

In [1]:

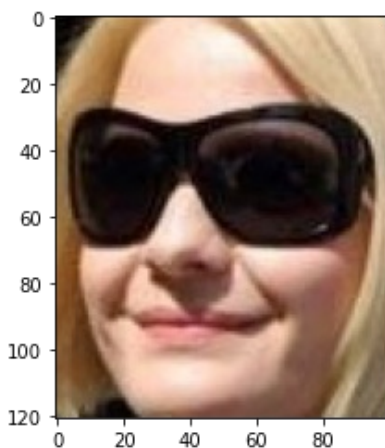
```
import tensorflow as tf
from tensorflow.keras import datasets, layers, models
from keras.preprocessing.image import ImageDataGenerator
import matplotlib.pyplot as plt
import numpy as np
from keras.preprocessing.image import ImageDataGenerator
from keras.models import Sequential
from keras.layers import Conv2D, MaxPooling2D
from keras.layers import Activation, Dropout, Flatten, Dense
from keras.preprocessing import image
import keras
import cv2
from PIL import ImageFile
from tensorflow.keras import datasets, layers, models
import matplotlib.pyplot as plt
%matplotlib inline
from keras.utils import to_categorical
from sklearn.model_selection import train_test_split
import pandas as pd
```

In [2]:

```
ImageFile.LOAD_TRUNCATED_IMAGES = True
img = image.load_img("/Users/mehme/Datasets/Gender Classification Dataset/gender/train/female/0005.jpg")
plt.imshow(img)

cv2.imread("/Users/mehme/Datasets/Gender Classification Dataset/gender/train/female/0005.jpg").shape

train = ImageDataGenerator(rescale = 1/255)
test = ImageDataGenerator(rescale = 1/255)
```



In [3]:

```
train_dataset = train.flow_from_directory("C:/Users/mehme/Datasets/Gender Classification Dataset/gender/train",
                                          target_size = (32,32),
                                          batch_size = 3,
                                          class_mode = "binary"
                                          )

test_dataset = test.flow_from_directory("C:/Users/mehme/Datasets/Gender Classification Dataset/gender/test",
                                       target_size = (32,32),
                                       batch_size = 3,
                                       class_mode = "binary"
                                       )
```

Found 3491 images belonging to 2 classes.
Found 200 images belonging to 2 classes.

In [4]:

```
train_dataset.class_indices
train_dataset.classes
```

Out[4]:

```
array([0, 0, 0, ..., 1, 1, 1])
```

In [5]:

```
test_dataset.class_indices
test_dataset.classes
```

Out[5]:

```
array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
       1, 1])
```

In [6]:

```
cnn = models.Sequential([
    tf.keras.layers.Conv2D(32, (3,3), activation="relu", input_shape=(32, 32, 3)),
    tf.keras.layers.MaxPool2D(2,2),

    tf.keras.layers.Flatten(),
    tf.keras.layers.Dense(64, activation="relu"),
    tf.keras.layers.Dense(2, activation="softmax")
])
```

In [7]:

```
cnn.compile(optimizer="adam",
            loss='sparse_categorical_crossentropy',
            metrics=['accuracy'])
```

In [9]:

```
history1 =cnn.fit(train_dataset,
                  steps_per_epoch = 50,
                  batch_size = 3,
                  epochs = 100,
                  validation_data = test_dataset
                )
```

Epoch 1/100

50/50 [=====] - 0s 8ms/step - loss: 0.0865 - accuracy: 0.9799 - val_loss: 0.2055 - val_accuracy: 0.9450

Epoch 2/100

50/50 [=====] - 0s 7ms/step - loss: 0.0529 - accuracy: 0.9867 - val_loss: 0.2380 - val_accuracy: 0.9350

Epoch 3/100

50/50 [=====] - 0s 7ms/step - loss: 0.1276 - accuracy: 0.9600 - val_loss: 0.2385 - val_accuracy: 0.9200

Epoch 4/100

50/50 [=====] - 0s 7ms/step - loss: 0.2485 - accuracy: 0.9200 - val_loss: 0.2069 - val_accuracy: 0.9150

Epoch 5/100

50/50 [=====] - 0s 7ms/step - loss: 0.1778 - accuracy: 0.9200 - val_loss: 0.2135 - val_accuracy: 0.9100

Epoch 6/100

50/50 [=====] - 0s 7ms/step - loss: 0.1572 - accuracy: 0.9667 - val_loss: 0.2010 - val_accuracy: 0.9050

```
val_loss: 0.2918 - val_accuracy: 0.8850
Epoch 7/100
50/50 [=====] - 0s 7ms/step - loss: 0.1459 - accuracy: 0.9200 -
val_loss: 0.2781 - val_accuracy: 0.9050
Epoch 8/100
50/50 [=====] - 0s 7ms/step - loss: 0.1833 - accuracy: 0.9133 -
val_loss: 0.2206 - val_accuracy: 0.9250
Epoch 9/100
50/50 [=====] - 0s 7ms/step - loss: 0.1178 - accuracy: 0.9667 -
val_loss: 0.2612 - val_accuracy: 0.9050
Epoch 10/100
50/50 [=====] - 0s 7ms/step - loss: 0.1815 - accuracy: 0.9267 -
val_loss: 0.2244 - val_accuracy: 0.9350
Epoch 11/100
50/50 [=====] - 0s 8ms/step - loss: 0.0745 - accuracy: 0.9733 -
val_loss: 0.2167 - val_accuracy: 0.9250
Epoch 12/100
50/50 [=====] - 0s 7ms/step - loss: 0.1042 - accuracy: 0.9733 -
val_loss: 0.2355 - val_accuracy: 0.9400
Epoch 13/100
50/50 [=====] - 0s 8ms/step - loss: 0.1296 - accuracy: 0.9467 -
val_loss: 0.1863 - val_accuracy: 0.9350
Epoch 14/100
50/50 [=====] - 0s 7ms/step - loss: 0.1211 - accuracy: 0.9600 -
val_loss: 0.2364 - val_accuracy: 0.9350
Epoch 15/100
50/50 [=====] - 0s 7ms/step - loss: 0.1003 - accuracy: 0.9733 -
val_loss: 0.2656 - val_accuracy: 0.9200
Epoch 16/100
50/50 [=====] - 0s 7ms/step - loss: 0.0823 - accuracy: 0.9667 -
val_loss: 0.2549 - val_accuracy: 0.9300
Epoch 17/100
50/50 [=====] - 0s 8ms/step - loss: 0.1648 - accuracy: 0.9400 -
val_loss: 0.2433 - val_accuracy: 0.9300
Epoch 18/100
50/50 [=====] - 0s 7ms/step - loss: 0.1086 - accuracy: 0.9400 -
val_loss: 0.2703 - val_accuracy: 0.8950
Epoch 19/100
50/50 [=====] - 0s 8ms/step - loss: 0.1255 - accuracy: 0.9400 -
val_loss: 0.2297 - val_accuracy: 0.9250
Epoch 20/100
50/50 [=====] - 0s 7ms/step - loss: 0.1376 - accuracy: 0.9667 -
val_loss: 0.2121 - val_accuracy: 0.9450
Epoch 21/100
50/50 [=====] - 0s 8ms/step - loss: 0.0808 - accuracy: 0.9533 -
val_loss: 0.2292 - val_accuracy: 0.9250
Epoch 22/100
50/50 [=====] - 0s 7ms/step - loss: 0.1283 - accuracy: 0.9600 -
val_loss: 0.2112 - val_accuracy: 0.9400
Epoch 23/100
50/50 [=====] - 0s 7ms/step - loss: 0.1861 - accuracy: 0.9133 -
val_loss: 0.2451 - val_accuracy: 0.9150
Epoch 24/100
50/50 [=====] - 0s 7ms/step - loss: 0.1021 - accuracy: 0.9733 -
val_loss: 0.2119 - val_accuracy: 0.9250
Epoch 25/100
50/50 [=====] - 0s 7ms/step - loss: 0.1241 - accuracy: 0.9667 -
val_loss: 0.2550 - val_accuracy: 0.9150
Epoch 26/100
50/50 [=====] - 0s 7ms/step - loss: 0.1587 - accuracy: 0.9400 -
val_loss: 0.2346 - val_accuracy: 0.9150
Epoch 27/100
50/50 [=====] - 0s 7ms/step - loss: 0.0534 - accuracy: 0.9933 -
val_loss: 0.2486 - val_accuracy: 0.9150
Epoch 28/100
50/50 [=====] - 0s 7ms/step - loss: 0.0599 - accuracy: 0.9733 -
val_loss: 0.2234 - val_accuracy: 0.9250
Epoch 29/100
50/50 [=====] - 0s 7ms/step - loss: 0.1390 - accuracy: 0.9667 -
val_loss: 0.2252 - val_accuracy: 0.9250
Epoch 30/100
50/50 [=====] - 0s 7ms/step - loss: 0.0940 - accuracy: 0.9533 -
val_loss: 0.2600 - val_accuracy: 0.8750
```

```
val_loss: 0.3699 - val_accuracy: 0.8750
Epoch 31/100
50/50 [=====] - 0s 7ms/step - loss: 0.0610 - accuracy: 0.9867 -
val_loss: 0.2623 - val_accuracy: 0.9050
Epoch 32/100
50/50 [=====] - 0s 7ms/step - loss: 0.0812 - accuracy: 0.9600 -
val_loss: 0.2405 - val_accuracy: 0.9150
Epoch 33/100
50/50 [=====] - 0s 7ms/step - loss: 0.1009 - accuracy: 0.9600 -
val_loss: 0.2075 - val_accuracy: 0.9400
Epoch 34/100
50/50 [=====] - 0s 7ms/step - loss: 0.1329 - accuracy: 0.9667 -
val_loss: 0.2464 - val_accuracy: 0.9000
Epoch 35/100
50/50 [=====] - 0s 7ms/step - loss: 0.0929 - accuracy: 0.9667 -
val_loss: 0.2273 - val_accuracy: 0.9250
Epoch 36/100
50/50 [=====] - 0s 7ms/step - loss: 0.0635 - accuracy: 0.9867 -
val_loss: 0.2614 - val_accuracy: 0.9350
Epoch 37/100
50/50 [=====] - 0s 7ms/step - loss: 0.0751 - accuracy: 0.9732 -
val_loss: 0.2544 - val_accuracy: 0.9300
Epoch 38/100
50/50 [=====] - 0s 7ms/step - loss: 0.0838 - accuracy: 0.9800 -
val_loss: 0.2463 - val_accuracy: 0.9300
Epoch 39/100
50/50 [=====] - 0s 7ms/step - loss: 0.1032 - accuracy: 0.9600 -
val_loss: 0.2848 - val_accuracy: 0.9200
Epoch 40/100
50/50 [=====] - 0s 8ms/step - loss: 0.1088 - accuracy: 0.9533 -
val_loss: 0.2653 - val_accuracy: 0.9200
Epoch 41/100
50/50 [=====] - 0s 7ms/step - loss: 0.0647 - accuracy: 0.9667 -
val_loss: 0.2562 - val_accuracy: 0.9200
Epoch 42/100
50/50 [=====] - 0s 8ms/step - loss: 0.1034 - accuracy: 0.9664 -
val_loss: 0.3020 - val_accuracy: 0.8900
Epoch 43/100
50/50 [=====] - 0s 7ms/step - loss: 0.1461 - accuracy: 0.9467 -
val_loss: 0.2419 - val_accuracy: 0.9450
Epoch 44/100
50/50 [=====] - 0s 8ms/step - loss: 0.0962 - accuracy: 0.9667 -
val_loss: 0.2881 - val_accuracy: 0.9100
Epoch 45/100
50/50 [=====] - 0s 7ms/step - loss: 0.0944 - accuracy: 0.9533 -
val_loss: 0.2135 - val_accuracy: 0.9500
Epoch 46/100
50/50 [=====] - 0s 8ms/step - loss: 0.1319 - accuracy: 0.9533 -
val_loss: 0.2260 - val_accuracy: 0.9450
Epoch 47/100
50/50 [=====] - 0s 7ms/step - loss: 0.0361 - accuracy: 0.9933 -
val_loss: 0.2147 - val_accuracy: 0.9400
Epoch 48/100
50/50 [=====] - 0s 7ms/step - loss: 0.0704 - accuracy: 0.9733 -
val_loss: 0.2547 - val_accuracy: 0.9550
Epoch 49/100
50/50 [=====] - 0s 7ms/step - loss: 0.1026 - accuracy: 0.9600 -
val_loss: 0.3199 - val_accuracy: 0.9150
Epoch 50/100
50/50 [=====] - 0s 7ms/step - loss: 0.0579 - accuracy: 0.9733 -
val_loss: 0.2500 - val_accuracy: 0.9300
Epoch 51/100
50/50 [=====] - 0s 7ms/step - loss: 0.0353 - accuracy: 0.9800 -
val_loss: 0.2640 - val_accuracy: 0.9350
Epoch 52/100
50/50 [=====] - 0s 7ms/step - loss: 0.0787 - accuracy: 0.9733 -
val_loss: 0.3080 - val_accuracy: 0.9200
Epoch 53/100
50/50 [=====] - 0s 7ms/step - loss: 0.1236 - accuracy: 0.9667 -
val_loss: 0.2445 - val_accuracy: 0.9050
Epoch 54/100
50/50 [=====] - 0s 7ms/step - loss: 0.0791 - accuracy: 0.9667 -
val_loss: 0.2200 - val_accuracy: 0.9500
```

```
val_loss: 0.2008 - val_accuracy: 0.9500
Epoch 55/100
50/50 [=====] - 0s 7ms/step - loss: 0.0747 - accuracy: 0.9733 -
val_loss: 0.2163 - val_accuracy: 0.9400
Epoch 56/100
50/50 [=====] - 0s 7ms/step - loss: 0.1075 - accuracy: 0.9600 -
val_loss: 0.2395 - val_accuracy: 0.9300
Epoch 57/100
50/50 [=====] - 0s 7ms/step - loss: 0.0639 - accuracy: 0.9667 -
val_loss: 0.2792 - val_accuracy: 0.9150
Epoch 58/100
50/50 [=====] - 0s 7ms/step - loss: 0.0368 - accuracy: 0.9867 -
val_loss: 0.3624 - val_accuracy: 0.9000
Epoch 59/100
50/50 [=====] - 0s 8ms/step - loss: 0.0750 - accuracy: 0.9600 -
val_loss: 0.2135 - val_accuracy: 0.9450
Epoch 60/100
50/50 [=====] - 0s 7ms/step - loss: 0.1208 - accuracy: 0.9600 -
val_loss: 0.2500 - val_accuracy: 0.9150
Epoch 61/100
50/50 [=====] - 0s 7ms/step - loss: 0.0704 - accuracy: 0.9733 -
val_loss: 0.2197 - val_accuracy: 0.9200
Epoch 62/100
50/50 [=====] - 0s 7ms/step - loss: 0.0678 - accuracy: 0.9800 -
val_loss: 0.2474 - val_accuracy: 0.9200
Epoch 63/100
50/50 [=====] - 0s 7ms/step - loss: 0.0646 - accuracy: 0.9733 -
val_loss: 0.2219 - val_accuracy: 0.9250
Epoch 64/100
50/50 [=====] - 0s 7ms/step - loss: 0.1114 - accuracy: 0.9467 -
val_loss: 0.2908 - val_accuracy: 0.8950
Epoch 65/100
50/50 [=====] - 0s 7ms/step - loss: 0.2274 - accuracy: 0.9067 -
val_loss: 0.2255 - val_accuracy: 0.9050
Epoch 66/100
50/50 [=====] - 0s 7ms/step - loss: 0.0686 - accuracy: 0.9800 -
val_loss: 0.3386 - val_accuracy: 0.8800
Epoch 67/100
50/50 [=====] - 0s 7ms/step - loss: 0.0692 - accuracy: 0.9933 -
val_loss: 0.2468 - val_accuracy: 0.9400
Epoch 68/100
50/50 [=====] - 0s 7ms/step - loss: 0.1128 - accuracy: 0.9400 -
val_loss: 0.2411 - val_accuracy: 0.9100
Epoch 69/100
50/50 [=====] - 0s 8ms/step - loss: 0.0883 - accuracy: 0.9867 -
val_loss: 0.2190 - val_accuracy: 0.9400
Epoch 70/100
50/50 [=====] - 0s 7ms/step - loss: 0.0225 - accuracy: 1.0000 -
val_loss: 0.2580 - val_accuracy: 0.9300
Epoch 71/100
50/50 [=====] - 0s 7ms/step - loss: 0.0420 - accuracy: 0.9867 -
val_loss: 0.2323 - val_accuracy: 0.9400
Epoch 72/100
50/50 [=====] - 0s 8ms/step - loss: 0.0440 - accuracy: 0.9933 -
val_loss: 0.2282 - val_accuracy: 0.9500
Epoch 73/100
50/50 [=====] - 0s 7ms/step - loss: 0.0405 - accuracy: 0.9800 -
val_loss: 0.2276 - val_accuracy: 0.9400
Epoch 74/100
50/50 [=====] - 0s 7ms/step - loss: 0.0545 - accuracy: 0.9733 -
val_loss: 0.2545 - val_accuracy: 0.9350
Epoch 75/100
50/50 [=====] - 0s 7ms/step - loss: 0.1580 - accuracy: 0.9667 -
val_loss: 0.2277 - val_accuracy: 0.9400
Epoch 76/100
50/50 [=====] - 0s 7ms/step - loss: 0.0816 - accuracy: 0.9667 -
val_loss: 0.2281 - val_accuracy: 0.9250
Epoch 77/100
50/50 [=====] - 0s 7ms/step - loss: 0.0953 - accuracy: 0.9667 -
val_loss: 0.3295 - val_accuracy: 0.9200
Epoch 78/100
50/50 [=====] - 0s 7ms/step - loss: 0.1640 - accuracy: 0.9333 -
val_loss: 0.4727 - val_accuracy: 0.9250
```

```

val_loss: 0.4727 - val_accuracy: 0.8350
Epoch 79/100
50/50 [=====] - 0s 7ms/step - loss: 0.0956 - accuracy: 0.9533 -
val_loss: 0.2062 - val_accuracy: 0.9500
Epoch 80/100
50/50 [=====] - 0s 7ms/step - loss: 0.0314 - accuracy: 0.9933 -
val_loss: 0.2207 - val_accuracy: 0.9150
Epoch 81/100
50/50 [=====] - 0s 7ms/step - loss: 0.1005 - accuracy: 0.9800 -
val_loss: 0.2088 - val_accuracy: 0.9500
Epoch 82/100
50/50 [=====] - 0s 8ms/step - loss: 0.0349 - accuracy: 0.9933 -
val_loss: 0.2745 - val_accuracy: 0.9300
Epoch 83/100
50/50 [=====] - 0s 7ms/step - loss: 0.0347 - accuracy: 0.9733 -
val_loss: 0.3763 - val_accuracy: 0.8950
Epoch 84/100
50/50 [=====] - 0s 7ms/step - loss: 0.0724 - accuracy: 0.9733 -
val_loss: 0.3398 - val_accuracy: 0.9250
Epoch 85/100
50/50 [=====] - 0s 7ms/step - loss: 0.1291 - accuracy: 0.9533 -
val_loss: 0.3123 - val_accuracy: 0.9050
Epoch 86/100
50/50 [=====] - 0s 7ms/step - loss: 0.0900 - accuracy: 0.9600 -
val_loss: 0.2771 - val_accuracy: 0.9250
Epoch 87/100
50/50 [=====] - 0s 7ms/step - loss: 0.0732 - accuracy: 0.9667 -
val_loss: 0.2684 - val_accuracy: 0.9350
Epoch 88/100
50/50 [=====] - 0s 7ms/step - loss: 0.0851 - accuracy: 0.9667 -
val_loss: 0.2383 - val_accuracy: 0.9200
Epoch 89/100
50/50 [=====] - 0s 7ms/step - loss: 0.0740 - accuracy: 0.9800 -
val_loss: 0.2275 - val_accuracy: 0.9300
Epoch 90/100
50/50 [=====] - 0s 7ms/step - loss: 0.0530 - accuracy: 0.9733 -
val_loss: 0.2377 - val_accuracy: 0.9350
Epoch 91/100
50/50 [=====] - 0s 7ms/step - loss: 0.0518 - accuracy: 0.9867 -
val_loss: 0.2539 - val_accuracy: 0.9300
Epoch 92/100
50/50 [=====] - 0s 7ms/step - loss: 0.0431 - accuracy: 0.9800 -
val_loss: 0.2730 - val_accuracy: 0.9300
Epoch 93/100
50/50 [=====] - 0s 7ms/step - loss: 0.0344 - accuracy: 0.9933 -
val_loss: 0.2846 - val_accuracy: 0.9200
Epoch 94/100
50/50 [=====] - 0s 7ms/step - loss: 0.0202 - accuracy: 0.9933 -
val_loss: 0.2544 - val_accuracy: 0.9350
Epoch 95/100
50/50 [=====] - 0s 7ms/step - loss: 0.0514 - accuracy: 0.9800 -
val_loss: 0.2929 - val_accuracy: 0.9200
Epoch 96/100
50/50 [=====] - 0s 7ms/step - loss: 0.0228 - accuracy: 0.9933 -
val_loss: 0.3022 - val_accuracy: 0.9150
Epoch 97/100
50/50 [=====] - 0s 7ms/step - loss: 0.0145 - accuracy: 1.0000 -
val_loss: 0.3245 - val_accuracy: 0.9250
Epoch 98/100
50/50 [=====] - 0s 7ms/step - loss: 0.0817 - accuracy: 0.9667 -
val_loss: 0.3578 - val_accuracy: 0.9200
Epoch 99/100
50/50 [=====] - 0s 8ms/step - loss: 0.0937 - accuracy: 0.9467 -
val_loss: 0.1895 - val_accuracy: 0.9400
Epoch 100/100
50/50 [=====] - 0s 7ms/step - loss: 0.0655 - accuracy: 0.9667 -
val_loss: 0.2171 - val_accuracy: 0.9200

```

In [10]:

```
cnn.summary()
```

```
Model: "sequential_1"
```

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 30, 30, 32)	896
max_pooling2d (MaxPooling2D)	(None, 15, 15, 32)	0
flatten (Flatten)	(None, 7200)	0
dense (Dense)	(None, 64)	460864
dense_1 (Dense)	(None, 2)	130
Total params: 461,890		
Trainable params: 461,890		
Non-trainable params: 0		

In [12]:

```
dosya = open("C:/Users/mehme/Datasets/Gender Classification Dataset/gender/deneme.txt","r",encoding="utf-8")
dizi = dosya.read().split('\n')

#ConfusionMatrix degerleri
tp = 0
tn = 0
fp = 0
fn = 0

for i in range(1, 101):

    path = 'C:/Users/mehme/Datasets/Gender Classification Dataset/gender/deneme/' + '('+str(i) + ')' + '.jpg'
    test_image = image.load_img(path , target_size=(32, 32))
    test_image = image.img_to_array(test_image)
    test_image = np.expand_dims(test_image, axis = 0)
    result = cnn.predict(test_image)
    train_dataset.class_indices
    if result[0][0] == 1:
        prediction = 'Kadın'
    else:
        prediction = 'Erkek'
    print(str(i)+ ".Deger : " + prediction + " - Gercek Deger : " + dizi[i] )
    if str(prediction) == 'Kadın' and str(dizi[i]) == 'Kadın':
        tp = tp + 1
    if str(prediction) == 'Erkek' and str(dizi[i]) == 'Erkek':
        tn = tn + 1
    if str(prediction) == 'Kadın' and str(dizi[i]) == 'Kadın':
        fp = fp + 1
    if str(prediction) == 'Kadın' and str(dizi[i]) == 'Kadın':
        fn = fn + 1

print('-----')
print('Confusion Matrix Sonuclari :")
print("")
print("TP : " + str(tp) + " TN : " + str(tn) + " FP : " + str(fp) + " FN : " + str(fn) )
print("Recall : " + str(tp/(tp+fn)))
print("Precision : " + str(tp/(tp+fp)))
print("Accuracy : " + str((tp+tn)/100))
print("F1 - Score : " + str((2*tp)/(2*tp+fp+fn)))
#tp-kadın kadın
#tn-erkek erkek
#fp-kadın erkek
#fn-erkek kadın
```

- 1.Deger : Kadın - Gercek Deger : Kadın
- 2.Deger : Kadın - Gercek Deger : Kadın
- 3.Deger : Kadın - Gercek Deger : Kadın
- 4.Deger : Kadın - Gercek Deger : Kadın
- 5.Deger : Kadın - Gercek Deger : Kadın

[illegible]


```
77.Deger : Erkek - Gercek Deger : Erkek
78.Deger : Erkek - Gercek Deger : Erkek
79.Deger : Erkek - Gercek Deger : Erkek
80.Deger : Erkek - Gercek Deger : Erkek
81.Deger : Erkek - Gercek Deger : Erkek
82.Deger : Erkek - Gercek Deger : Erkek
83.Deger : Kadın - Gercek Deger : Erkek
84.Deger : Erkek - Gercek Deger : Erkek
85.Deger : Erkek - Gercek Deger : Erkek
86.Deger : Erkek - Gercek Deger : Erkek
87.Deger : Erkek - Gercek Deger : Erkek
88.Deger : Erkek - Gercek Deger : Erkek
89.Deger : Erkek - Gercek Deger : Erkek
90.Deger : Erkek - Gercek Deger : Erkek
91.Deger : Kadın - Gercek Deger : Erkek
92.Deger : Erkek - Gercek Deger : Erkek
93.Deger : Erkek - Gercek Deger : Erkek
94.Deger : Erkek - Gercek Deger : Erkek
95.Deger : Erkek - Gercek Deger : Erkek
96.Deger : Erkek - Gercek Deger : Erkek
97.Deger : Erkek - Gercek Deger : Erkek
98.Deger : Erkek - Gercek Deger : Erkek
99.Deger : Erkek - Gercek Deger : Erkek
100.Deger : Erkek - Gercek Deger : Erkek
```

Confusion Matrix Sonuclari :

TP : 46 TN : 47 FP : 46 FN : 46
Recall : 0.5
Precision : 0.5
Accuracy : 0.93
F1 - Score : 0.5

In [14]:

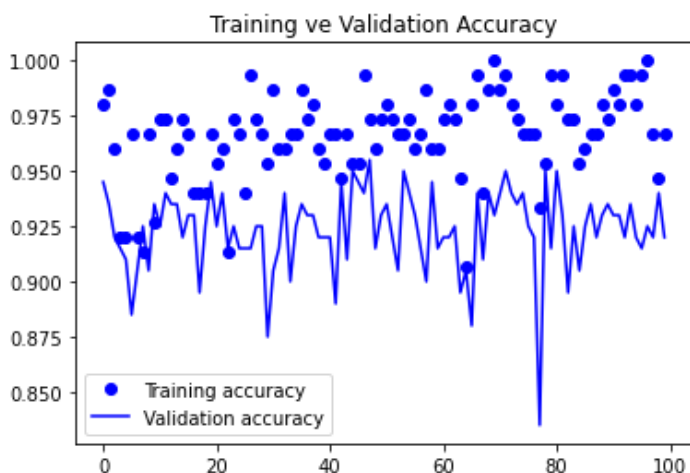
```
import matplotlib.pyplot as plt
%matplotlib inline

accuracy = history1.history['accuracy']
val_accuracy = history1.history['val_accuracy']
loss = history1.history['loss']
val_loss = history1.history['val_loss']
epochs = range(len(accuracy))

plt.plot(epochs, accuracy, 'bo', label='Training accuracy')
plt.plot(epochs, val_accuracy, 'b', label='Validation accuracy')
plt.title('Training ve Validation Accuracy')
plt.legend()
plt.figure()
```

Out[14]:

<Figure size 432x288 with 0 Axes>



<Figure size 432x288 with 0 Axes>

In []: