In [ ]:

```
import pandas as pd
import torch
from torchvision.io import read_image
from torch.utils.data import Dataset, DataLoader
import os

image_width, image_height = 28, 28
```

Create a custom dataset with the flowers dataset.

In [ ]:

```
class FlowerDataset(Dataset):
    def __init__(self, flowers_labels, img_dir):
        self.img_labels = pd.read_csv(flowers_labels)
        self.img_dir = img_dir

    def __len__(self):
        return len(self.img_labels)

    # Assuming flowers.csv has image path and label is in index 0 and 1
    def __getitem__(self, idx):
        img_path = os.path.join(self.img_dir, self.img_labels.iloc[idx, 0])
        image = read_image(img_path)
        label = self.img_labels.iloc[idx, 1]
        return image, label
```

Create a data loader with the flowers dataset with shuffling

In [ ]:

```
flower_dataset = FlowerDataset(flowers_labels='flowers.csv', img_dir='flower
train_dataloader = DataLoader(flower_dataset, batch_size=32, shuffle=True)
```

Fetch a batch from the dataloader

In [ ]:

```
train_features, train_labels = next(iter(train_dataloader))
```

Create a simple neural network with a Linear layer and Relu layer

In [ ]:

```
class NeuralNetwork(torch.nn.Module):
    def __init__(self):
        super(NeuralNetwork, self).__init__()
        self.flatten = torch.nn.Flatten()
        self.linear_relu = torch.nn.Sequential(
            torch.nn.Linear(image_width*image_height, 512),
            torch.nn.ReLU(),
            torch.nn.Linear(512, 512),
            torch.nn.ReLU(),
            torch.nn.Linear(512, 10),
        )
    def forward(self, x):
        x = self.flatten(x)
        return self.linear_relu(x)
```