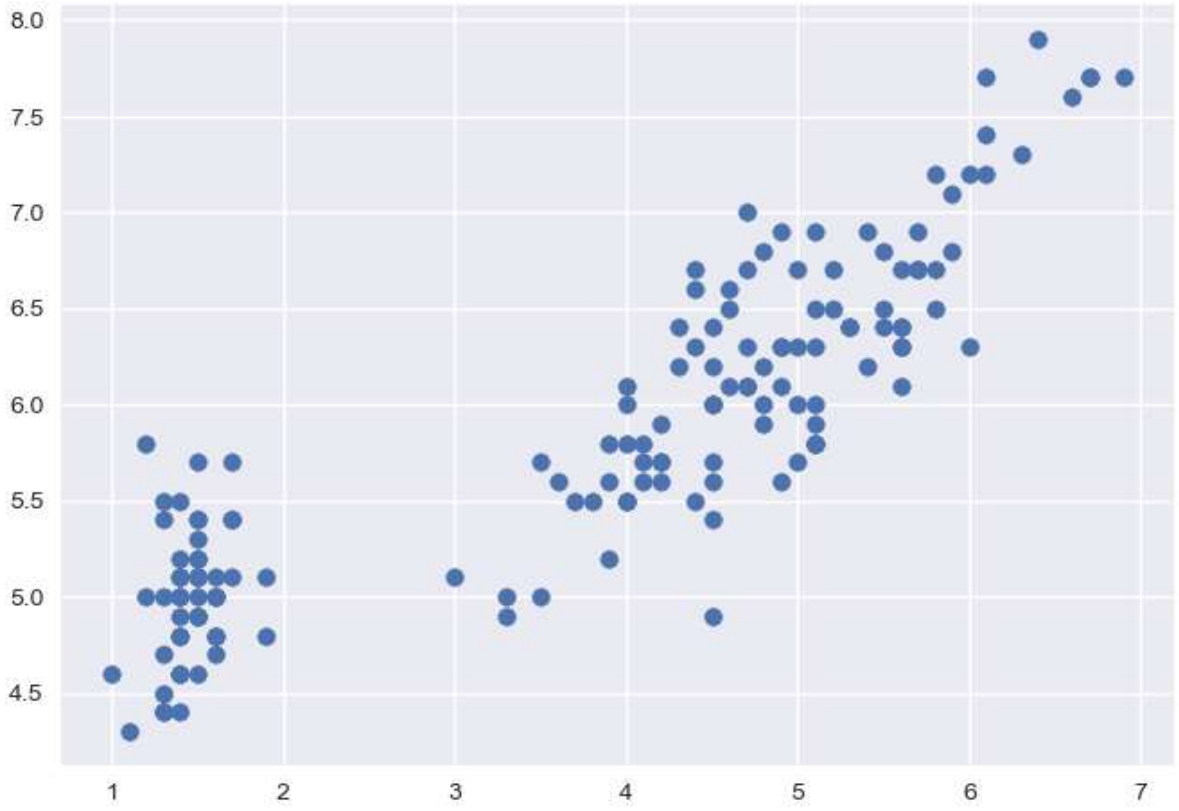
Out[46]:
```
array([5.1, 4.9, 4.7, 4.6, 5. , 5.4, 4.6, 5. , 4.4, 4.9, 5.4, 4.8, 4.8,
       4.3, 5.8, 5.7, 5.4, 5.1, 5.7, 5.1, 5.4, 5.1, 4.6, 5.1, 4.8, 5. ,
       5. , 5.2, 5.2, 4.7, 4.8, 5.4, 5.2, 5.5, 4.9, 5. , 5.5, 4.9, 4.4,
       5.1, 5. , 4.5, 4.4, 5. , 5.1, 4.8, 5.1, 4.6, 5.3, 5. , 7. , 6.4,
       6.9, 5.5, 6.5, 5.7, 6.3, 4.9, 6.6, 5.2, 5. , 5.9, 6. , 6.1, 5.6,
       6.7, 5.6, 5.8, 6.2, 5.6, 5.9, 6.1, 6.3, 6.1, 6.4, 6.6, 6.8, 6.7,
       6. , 5.7, 5.5, 5.5, 5.8, 6. , 5.4, 6. , 6.7, 6.3, 5.6, 5.5, 5.5,
       6.1, 5.8, 5. , 5.6, 5.7, 5.7, 6.2, 5.1, 5.7, 6.3, 5.8, 7.1, 6.3,
       6.5, 7.6, 4.9, 7.3, 6.7, 7.2, 6.5, 6.4, 6.8, 5.7, 5.8, 6.4, 6.5,
       7.7, 7.7, 6. , 6.9, 5.6, 7.7, 6.3, 6.7, 7.2, 6.2, 6.1, 6.4, 7.2,
       7.4, 7.9, 6.4, 6.3, 6.1, 7.7, 6.3, 6.4, 6. , 6.9, 6.7, 6.9, 5.8,
       6.8, 6.7, 6.7, 6.3, 6.5, 6.2, 5.9])
```

In [52]:
```
#plottinga graph of xs and ys
import matplotlib.pyplot as plt
from statistics import mean
from matplotlib import style

style.use('seaborn')
plt.scatter(xs,ys, label = 'petal_length and sepal_length')
```

```
C:\Users\Atwongire Vianney\AppData\Local\Temp\ipykernel_4692\2344462897.py:6:
MatplotlibDeprecationWarning: The seaborn styles shipped by Matplotlib are de
precated since 3.6, as they no longer correspond to the styles shipped by sea
born. However, they will remain available as 'seaborn-v0_8-<style>'. Alternat
ively, directly use the seaborn API instead.
  style.use('seaborn')
```

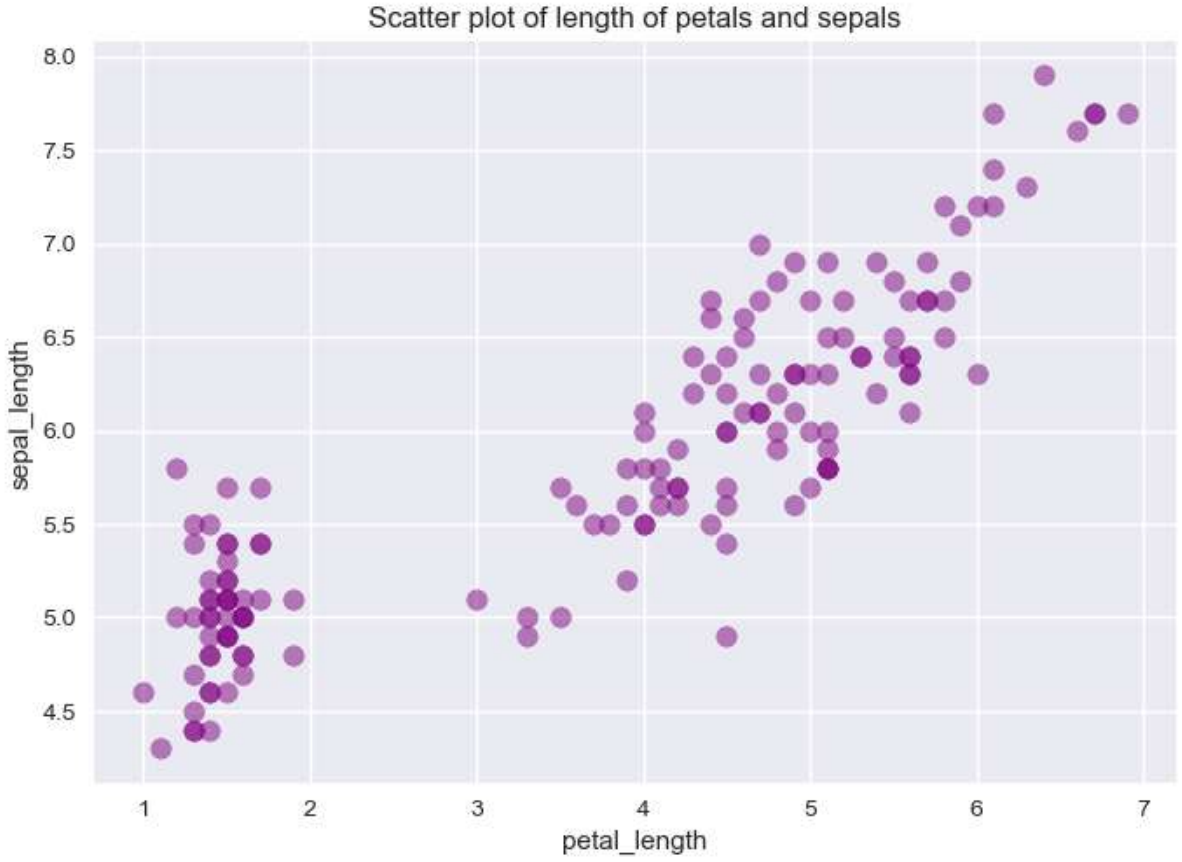
Out[52]:
```
<matplotlib.collections.PathCollection at 0x27aaafd17d0>
```



In [53]:
```
#to visualise data
style.use('seaborn')
plt.scatter(xs,ys, label = 'petal_length and sepal_length' ,alpha=0.5,color = '
plt.xlabel('petal_length')
plt.ylabel('sepal_length')
plt.title('Scatter plot of length of petals and sepals')
```

```
C:\Users\Atwongire Vianney\AppData\Local\Temp\ipykernel_4692\1546589265.py:2:
MatplotlibDeprecationWarning: The seaborn styles shipped by Matplotlib are de
precated since 3.6, as they no longer correspond to the styles shipped by sea
born. However, they will remain available as 'seaborn-v0_8-<style>'. Alternat
ively, directly use the seaborn API instead.
  style.use('seaborn')
```

Out[53]:
```
Text(0.5, 1.0, 'Scatter plot of length of petals and sepals')
```



In [67]:
```
#building a linear regression
slope,y_intercept = line_of_the_best_fit(xs,ys)
regression_line = [slope*x + y_intercept for x in xs]
#predictions
petal_length = 1.1
sepal_length = slope*petal_length + y_intercept
```

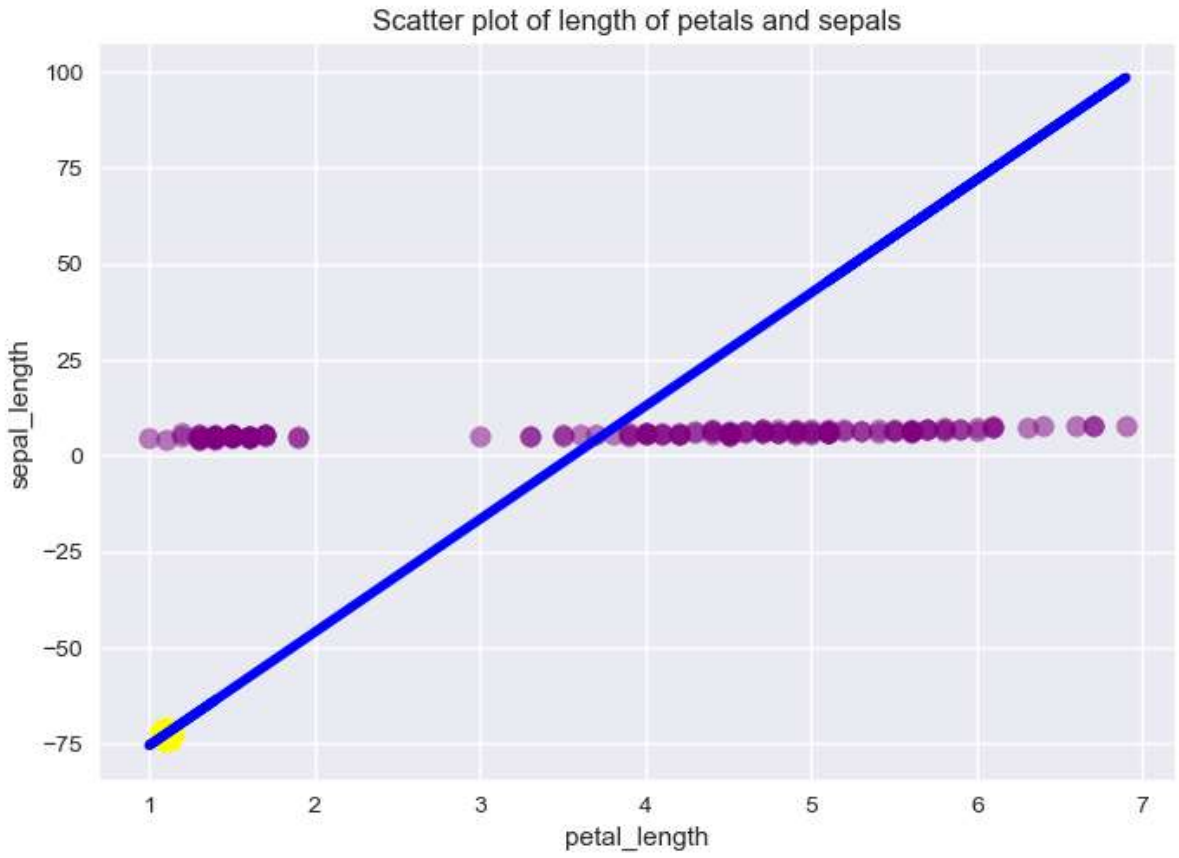
In [66]:
```
petal_length
```

Out[66]:
```
45.37847289964432
```

In [68]:
```
style.use('seaborn')
plt.scatter(xs,ys, label = 'petal_length and sepal_length' ,alpha=0.5,color = '
plt.scatter(petal_length,sepal_length,label='prediction of sepal_length', color
plt.plot(xs,regression_line,label='line of the best fit',color='blue',linewidth
plt.xlabel('petal_length')
plt.ylabel('sepal_length')
plt.title('Scatter plot of length of petals and sepals')
```

```
C:\Users\Atwongire Vianney\AppData\Local\Temp\ipykernel_4692\1222711858.py:1:
MatplotlibDeprecationWarning: The seaborn styles shipped by Matplotlib are de
precated since 3.6, as they no longer correspond to the styles shipped by sea
born. However, they will remain available as 'seaborn-v0_8-<style>'. Alternat
ively, directly use the seaborn API instead.
  style.use('seaborn')
```

Out[68]: Text(0.5, 1.0, 'Scatter plot of length of petals and sepals')



In [ ]: