

# Loading data with pytorch Dataloader

Dataset link: <https://www.kaggle.com/datasets/alxmamaev/flowers-recognition>

- Below function is gonna map into mini-batch with given actual parameter to the formal parameter of batch\_size
  - `train_data_loader= DataLoader(train_data, batch_size=32, shuffle=True)`
- 
- Iterate through `train_data_loader` in mini-batch

for features, labels in `train_data_loader`:

```
print(features.shape)
```

```
print(labels.shape)
```

# Prepare labels

```
labels=set()
root=r'/home/jeff/demo/pytorch-flowers-classification/flowers'
labels=sorted(os.listdir(root))

label_dict=dict()
for i in range(len(labels)):
    label_dict[labels[i]]=i

print(label_dict)
```

- How to customize **train\_data**?

```
class MyDataset(torch.utils.data.Dataset):
    def __init__(self, feature_list, label_list):
        self.feature_list = feature_list
        self.label_list = label_list

    def __getitem__(self, index):

        self.feature_list[index] = cv2.resize(self.feature_list[index], (100, 100))/255.

        feature = torch.from_numpy(self.feature_list[index]).float().permute(2,0,1)

        label = torch.from_numpy(np.asarray(self.label_list[index])).long()

        return feature, label

    def __len__(self):
        return len(self.label_list)
```