In [92]:
```
import numpy as np
import pandas as pd

import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.metrics import ConfusionMatrixDisplay, classification_report
```

In [15]:
```
# Loading the data
df = pd.read_csv("heart.csv")
df
```

Out[15]:

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR	ExAng	Oldpeak
0	1	63	1	typical	145	233	1	2	150	0	2.3
1	2	67	1	asymptomatic	160	286	0	2	108	1	1.5
2	3	67	1	asymptomatic	120	229	0	2	129	1	2.6
3	4	37	1	nonanginal	130	250	0	0	187	0	3.5
4	5	41	0	nontypical	130	204	0	2	172	0	1.4
...	...	...	...	...	...	...	...	...	...	...	...
298	299	45	1	typical	110	264	0	0	132	0	1.2
299	300	68	1	asymptomatic	144	193	1	0	141	0	3.4
300	301	57	1	asymptomatic	130	131	0	0	115	1	1.2
301	302	57	0	nontypical	130	236	0	2	174	0	0.0
302	303	38	1	nonanginal	138	175	0	0	173	0	0.0

303 rows × 15 columns

In [16]:
```
#first six rows
df.head(6)
```

Out[16]:

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR	ExAng	Oldpeak	Sl
0	1	63	1	typical	145	233	1	2	150	0	2.3	
1	2	67	1	asymptomatic	160	286	0	2	108	1	1.5	
2	3	67	1	asymptomatic	120	229	0	2	129	1	2.6	
3	4	37	1	nonanginal	130	250	0	0	187	0	3.5	
4	5	41	0	nontypical	130	204	0	2	172	0	1.4	
5	6	56	1	nontypical	120	236	0	0	178	0	0.8	

In [17]:
```
# datatype of each column
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 303 entries, 0 to 302
Data columns (total 15 columns):
 #   Column      Non-Null Count  Dtype  
---  ------      --------------  -----  
 0   Unnamed: 0  303 non-null    int64  
 1   Age         303 non-null    int64  
 2   Sex         303 non-null    int64  
 3   ChestPain   303 non-null    object 
 4   RestBP      303 non-null    int64  
 5   Chol        303 non-null    int64  
 6   Fbs         303 non-null    int64  
 7   RestECG     303 non-null    int64  
 8   MaxHR       303 non-null    int64  
 9   ExAng       303 non-null    int64  
 10  Oldpeak     303 non-null    float64
 11  Slope       303 non-null    int64  
 12  Ca          303 non-null    int64  
 13  Thal        303 non-null    object 
 14  AHD         303 non-null    object 
dtypes: float64(1), int64(11), object(3)
memory usage: 35.6+ KB
```

In [18]:
```
# shape of rows
df.shape[0]
```

Out[18]:
```
303
```

In [19]:
```
# shape of columns
df.shape[1]
```

Out[19]:
```
15
```

In [20]:
```
# find missing values
df.isnull().sum()
```

Out[20]:
```
Unnamed: 0    0
Age           0
Sex           0
ChestPain     0
RestBP        0
Chol          0
Fbs           0
RestECG       0
MaxHR         0
ExAng         0
Oldpeak       0
Slope         0
Ca            0
Thal          0
AHD           0
dtype: int64
```

In [23]:
```
# mean age of patients
df["Age"].mean()
```

Out[23]:
```
54.43894389438944
```

In [27]:
```
# each column
df.describe()
```

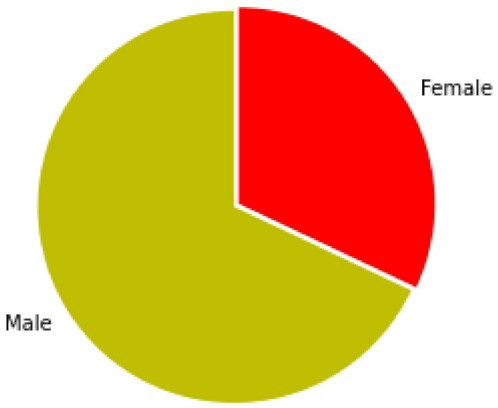
Out[27]:

	Unnamed: 0	Age	Sex	RestBP	Chol	Fbs	RestECG	Ma
count	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.000000	303.00
mean	152.000000	54.438944	0.679868	131.689769	246.693069	0.148515	0.990099	149.60
std	87.612784	9.038662	0.467299	17.599748	51.776918	0.356198	0.994971	22.87
min	1.000000	29.000000	0.000000	94.000000	126.000000	0.000000	0.000000	71.00
25%	76.500000	48.000000	0.000000	120.000000	211.000000	0.000000	0.000000	133.50
50%	152.000000	56.000000	1.000000	130.000000	241.000000	0.000000	1.000000	153.00
75%	227.500000	61.000000	1.000000	140.000000	275.000000	0.000000	2.000000	166.00
max	303.000000	77.000000	1.000000	200.000000	564.000000	1.000000	2.000000	202.00

In [85]:
```
#check fetures of various attributes
# 1.Sex
plt.figure(figsize=(6,4))

sizes=[len(df[df["Sex"] ==1]),
      len(df[df["Sex"] ==0])]
labels="Male","Female"

plt.pie(sizes, explode=(0.030,0), labels=labels,
        colors=["y","r"], startangle=90)
plt.axis("equal")
plt.show()
```

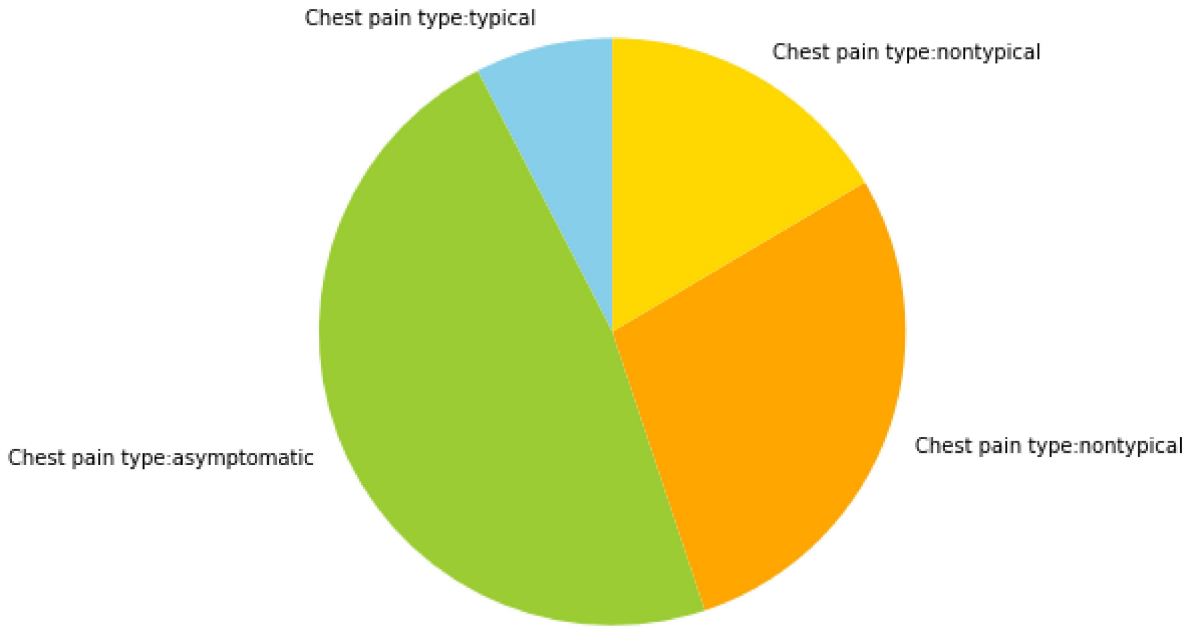


In [74]:
```
# 2.Chest Pain
plt.figure(figsize=(8,6))

label = "Chest pain type: typical","Chest pain type: asymptomatic","Chest pain ty
sizes = [len(df[df["ChestPain"] == "typical"]),
        len(df[df["ChestPain"] == "asymptomatic"]),
        len(df[df["ChestPain"] == "nonanginal"]),
        len(df[df["ChestPain"] == "nontypical"])]
colors = ["skyblue","yellowgreen","orange","gold"]
explode = (0,0,0,0)

plt.pie(sizes, explode = explode, labels= label, colors=colors,
       startangle=90)

plt.axis("equal")
plt.show()
```

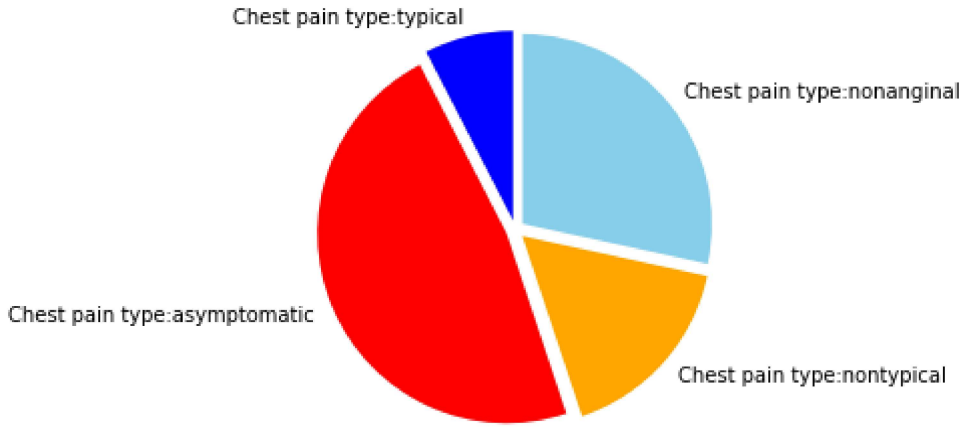


In [84]:
```
# ChestPain
plt.figure(figsize=(6,4))

labels = ( "Chest pain type:typical","Chest pain type:asymptomatic",
        "Chest pain type:nontypical","Chest pain type:nonanginal")
sizes = [len(df[df["ChestPain"] == "typical"]),
        len(df[df["ChestPain"] == "asymptomatic"]),
        len(df[df["ChestPain"] == "nontypical"]),
        len(df[df["ChestPain"] == "nonanginal"])]

plt.pie(sizes, explode=(0.05,0.05,0.05,0.05), labels = labels,
        colors=["b","r","orange","skyblue"], startangle=90)

plt.axis("equal")
plt.show()
```

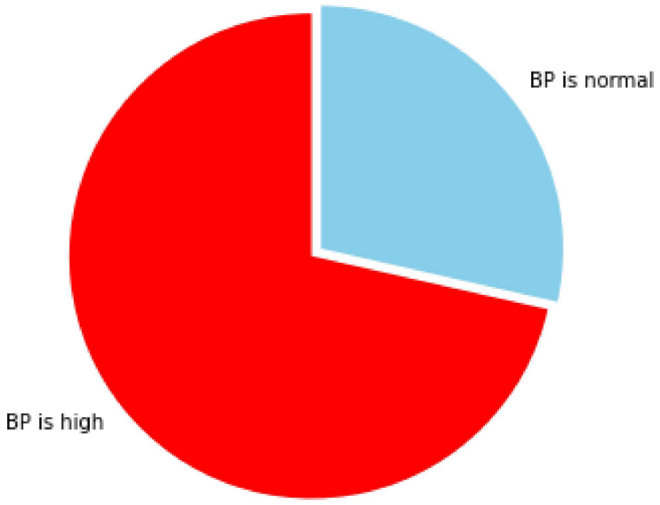


In [87]:
```
# 3.Rest bp
plt.figure(figsize=(7,5))

label = ("BP is high","BP is normal")
sizes = [len(df[df["RestBP"] >= 120]),
        len(df[df["RestBP"] <=120])]

plt.pie(sizes, explode=(0.05,0), labels=label,
        colors=["r","skyblue"], startangle=90)

plt.axis("equal")
plt.show()
```



In [93]:
```
# randomly divide dataset in training and testing

x_train,x_test = train_test_split(df,random_state=0,test_size=0.25)
```

In [94]:
```
x_train
```

Out[94]:

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR	ExAng	Oldpeak
173	174	62	0	asymptomatic	140	394	0	2	157	0	1.2
261	262	58	0	nontypical	136	319	1	2	152	0	0.0
37	38	57	1	asymptomatic	150	276	0	2	112	1	0.6
101	102	34	1	typical	118	182	0	2	174	0	0.0
166	167	52	1	nonanginal	138	223	0	0	169	0	0.0
...	...	...	...	...	...	...	...	...	...	...	...
251	252	58	1	asymptomatic	146	218	0	0	105	0	2.0
192	193	43	1	asymptomatic	132	247	1	2	143	1	0.1
117	118	35	0	asymptomatic	138	183	0	0	182	0	1.4
47	48	50	1	asymptomatic	150	243	0	2	128	0	2.6
172	173	59	0	asymptomatic	174	249	0	0	143	1	0.0

227 rows × 15 columns

In [95]:
```
x_test
```

Out[95]:

	Unnamed: 0	Age	Sex	ChestPain	RestBP	Chol	Fbs	RestECG	MaxHR	ExAng	Oldpeak
225	226	34	0	nontypical	118	210	0	0	192	0	0.7
152	153	67	0	nonanginal	115	564	0	2	160	0	1.6
228	229	54	1	asymptomatic	110	206	0	2	108	1	0.0
201	202	64	0	asymptomatic	180	325	0	0	154	1	0.0
52	53	44	1	asymptomatic	112	290	0	2	153	0	0.0
...	...	...	...	...	...	...	...	...	...	...	...
46	47	51	1	nonanginal	110	175	0	0	123	0	0.6
160	161	46	1	nontypical	101	197	1	0	156	0	0.0
232	233	49	1	nonanginal	118	149	0	2	126	0	0.8
181	182	56	0	asymptomatic	134	409	0	2	150	1	1.9
27	28	66	0	typical	150	226	0	0	114	0	2.6

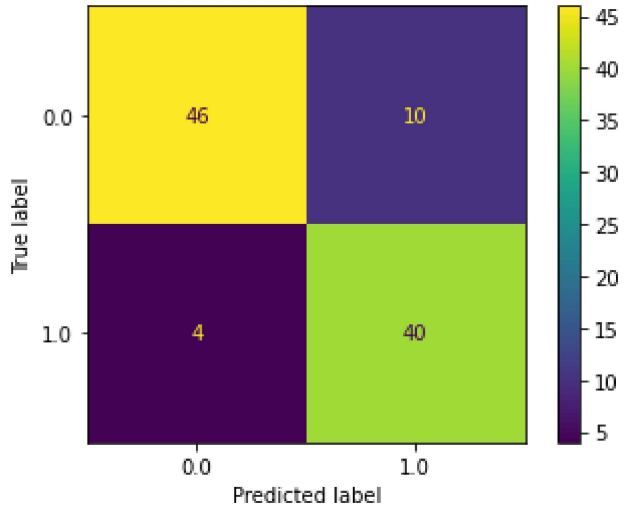
76 rows × 15 columns

In [101]:
```
actual = list(np.ones(44)) + list(np.zeros(56))
pred = list(np.ones(40)) + list(np.zeros(50)) + list(np.ones(10))
```

In [102]:
```
conf = ConfusionMatrixDisplay
```

In [103]:
```
conf.from_predictions(actual,pred)
```

Out[103]:
```
<sklearn.metrics._plot.confusion_matrix.ConfusionMatrixDisplay at 0x1f234cb3f10
>
```



In [105]:
```
print(classification_report(actual,pred))
```

```
              precision    recall  f1-score   support

         0.0       0.92      0.82      0.87        56
         1.0       0.80      0.91      0.85        44

    accuracy                           0.86       100
   macro avg       0.86      0.87      0.86       100
weighted avg       0.87      0.86      0.86       100
```

In [ ]: