

In [1]:

```
import pandas as pd
from matplotlib import pyplot as plt
%matplotlib inline
```

In [2]:

```
data=pd.read_csv(r"C:\Users\Prathyusha\Downloads\BreastCancerPrediction.csv")
data
```

Out[2]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothn
0	842302	M	17.99	10.38	122.80	1001.0	
1	842517	M	20.57	17.77	132.90	1326.0	
2	84300903	M	19.69	21.25	130.00	1203.0	
3	84348301	M	11.42	20.38	77.58	386.1	
4	84358402	M	20.29	14.34	135.10	1297.0	
...	...	...	...	...	...	...	
564	926424	M	21.56	22.39	142.00	1479.0	
565	926682	M	20.13	28.25	131.20	1261.0	
566	926954	M	16.60	28.08	108.30	858.1	
567	927241	M	20.60	29.33	140.10	1265.0	
568	92751	B	7.76	24.54	47.92	181.0	

569 rows × 33 columns



In [3]:

```
data.head()
```

Out[3]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothnes
0	842302	M	17.99	10.38	122.80	1001.0	
1	842517	M	20.57	17.77	132.90	1326.0	
2	84300903	M	19.69	21.25	130.00	1203.0	
3	84348301	M	11.42	20.38	77.58	386.1	
4	84358402	M	20.29	14.34	135.10	1297.0	

5 rows × 33 columns



In [4]:

```
data.tail()
```

Out[4]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothnes
564	926424	M	21.56	22.39	142.00	1479.0	
565	926682	M	20.13	28.25	131.20	1261.0	
566	926954	M	16.60	28.08	108.30	858.1	
567	927241	M	20.60	29.33	140.10	1265.0	
568	92751	B	7.76	24.54	47.92	181.0	

5 rows × 33 columns



In [5]:

```
data.describe()
```

Out[5]:

	id	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_
count	5.690000e+02	569.000000	569.000000	569.000000	569.000000	569.0
mean	3.037183e+07	14.127292	19.289649	91.969033	654.889104	0.0
std	1.250206e+08	3.524049	4.301036	24.298981	351.914129	0.0
min	8.670000e+03	6.981000	9.710000	43.790000	143.500000	0.0
25%	8.692180e+05	11.700000	16.170000	75.170000	420.300000	0.0
50%	9.060240e+05	13.370000	18.840000	86.240000	551.100000	0.0
75%	8.813129e+06	15.780000	21.800000	104.100000	782.700000	0.1
max	9.113205e+08	28.110000	39.280000	188.500000	2501.000000	0.1

8 rows × 32 columns



In [6]:

```
data.shape
```

Out[6]:

(569, 33)

In [7]:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 569 entries, 0 to 568
```

```
Data columns (total 33 columns):
```

#	Column	Non-Null Count	Dtype
0	id	569 non-null	int64
1	diagnosis	569 non-null	object
2	radius_mean	569 non-null	float64
3	texture_mean	569 non-null	float64
4	perimeter_mean	569 non-null	float64
5	area_mean	569 non-null	float64
6	smoothness_mean	569 non-null	float64
7	compactness_mean	569 non-null	float64
8	concavity_mean	569 non-null	float64
9	concave points_mean	569 non-null	float64
10	symmetry_mean	569 non-null	float64
11	fractal_dimension_mean	569 non-null	float64
12	radius_se	569 non-null	float64
13	texture_se	569 non-null	float64
14	perimeter_se	569 non-null	float64
15	area_se	569 non-null	float64
16	smoothness_se	569 non-null	float64
17	compactness_se	569 non-null	float64
18	concavity_se	569 non-null	float64
19	concave points_se	569 non-null	float64
20	symmetry_se	569 non-null	float64
21	fractal_dimension_se	569 non-null	float64
22	radius_worst	569 non-null	float64
23	texture_worst	569 non-null	float64
24	perimeter_worst	569 non-null	float64
25	area_worst	569 non-null	float64
26	smoothness_worst	569 non-null	float64
27	compactness_worst	569 non-null	float64
28	concavity_worst	569 non-null	float64
29	concave points_worst	569 non-null	float64
30	symmetry_worst	569 non-null	float64
31	fractal_dimension_worst	569 non-null	float64
32	Unnamed: 32	0 non-null	float64

```
dtypes: float64(31), int64(1), object(1)
```

```
memory usage: 146.8+ KB
```

In [8]:

```
data.isnull().sum()
```

Out[8]:

id	0
diagnosis	0
radius_mean	0
texture_mean	0
perimeter_mean	0
area_mean	0
smoothness_mean	0
compactness_mean	0
concavity_mean	0
concave points_mean	0
symmetry_mean	0
fractal_dimension_mean	0
radius_se	0
texture_se	0
perimeter_se	0
area_se	0
smoothness_se	0
compactness_se	0
concavity_se	0
concave points_se	0
symmetry_se	0
fractal_dimension_se	0
radius_worst	0
texture_worst	0
perimeter_worst	0
area_worst	0
smoothness_worst	0
compactness_worst	0
concavity_worst	0
concave points_worst	0
symmetry_worst	0
fractal_dimension_worst	0
Unnamed: 32	569

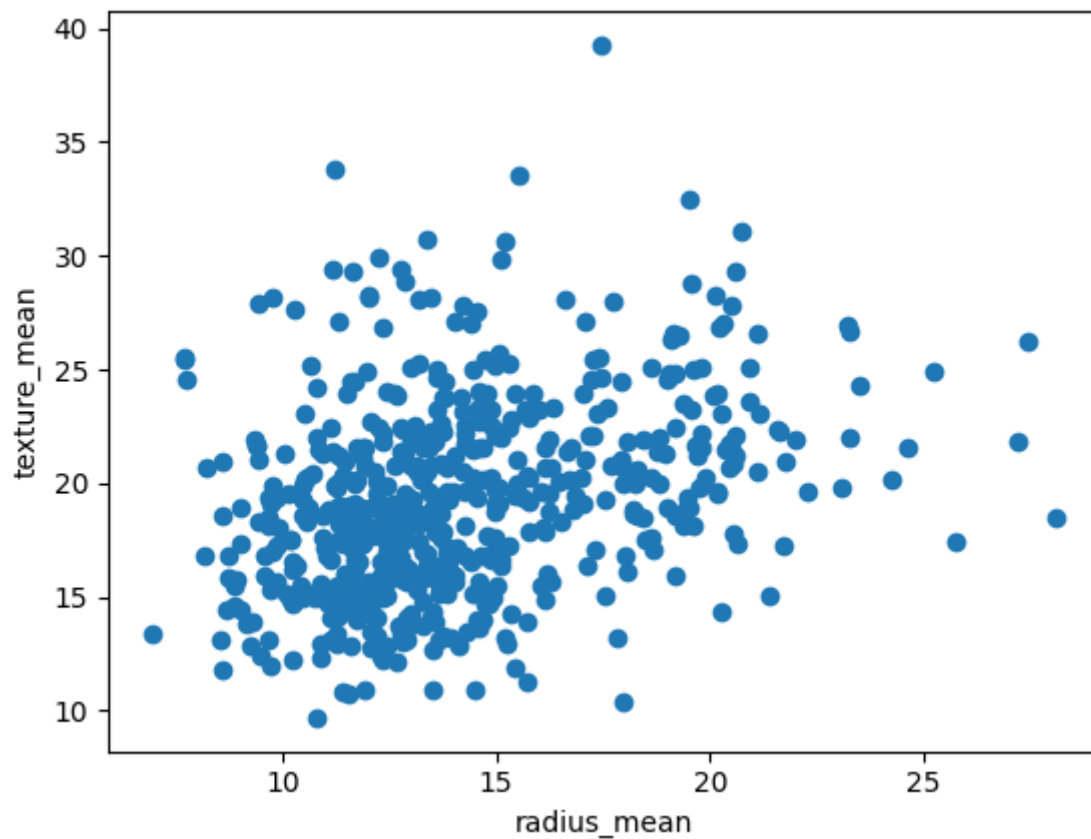
dtype: int64

In [9]:

```
plt.scatter(data["radius_mean"],data["texture_mean"])  
plt.xlabel("radius_mean")  
plt.ylabel("texture_mean")
```

Out[9]:

Text(0, 0.5, 'texture\_mean')



In [10]:

```
from sklearn.cluster import KMeans  
km=KMeans()  
km
```

Out[10]:

```
▼ KMeans  
KMeans()
```

In [11]:

```
y_predicted=km.fit_predict(data[["radius_mean","texture_mean"]])
y_predicted
```

C:\Users\Prathyusha\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\\_kmeans.py:870: FutureWarning: The default value of `n\_init` will change from 10 to 'auto' in 1.4. Set the value of `n\_init` explicitly to suppress the warning  
warnings.warn(

Out[11]:

```
array([5, 3, 3, 1, 3, 5, 3, 7, 2, 2, 7, 7, 0, 2, 2, 6, 7, 7, 3, 5, 5, 4,
       5, 0, 7, 5, 7, 3, 2, 5, 0, 1, 0, 0, 7, 7, 7, 1, 2, 7, 2, 2, 0, 7,
       2, 3, 1, 1, 4, 2, 2, 5, 1, 3, 7, 1, 3, 7, 1, 4, 4, 1, 2, 4, 2, 2,
       1, 1, 1, 5, 3, 4, 0, 5, 1, 7, 4, 5, 0, 1, 2, 5, 0, 0, 4, 3, 7, 0,
       2, 5, 2, 7, 5, 1, 7, 0, 1, 1, 4, 7, 2, 4, 1, 1, 1, 5, 1, 1, 3, 2,
       1, 2, 7, 1, 4, 2, 4, 5, 7, 3, 4, 3, 3, 5, 5, 5, 2, 3, 5, 0, 4, 7,
       7, 5, 3, 2, 1, 4, 5, 4, 4, 7, 1, 5, 4, 4, 1, 7, 5, 1, 2, 1, 4, 4,
       5, 1, 7, 7, 4, 4, 1, 3, 3, 2, 3, 7, 4, 7, 0, 5, 4, 7, 5, 4, 4, 4,
       1, 7, 2, 4, 3, 0, 7, 4, 7, 4, 3, 1, 1, 5, 2, 2, 1, 6, 2, 5, 2, 3,
       3, 7, 1, 7, 0, 2, 1, 5, 1, 7, 2, 5, 3, 1, 3, 0, 2, 5, 1, 1, 3, 0,
       5, 5, 1, 7, 5, 5, 4, 5, 2, 2, 7, 6, 6, 0, 4, 7, 0, 3, 6, 6, 5, 4,
       1, 2, 0, 1, 1, 4, 2, 4, 0, 1, 3, 5, 3, 5, 0, 5, 7, 6, 0, 7, 7, 7,
       7, 0, 1, 2, 5, 1, 5, 4, 3, 4, 0, 1, 4, 3, 1, 5, 0, 4, 3, 7, 5, 1,
       2, 4, 1, 1, 7, 7, 5, 1, 4, 5, 4, 1, 7, 2, 3, 1, 0, 1, 1, 2, 5, 4,
       4, 5, 1, 5, 4, 4, 1, 1, 4, 3, 1, 1, 4, 3, 4, 3, 4, 1, 5, 1, 7, 7,
       5, 1, 1, 4, 1, 7, 5, 3, 1, 0, 5, 1, 4, 3, 4, 4, 1, 5, 4, 4, 1, 7,
       3, 2, 4, 1, 1, 5, 4, 1, 1, 2, 1, 7, 5, 3, 0, 1, 3, 3, 7, 5, 3, 3,
       5, 5, 1, 6, 5, 1, 4, 4, 2, 1, 5, 2, 4, 5, 4, 0, 4, 1, 7, 3, 1, 5,
       1, 1, 4, 1, 3, 4, 1, 5, 4, 1, 5, 2, 3, 1, 1, 1, 2, 7, 6, 2, 2, 7,
       4, 2, 1, 5, 4, 7, 1, 2, 4, 2, 1, 1, 7, 1, 3, 3, 5, 7, 1, 5, 7, 5,
       1, 0, 5, 1, 3, 2, 0, 5, 7, 3, 2, 0, 6, 5, 1, 6, 6, 2, 2, 6, 0, 0,
       6, 1, 1, 7, 7, 1, 0, 1, 1, 6, 5, 6, 4, 5, 7, 5, 4, 7, 1, 7, 5, 5,
       5, 5, 5, 3, 1, 7, 2, 5, 3, 4, 7, 7, 1, 1, 3, 3, 5, 2, 5, 3, 4, 4,
       1, 1, 5, 2, 4, 5, 7, 5, 7, 1, 3, 3, 1, 5, 4, 3, 1, 1, 4, 4, 1, 4,
       5, 4, 1, 1, 5, 3, 1, 3, 2, 2, 2, 2, 4, 2, 2, 6, 7, 2, 1, 1, 1, 2,
       2, 2, 6, 2, 6, 6, 1, 6, 2, 2, 6, 6, 6, 0, 3, 0, 6, 0, 2])
```

In [12]:

```
data["cluster"]=y_predicted
data.head()
```

Out[12]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothnes
0	842302	M	17.99	10.38	122.80	1001.0	
1	842517	M	20.57	17.77	132.90	1326.0	
2	84300903	M	19.69	21.25	130.00	1203.0	
3	84348301	M	11.42	20.38	77.58	386.1	
4	84358402	M	20.29	14.34	135.10	1297.0	

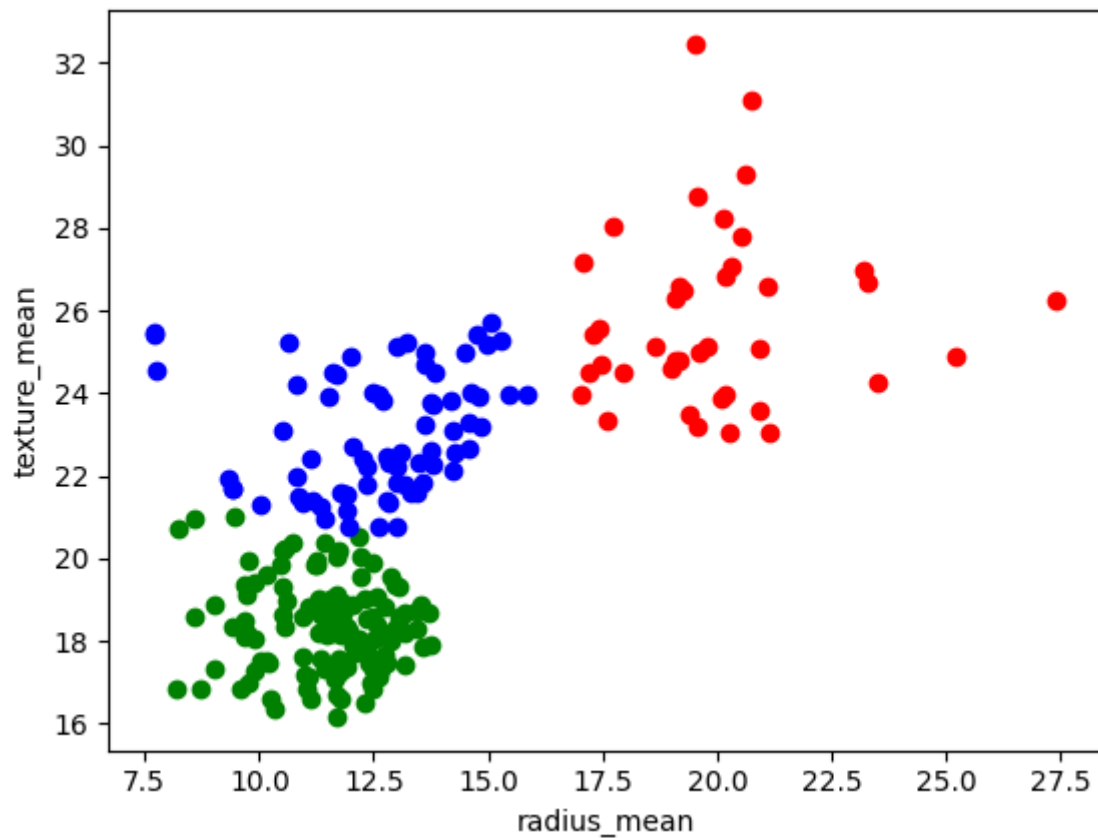
5 rows × 34 columns

In [13]:

```
data1=data[data.cluster==0]
data2=data[data.cluster==1]
data3=data[data.cluster==2]
plt.scatter(data1["radius_mean"],data1["texture_mean"],color="red")
plt.scatter(data2["radius_mean"],data2["texture_mean"],color="green")
plt.scatter(data3["radius_mean"],data3["texture_mean"],color="blue")
plt.xlabel("radius_mean")
plt.ylabel("texture_mean")
```

Out[13]:

Text(0, 0.5, 'texture\_mean')



In [14]:

```
from sklearn.preprocessing import MinMaxScaler
scaler=MinMaxScaler()
scaler.fit(data[["texture_mean"]])
data["texture_mean"]=scaler.transform(data[["texture_mean"]])
data.head()
```

Out[14]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothnes
0	842302	M	17.99	0.022658	122.80	1001.0	
1	842517	M	20.57	0.272574	132.90	1326.0	
2	84300903	M	19.69	0.390260	130.00	1203.0	
3	84348301	M	11.42	0.360839	77.58	386.1	
4	84358402	M	20.29	0.156578	135.10	1297.0	

5 rows × 34 columns



In [15]:

```
scaler.fit(data[["radius_mean"]])
data["radius_mean"]=scaler.transform(data[["radius_mean"]])
data.head()
```

Out[15]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothnes
0	842302	M	0.521037	0.022658	122.80	1001.0	
1	842517	M	0.643144	0.272574	132.90	1326.0	
2	84300903	M	0.601496	0.390260	130.00	1203.0	
3	84348301	M	0.210090	0.360839	77.58	386.1	
4	84358402	M	0.629893	0.156578	135.10	1297.0	

5 rows × 34 columns





In [16]:

```
y_predicted=km.fit_predict(data[["radius_mean","texture_mean"]])
y_predicted
```

C:\Users\Prathyusha\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\\_kmeans.py:870: FutureWarning: The default value of `n\_init` will change from 10 to 'auto' in 1.4. Set the value of `n\_init` explicitly to suppress the warning  
warnings.warn(

Out[16]:

```
array([4, 1, 1, 0, 1, 4, 1, 2, 2, 2, 2, 4, 6, 2, 2, 5, 2, 2, 1, 4, 4, 3,
       4, 6, 2, 1, 2, 1, 2, 1, 6, 0, 6, 6, 4, 2, 2, 0, 2, 2, 2, 0, 6, 2,
       2, 1, 3, 0, 3, 2, 0, 4, 0, 1, 2, 0, 1, 2, 0, 3, 3, 0, 2, 3, 2, 2,
       0, 0, 0, 4, 1, 3, 6, 4, 0, 2, 4, 1, 6, 0, 0, 4, 7, 6, 3, 1, 2, 6,
       2, 4, 2, 2, 4, 0, 2, 6, 0, 0, 3, 2, 2, 3, 0, 0, 0, 4, 0, 0, 7, 0,
       0, 0, 2, 0, 3, 0, 3, 4, 2, 1, 3, 1, 7, 4, 4, 4, 2, 1, 4, 6, 3, 2,
       2, 4, 1, 2, 0, 3, 4, 3, 3, 4, 0, 4, 3, 3, 0, 2, 4, 4, 2, 0, 3, 3,
       4, 0, 1, 1, 3, 3, 0, 1, 1, 2, 7, 2, 3, 1, 6, 4, 3, 2, 4, 3, 3, 3,
       0, 1, 2, 4, 7, 6, 2, 3, 2, 3, 1, 0, 0, 4, 2, 2, 0, 5, 2, 4, 2, 1,
       1, 2, 0, 1, 7, 2, 0, 4, 0, 1, 2, 4, 1, 0, 7, 6, 2, 4, 0, 0, 1, 6,
       4, 4, 0, 2, 4, 4, 3, 4, 2, 2, 1, 5, 5, 6, 3, 2, 7, 1, 5, 5, 4, 4,
       0, 2, 6, 0, 4, 4, 5, 3, 6, 0, 1, 1, 1, 4, 6, 4, 2, 5, 6, 6, 1, 2,
       1, 6, 0, 2, 4, 0, 4, 3, 7, 3, 6, 0, 3, 1, 4, 4, 6, 3, 1, 1, 4, 0,
       0, 4, 0, 0, 2, 2, 4, 0, 4, 4, 3, 0, 4, 0, 1, 0, 6, 0, 0, 5, 4, 3,
       4, 4, 0, 4, 4, 3, 0, 0, 3, 1, 0, 0, 3, 1, 4, 1, 3, 0, 4, 0, 2, 2,
       4, 0, 0, 3, 0, 1, 4, 1, 0, 7, 4, 3, 3, 1, 3, 3, 0, 4, 3, 3, 0, 2,
       7, 2, 3, 0, 0, 4, 3, 0, 0, 2, 0, 1, 4, 1, 6, 0, 1, 7, 2, 4, 1, 1,
       4, 4, 0, 5, 4, 0, 3, 3, 2, 0, 4, 2, 3, 4, 3, 6, 3, 3, 2, 7, 0, 4,
       0, 0, 3, 0, 1, 3, 0, 4, 3, 0, 4, 2, 1, 0, 0, 0, 0, 2, 5, 0, 0, 2,
       3, 0, 0, 4, 3, 2, 0, 0, 3, 0, 0, 0, 2, 0, 1, 1, 4, 2, 0, 4, 2, 4,
       0, 6, 4, 0, 1, 5, 6, 4, 2, 1, 0, 6, 5, 4, 0, 5, 5, 5, 5, 5, 6, 7,
       5, 0, 0, 2, 2, 0, 6, 0, 0, 5, 4, 5, 3, 4, 2, 4, 3, 2, 0, 2, 4, 4,
       4, 4, 4, 1, 3, 1, 2, 4, 1, 3, 2, 2, 0, 0, 1, 1, 4, 2, 4, 7, 3, 3,
       0, 0, 4, 2, 3, 4, 2, 4, 2, 0, 1, 1, 0, 4, 3, 7, 0, 2, 3, 3, 0, 3,
       4, 3, 0, 0, 4, 1, 0, 1, 2, 5, 5, 5, 3, 2, 2, 5, 2, 2, 3, 3, 0, 5,
       0, 0, 5, 0, 5, 5, 0, 5, 2, 5, 5, 5, 5, 6, 7, 6, 6, 6, 5])
```

In [17]:

```
data["New Cluster"]=y_predicted
data.head()
```

Out[17]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothnes
0	842302	M	0.521037	0.022658	122.80	1001.0	
1	842517	M	0.643144	0.272574	132.90	1326.0	
2	84300903	M	0.601496	0.390260	130.00	1203.0	
3	84348301	M	0.210090	0.360839	77.58	386.1	
4	84358402	M	0.629893	0.156578	135.10	1297.0	

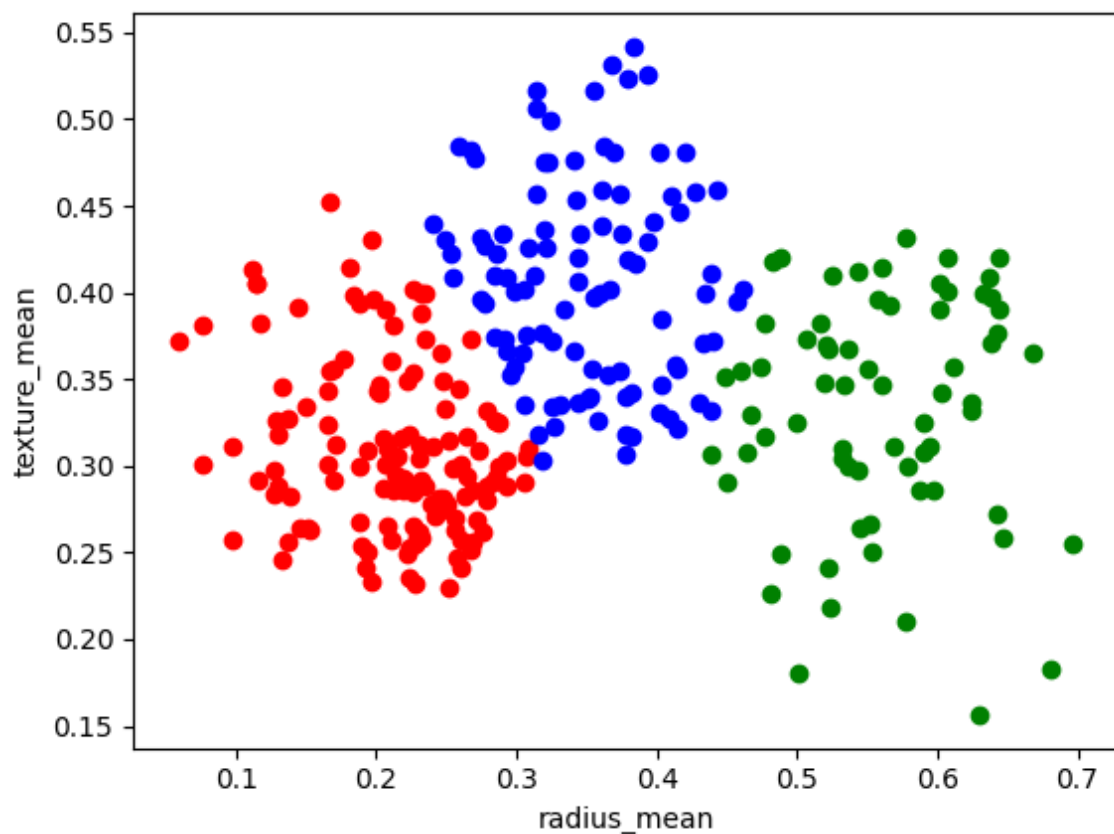
5 rows × 35 columns

In [19]:

```
data1=data[data["New Cluster"]==0]
data2=data[data["New Cluster"]==1]
data3=data[data["New Cluster"]==2]
plt.scatter(data1["radius_mean"],data1["texture_mean"],color="red")
plt.scatter(data2["radius_mean"],data2["texture_mean"],color="green")
plt.scatter(data3["radius_mean"],data3["texture_mean"],color="blue")
plt.xlabel("radius_mean")
plt.ylabel("texture_mean")
```

Out[19]:

Text(0, 0.5, 'texture\_mean')



In [20]:

```
km.cluster_centers_
```

Out[20]:

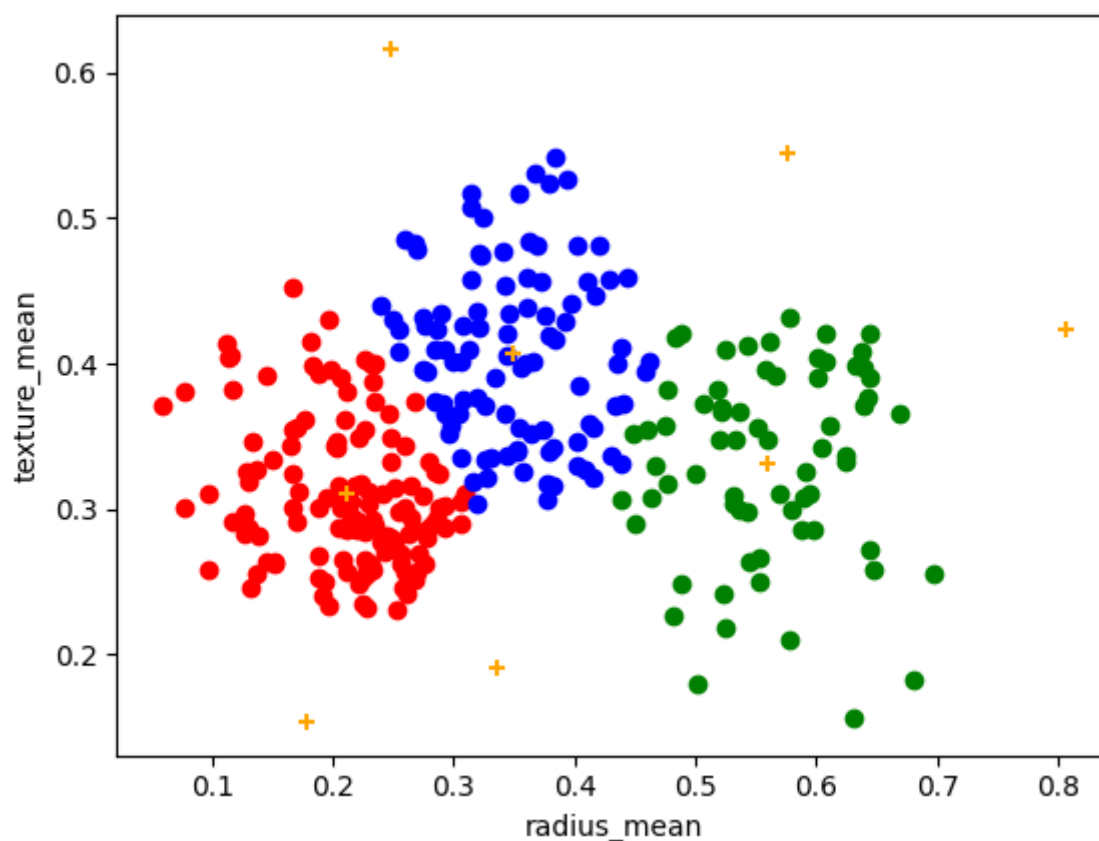
```
array([[0.21063269, 0.30993347],
       [0.55936641, 0.33176013],
       [0.34875763, 0.40662496],
       [0.17750575, 0.15412045],
       [0.33570532, 0.19063107],
       [0.24753115, 0.61622301],
       [0.57605341, 0.54408687],
       [0.80589822, 0.42316338]])
```

In [21]:

```
data1=data[data["New Cluster"]==0]
data2=data[data["New Cluster"]==1]
data3=data[data["New Cluster"]==2]
plt.scatter(data1["radius_mean"],data1["texture_mean"],color="red")
plt.scatter(data2["radius_mean"],data2["texture_mean"],color="green")
plt.scatter(data3["radius_mean"],data3["texture_mean"],color="blue")
plt.scatter(km.cluster_centers_[0],km.cluster_centers_[1],color="orange",marker="+")
plt.xlabel("radius_mean")
plt.ylabel("texture_mean")
```

Out[21]:

Text(0, 0.5, 'texture\_mean')



In [22]:

```
k_rng=range(1,10)
sse=[]
```

In [24]:

```
for k in k_rng:
    km=KMeans(n_clusters=k)
    km.fit(data[["radius_mean","texture_mean"]])
    sse.append(km.inertia_)
#km.inertia_ will give you the value of sum of square error
print(sse)
plt.plot(k_rng,sse)
plt.xlabel("K")
plt.ylabel("Sum of Squared Error")
```

C:\Users\Prathyusha\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\\_kmeans.py:870: FutureWarning: The default value of `n\_init` will change from 10 to 'auto' in 1.4. Set the value of `n\_init` explicitly to suppress the warning

```
warnings.warn(
```

C:\Users\Prathyusha\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\\_kmeans.py:870: FutureWarning: The default value of `n\_init` will change from 10 to 'auto' in 1.4. Set the value of `n\_init` explicitly to suppress the warning

```
warnings.warn(
```

C:\Users\Prathyusha\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\\_kmeans.py:870: FutureWarning: The default value of `n\_init` will change from 10 to 'auto' in 1.4. Set the value of `n\_init` explicitly to suppress the warning

```
warnings.warn(
```

C:\Users\Prathyusha\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\\_kmeans.py:870: FutureWarning: The default value of `n\_init` will change from 10 to 'auto' in 1.4. Set the value of `n\_init` explicitly to suppress the warning

```
warnings.warn(
```

C:\Users\Prathyusha\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\\_kmeans.py:870: FutureWarning: The default value of `n\_init` will change from 10 to 'auto' in 1.4. Set the value of `n\_init` explicitly to suppress the warning

```
warnings.warn(
```

C:\Users\Prathyusha\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\\_kmeans.py:870: FutureWarning: The default value of `n\_init` will change from 10 to 'auto' in 1.4. Set the value of `n\_init` explicitly to suppress the warning

```
warnings.warn(
```

C:\Users\Prathyusha\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\\_kmeans.py:870: FutureWarning: The default value of `n\_init` will change from 10 to 'auto' in 1.4. Set the value of `n\_init` explicitly to suppress the warning

```
warnings.warn(
```

C:\Users\Prathyusha\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\\_kmeans.py:870: FutureWarning: The default value of `n\_init` will change from 10 to 'auto' in 1.4. Set the value of `n\_init` explicitly to suppress the warning

```
warnings.warn(
```

C:\Users\Prathyusha\AppData\Local\Programs\Python\Python310\lib\site-packages\sklearn\cluster\\_kmeans.py:870: FutureWarning: The default value of `n\_init` will change from 10 to 'auto' in 1.4. Set the value of `n\_init` explicitly to suppress the warning

```
warnings.warn(
```

```
[27.81750759504308, 14.872296449956028, 10.2527514961052, 8.49005022151144, 7.030202097311372, 6.035990931162901, 5.117379110317933, 4.4444359608281525, 3.994645114512035]
```

Out[24]:

Text(0, 0.5, 'Sum of Squared Error')

