

In [2]:

```
import re
from sklearn.datasets import load_digits
from sklearn.model_selection import train_test_split
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sb
from sklearn import metrics
```

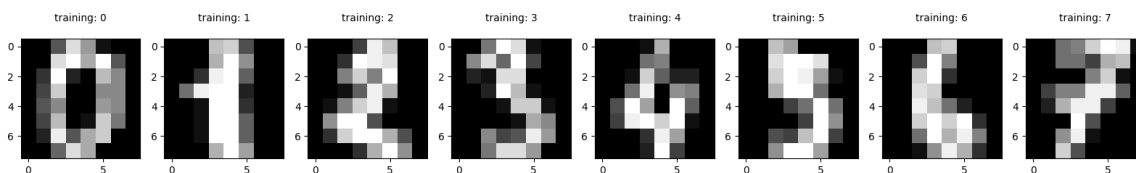
In [3]:

```
%matplotlib inline
digits=load_digits()
print("image data shape",digits.data.shape)
print("Label data shape",digits.target.shape)
```

```
image data shape (1797, 64)
Label data shape (1797,)
```

In [34]:

```
plt.figure(figsize=(20,4))
for index,(image,label) in enumerate(zip(digits.data[0:8],digits.target[0:8])):
    plt.subplot(1,8,index+1)
    plt.imshow(np.reshape(image,(8,8)),cmap=plt.cm.gray)
    plt.title('training: %i\n'%label,fontsize=10)
```



In [33]:

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(digits.data,digits.target,test_size=0.30,r
```

In [9]:

```
print(x_train.shape)
```

```
(1257, 64)
```

In [10]:

```
print(y_train.shape)
```

```
(1257,)
```

In [11]:

```
print(x_test.shape)
```

(540, 64)

In [12]:

```
print(y_test.shape)
```

(540,)

In [13]:

```
from sklearn.linear_model import LogisticRegression
a=LogisticRegression(max_iter=2000)
a.fit(x_train,y_train)
print(a.predict(x_test))
```

```
[0 4 1 2 0 0 8 7 6 6 3 6 8 7 4 7 4 3 2 6 7 3 4 7 1 0 7 4 8 3 4 0 5 5 5 1 2
 9 0 0 0 8 2 3 7 0 1 7 1 3 8 4 2 9 6 0 4 5 4 8 7 9 9 5 0 3 7 4 9 1 8 0 9 3
 0 2 7 8 1 1 9 3 3 2 2 3 8 2 4 6 9 5 8 4 7 3 7 2 5 6 5 6 1 0 6 4 8 1 5 6 8
 8 8 6 4 0 2 7 5 0 8 5 8 4 7 0 5 9 0 1 8 4 7 9 6 1 2 7 1 3 5 3 9 2 7 4 9 2
 0 0 9 2 8 4 0 9 7 0 1 4 1 8 0 7 9 1 9 7 2 7 7 0 5 3 4 0 5 2 3 0 3 0 1 9 5
 1 8 2 6 0 9 7 8 8 7 3 0 9 9 3 6 3 7 9 0 4 7 6 5 3 1 0 4 3 1 0 5 3 7 3 2 3
 5 0 7 4 5 3 0 0 5 7 5 2 4 2 9 3 3 0 8 2 1 3 1 7 4 7 4 9 2 1 1 4 7 1 7 9 2
 5 2 5 0 9 2 0 7 6 5 4 5 1 1 1 8 0 7 4 7 1 2 4 9 5 0 3 0 5 1 3 6 1 4 3 8 5
 2 7 4 6 4 8 3 0 1 5 7 7 8 3 4 8 8 5 2 2 1 7 8 3 9 8 4 4 2 5 5 0 1 9 1 0 8
 1 4 0 6 9 9 7 1 1 1 0 8 2 4 3 3 4 1 7 6 6 8 3 6 6 2 0 1 8 4 2 2 7 6 7 0 8
 4 8 0 4 6 2 1 9 5 6 0 1 3 2 2 9 4 7 2 2 2 0 8 7 1 8 8 3 2 4 1 1 1 2 1 8 3
 0 0 9 6 6 5 5 8 3 7 4 6 8 1 3 6 4 3 2 6 8 2 1 2 1 4 1 4 8 6 3 2 1 2 0 0 1
 6 1 6 9 7 7 9 3 8 5 5 7 1 4 1 7 8 9 8 3 0 2 2 9 0 4 6 7 3 1 5 9 9 7 9 4 0
 5 7 5 8 5 2 7 8 5 0 8 9 6 6 1 0 3 1 1 3 8 1 2 0 2 6 0 2 4 8 3 4 8 8 4 8 9
 7 3 5 1 9 2 9 2 9 0 6 3 5 6 8 9 9 0 9 1 2 2]
```

C:\Users\MY HOME\AppData\Local\Programs\Python\Python311\Lib\site-packages\sklearn\linear_model_logistic.py:458: ConvergenceWarning: lbfgs failed to converge (status=1):

STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:

<https://scikit-learn.org/stable/modules/preprocessing.html> (<https://scikit-learn.org/stable/modules/preprocessing.html>)

Please also refer to the documentation for alternative solver options:

https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression (https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression)

```
n_iter_i = _check_optimize_result(
```

In [14]:

```
score=a.score(x_test,y_test)
print(score)
```

0.9444444444444444

In []:

In []: