1. 라이브러리 불러오기

In [52]:
```
import tensorflow as tf
import numpy as np
from tensorflow.keras import datasets, layers, models
from keras.utils import plot_model
import matplotlib.pyplot as plt # !pip install matplotlib # 외부에서 설치
import pydot
from PIL import Image, ImageOps
```

2. 데이터 읽어오기

In [2]:
```
# 훈련용(Train -> 질문 : 답), 평가용(test -> 질문 : 답) 데이터 분류해서 가져오기
# 데이터가 수정되면 안됨 -> 튜플로 가져오기
(train_images, train_labels), (test_images, test_labels) = datasets.mnist.load_d
```

In [3]:
```
# 데이터 확인
train_images.shape, train_labels.shape, test_images.shape, test_labels.shape
# ((60000, 28, 28) -> 흑백 이미지(채널이 없음 -> 60000, 28, 28, 3 -> 컬러), (600
# 생략된 채널을 넣어줘야 한다. -> 구조를 reshape 사용해서 조정
```

Out[3]:
```
((60000, 28, 28), (60000,), (10000, 28, 28), (10000,))
```

In [24]:
```
# !pip install tensorflow_datasets
```

```
Collecting tensorflow_datasets
  Downloading tensorflow_datasets-4.9.4-py3-none-any.whl.metadata (9.2 kB)
Requirement already satisfied: absl-py in c:\users\hi\anaconda3\envs\p310_cnn\lib
\site-packages (from tensorflow_datasets) (2.1.0)
Collecting click (from tensorflow_datasets)
  Downloading click-8.1.7-py3-none-any.whl.metadata (3.0 kB)
Collecting dm-tree (from tensorflow_datasets)
  Downloading dm_tree-0.1.8-cp310-cp310-win_amd64.whl.metadata (2.0 kB)
Collecting etils>=0.9.0 (from etils[enp,epath,etree]>=0.9.0->tensorflow_datasets)
  Downloading etils-1.7.0-py3-none-any.whl.metadata (6.4 kB)
Requirement already satisfied: numpy in c:\users\hi\anaconda3\envs\p310_cnn\lib\s
ite-packages (from tensorflow_datasets) (1.26.4)
Collecting promise (from tensorflow_datasets)
  Downloading promise-2.3.tar.gz (19 kB)
  Preparing metadata (setup.py): started
  Preparing metadata (setup.py): finished with status 'done'
Requirement already satisfied: protobuf>=3.20 in c:\users\hi\anaconda3\envs\p310_
cnn\lib\site-packages (from tensorflow_datasets) (4.25.3)
Requirement already satisfied: psutil in c:\users\hi\anaconda3\envs\p310_cnn\lib
\site-packages (from tensorflow_datasets) (5.9.8)
Requirement already satisfied: requests>=2.19.0 in c:\users\hi\anaconda3\envs\p31
0_cnn\lib\site-packages (from tensorflow_datasets) (2.31.0)
Collecting tensorflow-metadata (from tensorflow_datasets)
  Downloading tensorflow_metadata-1.15.0-py3-none-any.whl.metadata (2.4 kB)
Requirement already satisfied: termcolor in c:\users\hi\anaconda3\envs\p310_cnn\l
ib\site-packages (from tensorflow_datasets) (2.4.0)
Collecting toml (from tensorflow_datasets)
  Downloading toml-0.10.2-py2.py3-none-any.whl.metadata (7.1 kB)
Collecting tqdm (from tensorflow_datasets)
  Downloading tqdm-4.66.4-py3-none-any.whl.metadata (57 kB)
     ---------------------------------------- 0.0/57.6 kB ? eta -:--:--
     ---------------------------------------- 57.6/57.6 kB 3.0 MB/s eta 0:00:00
Requirement already satisfied: wrapt in c:\users\hi\anaconda3\envs\p310_cnn\lib\s
ite-packages (from tensorflow_datasets) (1.16.0)
Collecting fsspec (from etils[enp,epath,etree]>=0.9.0->tensorflow_datasets)
  Downloading fsspec-2024.5.0-py3-none-any.whl.metadata (11 kB)
Collecting importlib_resources (from etils[enp,epath,etree]>=0.9.0->tensorflow_da
tasets)
  Downloading importlib_resources-6.4.0-py3-none-any.whl.metadata (3.9 kB)
Requirement already satisfied: typing_extensions in c:\users\hi\anaconda3\envs\p3
10_cnn\lib\site-packages (from etils[enp,epath,etree]>=0.9.0->tensorflow_dataset
s) (4.11.0)
Collecting zipp (from etils[enp,epath,etree]>=0.9.0->tensorflow_datasets)
  Downloading zipp-3.18.2-py3-none-any.whl.metadata (3.5 kB)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\hi\anaconda3
\envs\p310_cnn\lib\site-packages (from requests>=2.19.0->tensorflow_datasets) (3.
3.2)
Requirement already satisfied: idna<4,>=2.5 in c:\users\hi\anaconda3\envs\p310_cn
n\lib\site-packages (from requests>=2.19.0->tensorflow_datasets) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\hi\anaconda3\envs\p
310_cnn\lib\site-packages (from requests>=2.19.0->tensorflow_datasets) (2.2.1)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\hi\anaconda3\envs\p
310_cnn\lib\site-packages (from requests>=2.19.0->tensorflow_datasets) (2024.2.2)
Requirement already satisfied: colorama in c:\users\hi\anaconda3\envs\p310_cnn\li
b\site-packages (from click->tensorflow_datasets) (0.4.6)
Requirement already satisfied: six in c:\users\hi\anaconda3\envs\p310_cnn\lib\sit
e-packages (from promise->tensorflow_datasets) (1.16.0)
Collecting protobuf>=3.20 (from tensorflow_datasets)
  Downloading protobuf-3.20.3-cp310-cp310-win_amd64.whl.metadata (698 bytes)
Downloading tensorflow_datasets-4.9.4-py3-none-any.whl (5.1 MB)
```

```
   ---------------------------------------- 0.0/5.1 MB ? eta -:--:--
   ----- ---------------------------------- 0.7/5.1 MB 21.5 MB/s eta 0:00:01
   ------------- -------------------------- 1.7/5.1 MB 21.4 MB/s eta 0:00:01
   ------------------------- -------------- 3.2/5.1 MB 22.6 MB/s eta 0:00:01
   ---------------------------------------  5.0/5.1 MB 29.5 MB/s eta 0:00:01
   ---------------------------------------- 5.1/5.1 MB 24.9 MB/s eta 0:00:00
Downloading etils-1.7.0-py3-none-any.whl (152 kB)
   ---------------------------------------- 0.0/152.4 kB ? eta -:--:--
   ---------------------------------------- 152.4/152.4 kB 8.9 MB/s eta 0:00:00
Downloading click-8.1.7-py3-none-any.whl (97 kB)
   ---------------------------------------- 0.0/97.9 kB ? eta -:--:--
   ---------------------------------------- 97.9/97.9 kB ? eta 0:00:00
Downloading dm_tree-0.1.8-cp310-cp310-win_amd64.whl (101 kB)
   ---------------------------------------- 0.0/101.3 kB ? eta -:--:--
   ---------------------------------------- 101.3/101.3 kB 6.1 MB/s eta 0:00:00
Downloading tensorflow_metadata-1.15.0-py3-none-any.whl (28 kB)
Downloading protobuf-3.20.3-cp310-cp310-win_amd64.whl (904 kB)
   ---------------------------------------- 0.0/904.0 kB ? eta -:--:--
   --------------------------------------- 904.0/904.0 kB 55.9 MB/s eta 0:00:00
Downloading toml-0.10.2-py2.py3-none-any.whl (16 kB)
Downloading tqdm-4.66.4-py3-none-any.whl (78 kB)
   ---------------------------------------- 0.0/78.3 kB ? eta -:--:--
   ---------------------------------------- 78.3/78.3 kB 4.3 MB/s eta 0:00:00
Downloading fsspec-2024.5.0-py3-none-any.whl (316 kB)
   ---------------------------------------- 0.0/316.1 kB ? eta -:--:--
   --------------------------------------- 316.1/316.1 kB 19.1 MB/s eta 0:00:00
Downloading importlib_resources-6.4.0-py3-none-any.whl (38 kB)
Downloading zipp-3.18.2-py3-none-any.whl (8.3 kB)
Building wheels for collected packages: promise
  Building wheel for promise (setup.py): started
  Building wheel for promise (setup.py): finished with status 'done'
  Created wheel for promise: filename=promise-2.3-py3-none-any.whl size=21544 sha
256=05f4dab0975997600e365a49c2adab7b8ad2a27941959a56d6be1a9922020ef1
  Stored in directory: c:\users\hi\appdata\local\pip\cache\wheels\54\4e\28\3ed0e1
c8a752867445bab994d2340724928aa3ab059c57c8db
Successfully built promise
Installing collected packages: dm-tree, zipp, tqdm, toml, protobuf, promise, impo
rtlib_resources, fsspec, etils, click, tensorflow-metadata, tensorflow_datasets
  Attempting uninstall: protobuf
    Found existing installation: protobuf 4.25.3
    Uninstalling protobuf-4.25.3:
      Successfully uninstalled protobuf-4.25.3
Successfully installed click-8.1.7 dm-tree-0.1.8 etils-1.7.0 fsspec-2024.5.0 impo
rtlib_resources-6.4.0 promise-2.3 protobuf-3.20.3 tensorflow-metadata-1.15.0 tens
orflow_datasets-4.9.4 toml-0.10.2 tqdm-4.66.4 zipp-3.18.2
```

```
  WARNING: Failed to remove contents in a temporary directory 'C:\Users\hi\anacon
da3\envs\p310_cnn\Lib\site-packages\google\~upb'.
  You can safely remove it manually.
```

In [4]:

```
import tensorflow_datasets as tfds # tf에서 제공되는 ds
tfds.list_builders()
```

Out[4]:
```
['abstract_reasoning',
 'accentdb',
 'aeslc',
 'aflw2k3d',
 'ag_news_subset',
 'ai2_arc',
 'ai2_arc_with_ir',
 'amazon_us_reviews',
 'anli',
 'answer_equivalence',
 'arc',
 'asqa',
 'asset',
 'assin2',
 'asu_table_top_converted_externally_to_rlds',
 'austin_buds_dataset_converted_externally_to_rlds',
 'austin_sailor_dataset_converted_externally_to_rlds',
 'austin_sirius_dataset_converted_externally_to_rlds',
 'bair_robot_pushing_small',
 'bc_z',
 'bccd',
 'beans',
 'bee_dataset',
 'beir',
 'berkeley_autolab_ur5',
 'berkeley_cable_routing',
 'berkeley_fanuc_manipulation',
 'berkeley_gnm_cory_hall',
 'berkeley_gnm_recon',
 'berkeley_gnm_sac_son',
 'berkeley_mvp_converted_externally_to_rlds',
 'berkeley_rpt_converted_externally_to_rlds',
 'big_patent',
 'bigearthnet',
 'billsum',
 'binarized_mnist',
 'binary_alpha_digits',
 'ble_wind_field',
 'blimp',
 'booksum',
 'bool_q',
 'bot_adversarial_dialogue',
 'bridge',
 'bucc',
 'c4',
 'c4_wsrs',
 'caltech101',
 'caltech_birds2010',
 'caltech_birds2011',
 'cardiotox',
 'cars196',
 'cassava',
 'cats_vs_dogs',
 'celeb_a',
 'celeb_a_hq',
 'cfq',
 'cherry_blossoms',
 'chexpert',
 'cifar10',
 'cifar100',
```

```
'cifar100_n',
'cifar10_1',
'cifar10_corrupted',
'cifar10_h',
'cifar10_n',
'citrus_leaves',
'cityscapes',
'civil_comments',
'clevr',
'clic',
'clinc_oos',
'cmaterdb',
'cmu_franka_exploration_dataset_converted_externally_to_rlds',
'cmu_play_fusion',
'cmu_stretch',
'cnn_dailymail',
'coco',
'coco_captions',
'coil100',
'colorectal_histology',
'colorectal_histology_large',
'columbia_cairlab_pusht_real',
'common_voice',
'conll2002',
'conll2003',
'controlled_noisy_web_labels',
'coqa',
'corr2cause',
'cos_e',
'cosmos_qa',
'covid19',
'covid19sum',
'crema_d',
'criteo',
'cs_restaurants',
'curated_breast_imaging_ddsm',
'cycle_gan',
'd4rl_adroit_door',
'd4rl_adroit_hammer',
'd4rl_adroit_pen',
'd4rl_adroit_relocate',
'd4rl_antmaze',
'd4rl_mujoco_ant',
'd4rl_mujoco_halfcheetah',
'd4rl_mujoco_hopper',
'd4rl_mujoco_walker2d',
'dart',
'databricks_dolly',
'davis',
'deep1b',
'deep_weeds',
'definite_pronoun_resolution',
'dementiabank',
'diabetic_retinopathy_detection',
'diamonds',
'div2k',
'dlr_edan_shared_control_converted_externally_to_rlds',
'dlr_sara_grid_clamp_converted_externally_to_rlds',
'dlr_sara_pour_converted_externally_to_rlds',
'dmlab',
```

```
'doc_nli',
'dolphin_number_word',
'domainnet',
'downsampled_imagenet',
'drop',
'dsprites',
'dtd',
'duke_ultrasound',
'e2e_cleaned',
'efron_morris75',
'emnist',
'eraser_multi_rc',
'esnli',
'eth_agent_affordances',
'eurosat',
'fashion_mnist',
'flic',
'flores',
'food101',
'forest_fires',
'fractal20220817_data',
'fuss',
'gap',
'geirhos_conflict_stimuli',
'gem',
'genomics_ood',
'german_credit_numeric',
'gigaword',
'glove100_angular',
'glue',
'goemotions',
'gov_report',
'gpt3',
'gref',
'groove',
'grounded_scan',
'gsm8k',
'gtzan',
'gtzan_music_speech',
'hellaswag',
'higgs',
'hillstrom',
'horses_or_humans',
'howell',
'i_naturalist2017',
'i_naturalist2018',
'i_naturalist2021',
'iamlab_cmu_pickup_insert_converted_externally_to_rlds',
'imagenet2012',
'imagenet2012_corrupted',
'imagenet2012_fewshot',
'imagenet2012_multilabel',
'imagenet2012_real',
'imagenet2012_subset',
'imagenet_a',
'imagenet_lt',
'imagenet_pi',
'imagenet_r',
'imagenet_resized',
'imagenet_sketch',
```

```
'imagenet_v2',
'imagenette',
'imagewang',
'imdb_reviews',
'imperialcollege_sawyer_wrist_cam',
'irc_disentanglement',
'iris',
'istella',
'jaco_play',
'kaist_nonprehensile_converted_externally_to_rlds',
'kddcup99',
'kitti',
'kmnist',
'kuka',
'laion400m',
'lambada',
'lfw',
'librispeech',
'librispeech_lm',
'libritts',
'ljspeech',
'lm1b',
'locomotion',
'lost_and_found',
'lsun',
'lvis',
'malaria',
'maniskill_dataset_converted_externally_to_rlds',
'math_dataset',
'math_qa',
'mctaco',
'media_sum',
'mlqa',
'mnist',
'mnist_corrupted',
'movie_lens',
'movie_rationales',
'movielens',
'moving_mnist',
'mrqa',
'mslr_web',
'mt_opt',
'mtnt',
'multi_news',
'multi_nli',
'multi_nli_mismatch',
'natural_instructions',
'natural_questions',
'natural_questions_open',
'newsroom',
'nsynth',
'nyu_depth_v2',
'nyu_door_opening_surprising_effectiveness',
'nyu_franka_play_dataset_converted_externally_to_rlds',
'nyu_rot_dataset_converted_externally_to_rlds',
'ogbg_molpcba',
'omniglot',
'open_images_challenge2019_detection',
'open_images_v4',
'openbookqa',
```

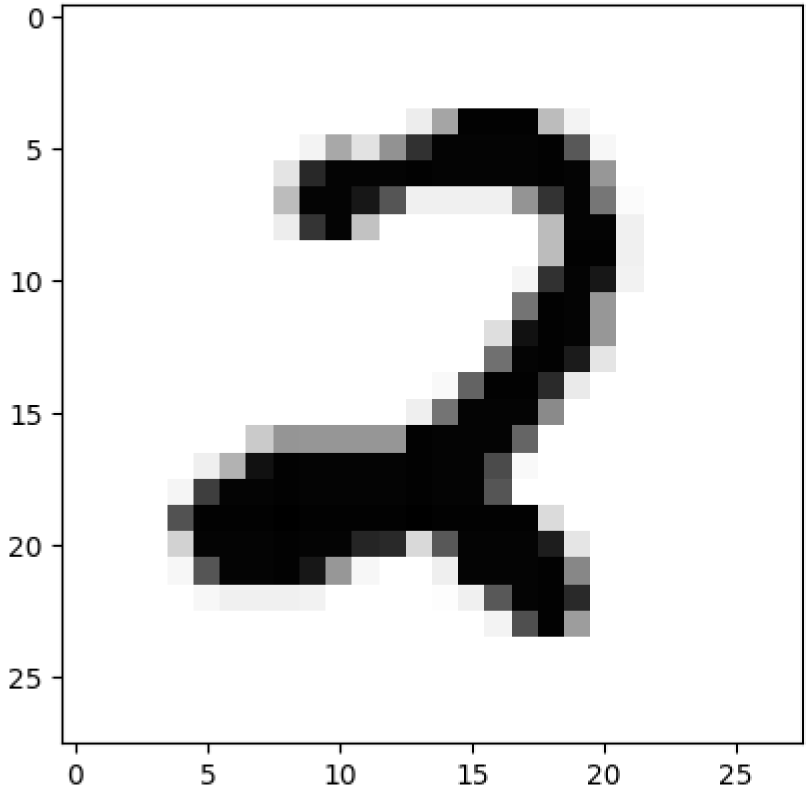
```
 'opinion_abstracts',
 'opinosis',
 'opus',
 'oxford_flowers102',
 'oxford_iiit_pet',
 'para_crawl',
 'pass',
 'patch_camelyon',
 'paws_wiki',
 'paws_x_wiki',
 'penguins',
 'pet_finder',
 'pg19',
 'piqa',
 'places365_small',
 'placesfull',
 'plant_leaves',
 'plant_village',
 'plantae_k',
 'protein_net',
 'q_re_cc',
 'qa4mre',
 'qasc',
 'quac',
 'quality',
 'quickdraw_bitmap',
 'race',
 'radon',
 'real_toxicity_prompts',
 'reddit',
 'reddit_disentanglement',
 'reddit_tifu',
 'ref_coco',
 'resisc45',
 'rlu_atari',
 'rlu_atari_checkpoints',
 'rlu_atari_checkpoints_ordered',
 'rlu_control_suite',
 'rlu_dmlab_explore_object_rewards_few',
 'rlu_dmlab_explore_object_rewards_many',
 'rlu_dmlab_rooms_select_nonmatching_object',
 'rlu_dmlab_rooms_watermaze',
 'rlu_dmlab_seekavoid_arena01',
 'rlu_locomotion',
 'rlu_rwrl',
 'robomimic_mg',
 'robomimic_mh',
 'robomimic_ph',
 'robonet',
 'robosuite_panda_pick_place_can',
 'roboturk',
 'rock_paper_scissors',
 'rock_you',
 's3o4d',
 'salient_span_wikipedia',
 'samsum',
 'savee',
 'scan',
 'scene_parse150',
 'schema_guided_dialogue',
```

```
 'sci_tail',
 'scicite',
 'scientific_papers',
 'scrolls',
 'segment_anything',
 'sentiment140',
 'shapes3d',
 'sift1m',
 'simpte',
 'siscore',
 'smallnorb',
 'smartwatch_gestures',
 'snli',
 'so2sat',
 'speech_commands',
 'spoken_digit',
 'squad',
 'squad_question_generation',
 'stanford_dogs',
 'stanford_hydra_dataset_converted_externally_to_rlds',
 'stanford_kuka_multimodal_dataset_converted_externally_to_rlds',
 'stanford_mask_vit_converted_externally_to_rlds',
 'stanford_online_products',
 'stanford_robocook_converted_externally_to_rlds',
 'star_cfq',
 'starcraft_video',
 'stl10',
 'story_cloze',
 'summscreen',
 'sun397',
 'super_glue',
 'svhn_cropped',
 'symmetric_solids',
 'taco_play',
 'tao',
 'tatoeba',
 'ted_hrlr_translate',
 'ted_multi_translate',
 'tedlium',
 'tf_flowers',
 'the300w_lp',
 'tiny_shakespeare',
 'titanic',
 'tokyo_u_lsmo_converted_externally_to_rlds',
 'toto',
 'trec',
 'trivia_qa',
 'tydi_qa',
 'uc_merced',
 'ucf101',
 'ucsd_kitchen_dataset_converted_externally_to_rlds',
 'ucsd_pick_and_place_dataset_converted_externally_to_rlds',
 'uiuc_d3field',
 'unified_qa',
 'universal_dependencies',
 'unnatural_instructions',
 'usc_cloth_sim_converted_externally_to_rlds',
 'user_libri_audio',
 'user_libri_text',
 'utaustin_mutex',
```

```
 'utokyo_pr2_opening_fridge_converted_externally_to_rlds',
 'utokyo_pr2_tabletop_manipulation_converted_externally_to_rlds',
 'utokyo_saytap_converted_externally_to_rlds',
 'utokyo_xarm_bimanual_converted_externally_to_rlds',
 'utokyo_xarm_pick_and_place_converted_externally_to_rlds',
 'vctk',
 'viola',
 'visual_domain_decathlon',
 'voc',
 'voxceleb',
 'voxforge',
 'waymo_open_dataset',
 'web_graph',
 'web_nlg',
 'web_questions',
 'webvid',
 'wider_face',
 'wiki40b',
 'wiki_auto',
 'wiki_bio',
 'wiki_dialog',
 'wiki_table_questions',
 'wiki_table_text',
 'wikiann',
 'wikihow',
 'wikipedia',
 'wikipedia_toxicity_subtypes',
 'wine_quality',
 'winogrande',
 'wit',
 'wit_kaggle',
 'wmt13_translate',
 'wmt14_translate',
 'wmt15_translate',
 'wmt16_translate',
 'wmt17_translate',
 'wmt18_translate',
 'wmt19_translate',
 'wmt_t2t_translate',
 'wmt_translate',
 'wordnet',
 'wsc273',
 'xnli',
 'xquad',
 'xsum',
 'xtreme_pawsx',
 'xtreme_pos',
 'xtreme_s',
 'xtreme_xnli',
 'yahoo_ltrc',
 'yelp_polarity_reviews',
 'yes_no',
 'youtube_vis']
```

In [5]:
```
# train_images, train_labels 데이터 확인
plt.imshow(train_images[365], cmap = 'Greys')
```

Out[5]:
```
<matplotlib.image.AxesImage at 0x1a809523100>
```



In [6]:
```
train_labels[1235]
```

Out[6]: 9

In [7]:
```
# 전처리 작업
train_images = train_images.reshape((60000, 28, 28, 1))
```

In [8]:
```
test_images = test_images.reshape((10000, 28, 28, 1))
```

In [9]:
```
train_images.shape, test_images.shape
```

Out[9]: ((60000, 28, 28, 1), (10000, 28, 28, 1))

In [10]:
```
train_images[35556]
```

Out[10]:
```
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],
 [  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],
         [ 14],
         [ 97],
         [177],
         [211],
         [130],
         [ 48],
         [  1],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0]],

        [[  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [ 10],
         [183],
         [254],
         [254],
         [254],
         [254],
         [254],
         [125],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0]],

        [[  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
```

```
        [127],
        [254],
        [252],
        [218],
        [175],
        [225],
        [254],
        [253],
        [118],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0]],

       [[  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [207],
        [254],
        [205],
        [  0],
        [  0],
        [ 14],
        [182],
        [254],
        [231],
        [ 11],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0]],

       [[  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [ 43],
        [245],
        [254],
```

```
        [134],
        [  0],
        [  0],
        [  0],
        [ 49],
        [252],
        [254],
        [112],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0]],

       [[  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [ 84],
        [254],
        [255],
        [ 92],
        [  0],
        [  0],
        [  0],
        [  0],
        [178],
        [254],
        [200],
        [ 21],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0]],

       [[  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [  0],
        [ 45],
        [245],
        [254],
        [141],
        [  0],
```

```
  [  0],
  [  0],
  [  0],
  [104],
  [254],
  [254],
  [169],
  [  2],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0]],

 [[  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [209],
  [254],
  [243],
  [ 22],
  [  0],
  [  0],
  [  0],
  [ 68],
  [239],
  [254],
  [254],
  [ 26],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0]],

 [[  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [ 86],
  [253],
  [254],
  [202],
  [ 22],
  [  6],
```

```
         [  6],
         [126],
         [245],
         [254],
         [254],
         [ 26],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0]],

        [[  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [ 99],
         [248],
         [254],
         [254],
         [254],
         [254],
         [254],
         [254],
         [254],
         [206],
         [ 10],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0]],

        [[  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [ 52],
         [180],
         [254],
         [254],
         [216],
         [171],
```

```
         [254],
         [254],
         [176],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0]],

        [[  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  0],
         [  2],
         [  7],
         [ 45],
         [  3],
         [ 35],
         [244],
         [254],
         [176],
         [  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],
         [193],
         [254],
```

```
 [176],
 [  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],
 [193],
 [254],
 [218],
 [ 14],
 [  0],
 [  0],
 [  0],
 [  0],
 [  0],
 [  0],
 [  0],
 [  0]],

[[  0],
 [  0],
 [  0],
 [  0],
 [  0],
 [  0],
 [  0],
 [  0],
 [  0],
 [  0],
 [  0],
 [  0],
 [  0],
 [  0],
 [  0],
 [  0],
 [193],
 [254],
 [254],
 [ 26],
```

```
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0]],

 [[  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [100],
  [254],
  [254],
  [ 26],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0]],

 [[  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [  0],
  [ 66],
  [251],
  [254],
  [ 53],
  [  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],
 [193],
 [254],
 [219],
 [  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],
 [135],
 [254],
 [234],
 [  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],
 [ 81],
 [225],
 [191],
 [  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]]], dtype=uint8)
```

In [11]:
```
# 0-255를 0과 1 사이의 실수로 변경
# train_images = train_images / 255.
# test_images = test_images / 255. -> 1줄로 처리
train_images, test_images = train_images / 255. , test_images / 255.
```

In [12]:
```
train_images[35556]
```

```
Out[12]: 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.        ],
         [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.        ],
         [0.05490196],
         [0.38039216],
         [0.69411765],
         [0.82745098],
         [0.50980392],
         [0.18823529],
         [0.00392157],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ]],

        [[0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.03921569],
         [0.71764706],
         [0.99607843],
         [0.99607843],
         [0.99607843],
         [0.99607843],
         [0.99607843],
         [0.49019608],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ]],

        [[0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
```

```
         [0.49803922],
         [0.99607843],
         [0.98823529],
         [0.85490196],
         [0.68627451],
         [0.88235294],
         [0.99607843],
         [0.99215686],
         [0.4627451 ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ]],

        [[0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.81176471],
         [0.99607843],
         [0.80392157],
         [0.        ],
         [0.        ],
         [0.05490196],
         [0.71372549],
         [0.99607843],
         [0.90588235],
         [0.04313725],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ]],

        [[0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.16862745],
         [0.96078431],
         [0.99607843],
```

```
        [0.5254902 ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.19215686],
        [0.98823529],
        [0.99607843],
        [0.43921569],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ]],

       [[0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.32941176],
        [0.99607843],
        [1.        ],
        [0.36078431],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.69803922],
        [0.99607843],
        [0.78431373],
        [0.08235294],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ]],

       [[0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.17647059],
        [0.96078431],
        [0.99607843],
        [0.55294118],
        [0.        ],
```

```
 [0.        ],
 [0.        ],
 [0.        ],
 [0.40784314],
 [0.99607843],
 [0.99607843],
 [0.6627451 ],
 [0.00784314],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ]],

[[0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.81960784],
 [0.99607843],
 [0.95294118],
 [0.08627451],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.26666667],
 [0.9372549 ],
 [0.99607843],
 [0.99607843],
 [0.10196078],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ]],

[[0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.3372549 ],
 [0.99215686],
 [0.99607843],
 [0.79215686],
 [0.08627451],
 [0.02352941],
```

```
         [0.02352941],
         [0.49411765],
         [0.96078431],
         [0.99607843],
         [0.99607843],
         [0.10196078],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ]],

        [[0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.38823529],
         [0.97254902],
         [0.99607843],
         [0.99607843],
         [0.99607843],
         [0.99607843],
         [0.99607843],
         [0.99607843],
         [0.99607843],
         [0.80784314],
         [0.03921569],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ]],

        [[0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.20392157],
         [0.70588235],
         [0.99607843],
         [0.99607843],
         [0.84705882],
         [0.67058824],
```

```
         [0.99607843],
         [0.99607843],
         [0.69019608],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ]],

        [[0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.        ],
         [0.00784314],
         [0.02745098],
         [0.17647059],
         [0.01176471],
         [0.1372549 ],
         [0.95686275],
         [0.99607843],
         [0.69019608],
         [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.        ],
         [0.75686275],
         [0.99607843],
```

```
        [0.69019608],
        [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.        ],
        [0.75686275],
        [0.99607843],
        [0.85490196],
        [0.05490196],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ]],

       [[0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.        ],
        [0.75686275],
        [0.99607843],
        [0.99607843],
        [0.10196078],
```

```
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ]],

[[0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.39215686],
 [0.99607843],
 [0.99607843],
 [0.10196078],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ]],

[[0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.        ],
 [0.25882353],
 [0.98431373],
 [0.99607843],
 [0.20784314],
 [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.        ],
 [0.75686275],
 [0.99607843],
 [0.85882353],
 [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.        ],
 [0.52941176],
 [0.99607843],
 [0.91764706],
 [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.        ],
        [0.31764706],
        [0.88235294],
        [0.74901961],
        [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.        ]]])
```

3. 신경망 만들기

In [13]:

```
# Keras 모델 설계
model = models.Sequential()
# 32개의 레이어, 3X3 합성곱 필터, activation function은 relu(음수는 0으로, 양수는
model.add(layers.Conv2D(32, (3,3), activation='relu', input_shape = (28, 28, 1))
model.add(layers.MaxPooling2D((2, 2))) # 절반으로
model.add(layers.Conv2D(64, (3, 3), activation = 'relu')) # CNN의 경우 3배로 늘리
model.add(layers.MaxPooling2D((2, 2))) # 절반으로
model.add(layers.Conv2D(64, (3, 3), activation = 'relu')) # CNN의 경우 3배로 늘리
model.add(layers.MaxPooling2D((2, 2))) # 절반으로
```

```
C:\Users\hi\anaconda3\envs\p310_cnn\lib\site-packages\keras\src\layers\convolutio
nal\base_conv.py:107: UserWarning: Do not pass an `input_shape`/`input_dim` argum
ent to a layer. When using Sequential models, prefer using an `Input(shape)` obje
ct as the first layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer, **kwargs)
```

In [14]:

```
model.summary()
```

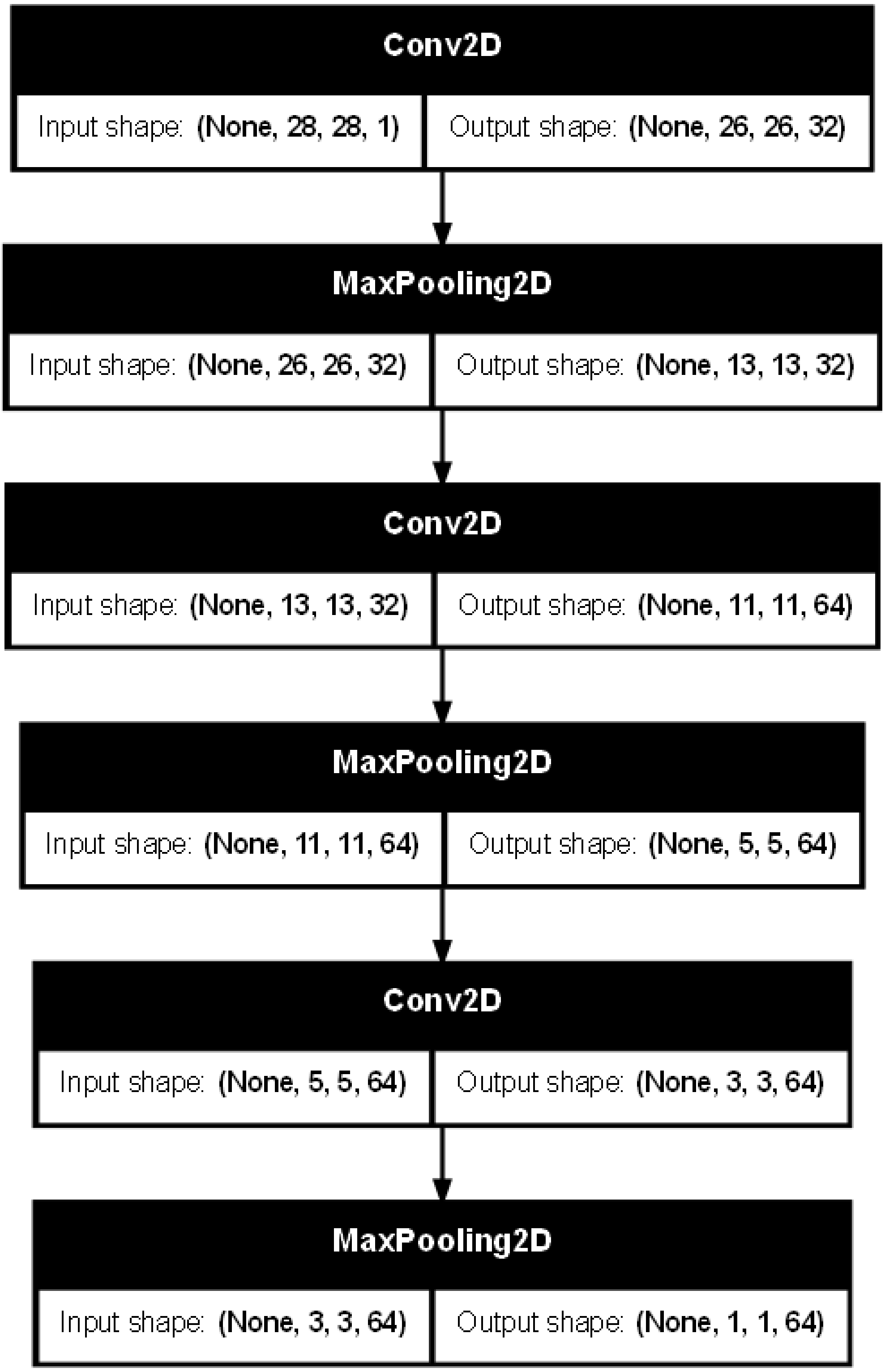
Model: "sequential"

Layer (type)	Output Shape
conv2d (Conv2D)	(None, 26, 26, 32)
max_pooling2d (MaxPooling2D)	(None, 13, 13, 32)
conv2d_1 (Conv2D)	(None, 11, 11, 64)
max_pooling2d_1 (MaxPooling2D)	(None, 5, 5, 64)
conv2d_2 (Conv2D)	(None, 3, 3, 64)
max_pooling2d_2 (MaxPooling2D)	(None, 1, 1, 64)

Total params: 55,744 (217.75 KB)
Trainable params: 55,744 (217.75 KB)
Non-trainable params: 0 (0.00 B)

In [16]:
```
plot_model(model, show_shapes=True, dpi=80)
```

Out[16]:



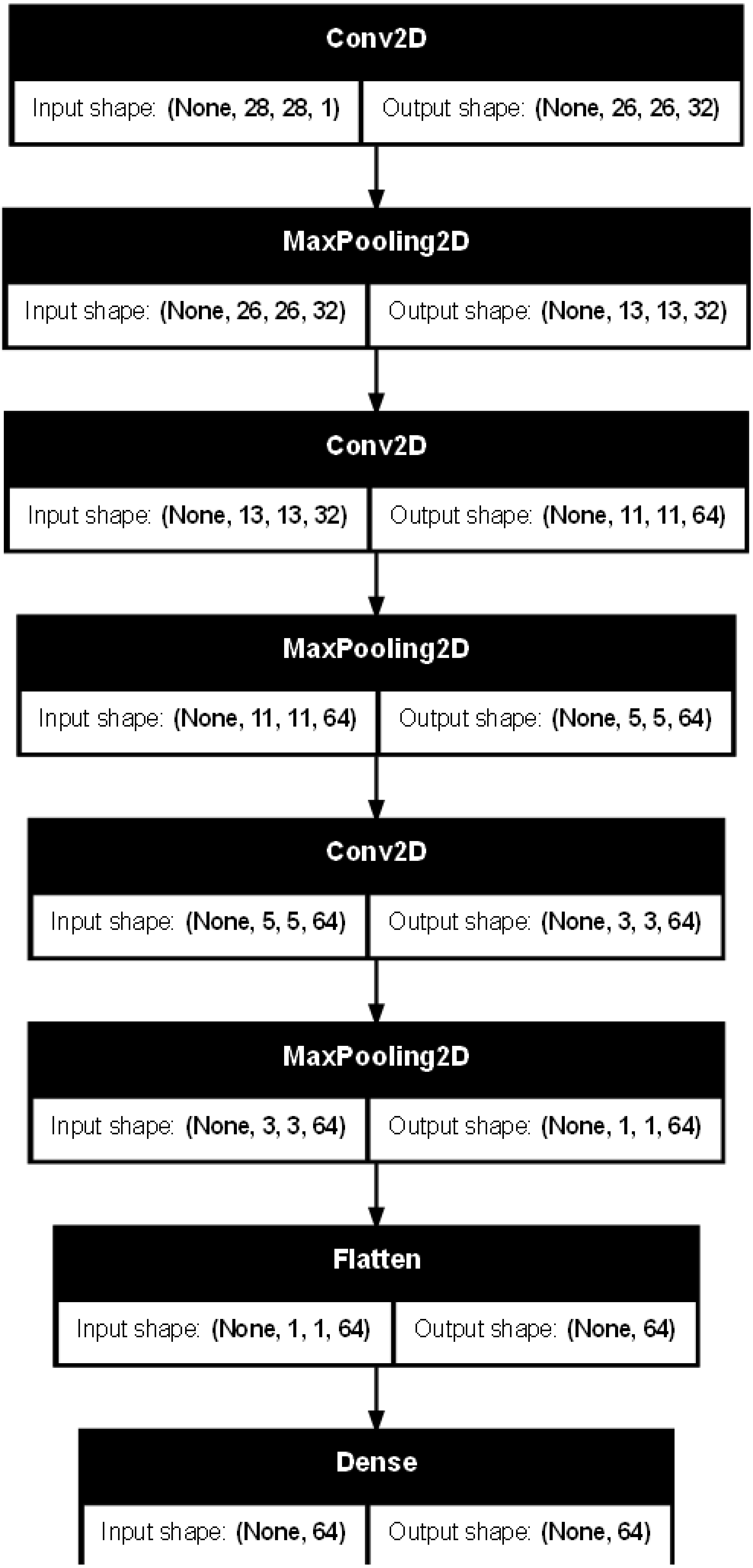
In [18]:

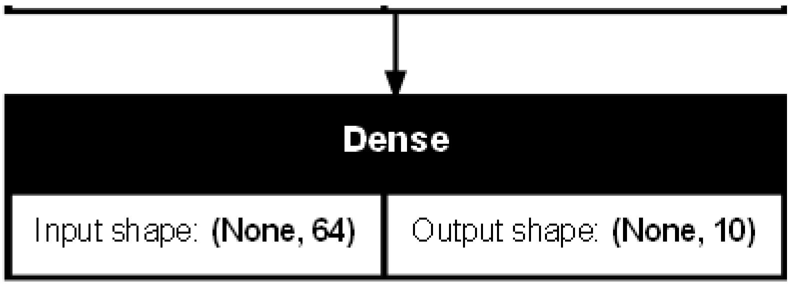
```
# 분류단계
model.add(layers.Flatten())
model.add(layers.Dense(64, activation='relu'))
model.add(layers.Dense(10, activation='softmax')) # 학습 시키0에서 9까지라서
```

In [19]:

```
plot_model(model, show_shapes=True, dpi=80)
```

Out[19]:


Conv2D
Input shape: (None, 28, 28, 1)
Output shape: (None, 26, 26, 32)
MaxPooling2D
Input shape: (None, 26, 26, 32)
Output shape: (None, 13, 13, 32)
Conv2D
Input shape: (None, 13, 13, 32)
Output shape: (None, 11, 11, 64)
MaxPooling2D
Input shape: (None, 11, 11, 64)
Output shape: (None, 5, 5, 64)
Conv2D
Input shape: (None, 5, 5, 64)
Output shape: (None, 3, 3, 64)
MaxPooling2D
Input shape: (None, 3, 3, 64)
Output shape: (None, 1, 1, 64)
Flatten
Input shape: (None, 1, 1, 64)
Output shape: (None, 64)
Dense
Input shape: (None, 64)
Output shape: (None, 64)



4. 모델 만들고 학습하기

In [21]:
```
model.compile(
    optimizer = 'adam',
    loss = 'sparse_categorical_crossentropy',
    metrics = ['accuracy']
)
```

In [22]:
```
# 모델 훈련 ->  # fit = 훈련 / verbose = 학습 상황 표시
history = model.fit(train_images, train_labels, batch_size = 16, epochs = 50,
                    verbose = 1, validation_data = (test_images, test_labels)
)
```

```
Epoch 1/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 17s 4ms/step - accuracy: 0.8772 - loss: 0.3824 - v
al_accuracy: 0.9787 - val_loss: 0.0694
Epoch 2/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 15s 4ms/step - accuracy: 0.9782 - loss: 0.0723 - v
al_accuracy: 0.9794 - val_loss: 0.0674
Epoch 3/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 15s 4ms/step - accuracy: 0.9864 - loss: 0.0472 - v
al_accuracy: 0.9870 - val_loss: 0.0442
Epoch 4/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9883 - loss: 0.0355 - v
al_accuracy: 0.9858 - val_loss: 0.0523
Epoch 5/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 17s 4ms/step - accuracy: 0.9913 - loss: 0.0297 - v
al_accuracy: 0.9865 - val_loss: 0.0533
Epoch 6/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 15s 4ms/step - accuracy: 0.9930 - loss: 0.0215 - v
al_accuracy: 0.9872 - val_loss: 0.0463
Epoch 7/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9944 - loss: 0.0179 - v
al_accuracy: 0.9877 - val_loss: 0.0549
Epoch 8/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 15s 4ms/step - accuracy: 0.9952 - loss: 0.0162 - v
al_accuracy: 0.9835 - val_loss: 0.0697
Epoch 9/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 21s 4ms/step - accuracy: 0.9950 - loss: 0.0143 - v
al_accuracy: 0.9851 - val_loss: 0.0680
Epoch 10/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9956 - loss: 0.0134 - v
al_accuracy: 0.9819 - val_loss: 0.0822
Epoch 11/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9963 - loss: 0.0128 - v
al_accuracy: 0.9843 - val_loss: 0.0745
Epoch 12/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9967 - loss: 0.0097 - v
al_accuracy: 0.9872 - val_loss: 0.0627
Epoch 13/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9963 - loss: 0.0109 - v
al_accuracy: 0.9856 - val_loss: 0.0667
Epoch 14/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 17s 4ms/step - accuracy: 0.9971 - loss: 0.0097 - v
al_accuracy: 0.9885 - val_loss: 0.0574
Epoch 15/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 17s 4ms/step - accuracy: 0.9972 - loss: 0.0085 - v
al_accuracy: 0.9891 - val_loss: 0.0583
Epoch 16/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9973 - loss: 0.0083 - v
al_accuracy: 0.9859 - val_loss: 0.0848
Epoch 17/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9974 - loss: 0.0081 - v
al_accuracy: 0.9876 - val_loss: 0.0744
Epoch 18/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 17s 4ms/step - accuracy: 0.9980 - loss: 0.0072 - v
al_accuracy: 0.9874 - val_loss: 0.0749
Epoch 19/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9979 - loss: 0.0081 - v
al_accuracy: 0.9847 - val_loss: 0.0847
Epoch 20/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 17s 4ms/step - accuracy: 0.9974 - loss: 0.0082 - v
al_accuracy: 0.9879 - val_loss: 0.0809
```

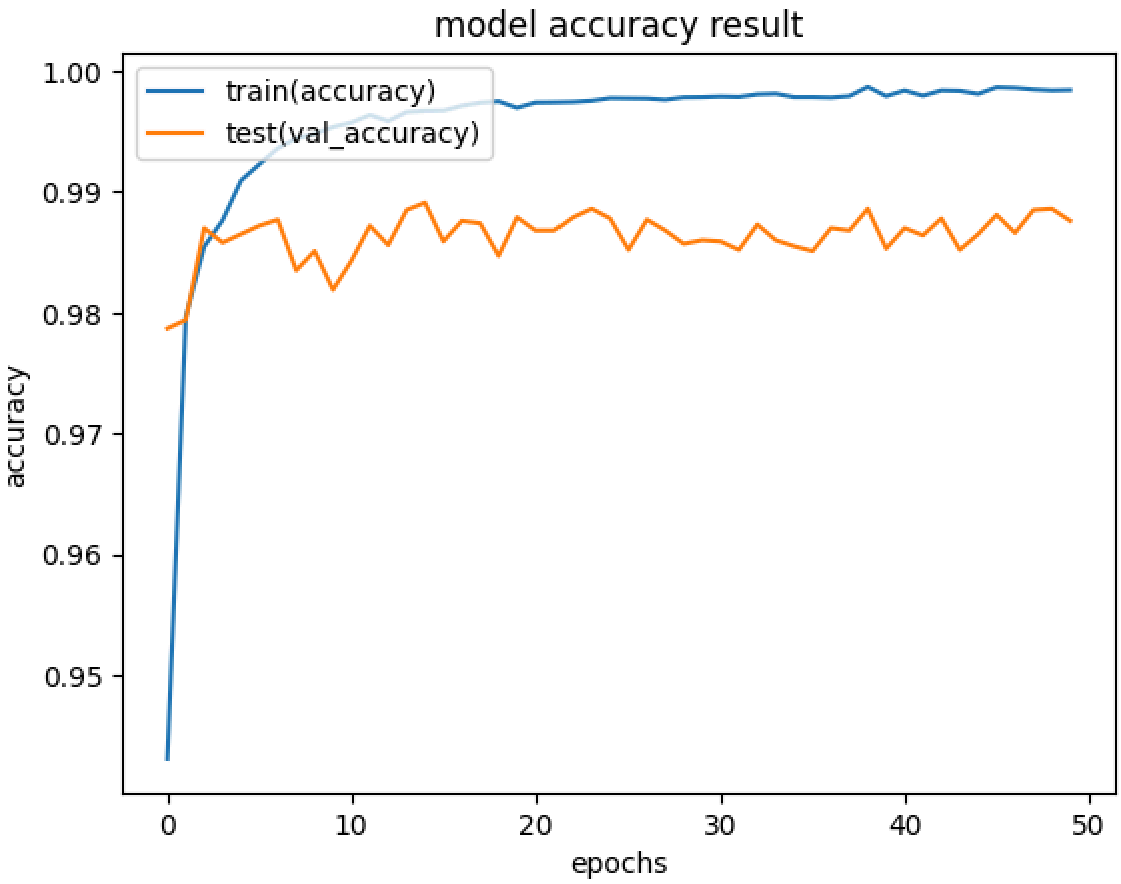
```
Epoch 21/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9974 - loss: 0.0079 - v
al_accuracy: 0.9868 - val_loss: 0.0830
Epoch 22/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 20s 4ms/step - accuracy: 0.9976 - loss: 0.0088 - v
al_accuracy: 0.9868 - val_loss: 0.0702
Epoch 23/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9974 - loss: 0.0080 - v
al_accuracy: 0.9879 - val_loss: 0.0725
Epoch 24/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9977 - loss: 0.0084 - v
al_accuracy: 0.9886 - val_loss: 0.0859
Epoch 25/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9984 - loss: 0.0050 - v
al_accuracy: 0.9878 - val_loss: 0.0895
Epoch 26/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9980 - loss: 0.0062 - v
al_accuracy: 0.9852 - val_loss: 0.1030
Epoch 27/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9976 - loss: 0.0078 - v
al_accuracy: 0.9877 - val_loss: 0.0916
Epoch 28/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 15s 4ms/step - accuracy: 0.9979 - loss: 0.0074 - v
al_accuracy: 0.9868 - val_loss: 0.0843
Epoch 29/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9978 - loss: 0.0073 - v
al_accuracy: 0.9857 - val_loss: 0.1017
Epoch 30/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 17s 4ms/step - accuracy: 0.9977 - loss: 0.0070 - v
al_accuracy: 0.9860 - val_loss: 0.1050
Epoch 31/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9986 - loss: 0.0056 - v
al_accuracy: 0.9859 - val_loss: 0.1061
Epoch 32/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 17s 4ms/step - accuracy: 0.9982 - loss: 0.0061 - v
al_accuracy: 0.9852 - val_loss: 0.1165
Epoch 33/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 17s 4ms/step - accuracy: 0.9979 - loss: 0.0078 - v
al_accuracy: 0.9873 - val_loss: 0.1020
Epoch 34/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 17s 4ms/step - accuracy: 0.9988 - loss: 0.0049 - v
al_accuracy: 0.9860 - val_loss: 0.1180
Epoch 35/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9979 - loss: 0.0081 - v
al_accuracy: 0.9855 - val_loss: 0.1076
Epoch 36/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9978 - loss: 0.0093 - v
al_accuracy: 0.9851 - val_loss: 0.1460
Epoch 37/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 17s 4ms/step - accuracy: 0.9975 - loss: 0.0101 - v
al_accuracy: 0.9870 - val_loss: 0.1322
Epoch 38/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 17s 5ms/step - accuracy: 0.9980 - loss: 0.0107 - v
al_accuracy: 0.9868 - val_loss: 0.1117
Epoch 39/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9989 - loss: 0.0051 - v
al_accuracy: 0.9886 - val_loss: 0.1065
Epoch 40/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9983 - loss: 0.0070 - v
al_accuracy: 0.9853 - val_loss: 0.1352
```

```
Epoch 41/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 15s 4ms/step - accuracy: 0.9983 - loss: 0.0067 - v
al_accuracy: 0.9870 - val_loss: 0.1248
Epoch 42/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9987 - loss: 0.0052 - v
al_accuracy: 0.9864 - val_loss: 0.1252
Epoch 43/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9988 - loss: 0.0051 - v
al_accuracy: 0.9878 - val_loss: 0.1090
Epoch 44/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9986 - loss: 0.0043 - v
al_accuracy: 0.9852 - val_loss: 0.1398
Epoch 45/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9984 - loss: 0.0072 - v
al_accuracy: 0.9865 - val_loss: 0.1286
Epoch 46/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9988 - loss: 0.0064 - v
al_accuracy: 0.9881 - val_loss: 0.1124
Epoch 47/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9985 - loss: 0.0056 - v
al_accuracy: 0.9866 - val_loss: 0.1288
Epoch 48/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9989 - loss: 0.0055 - v
al_accuracy: 0.9885 - val_loss: 0.1159
Epoch 49/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 17s 4ms/step - accuracy: 0.9985 - loss: 0.0059 - v
al_accuracy: 0.9886 - val_loss: 0.1218
Epoch 50/50
3750/3750 ━━━━━━━━━━━━━━━━━━━━ 16s 4ms/step - accuracy: 0.9987 - loss: 0.0053 - v
al_accuracy: 0.9876 - val_loss: 0.1330
```

In [27]:
```
# 훈련결과 정보 확인
history.history.keys()
```

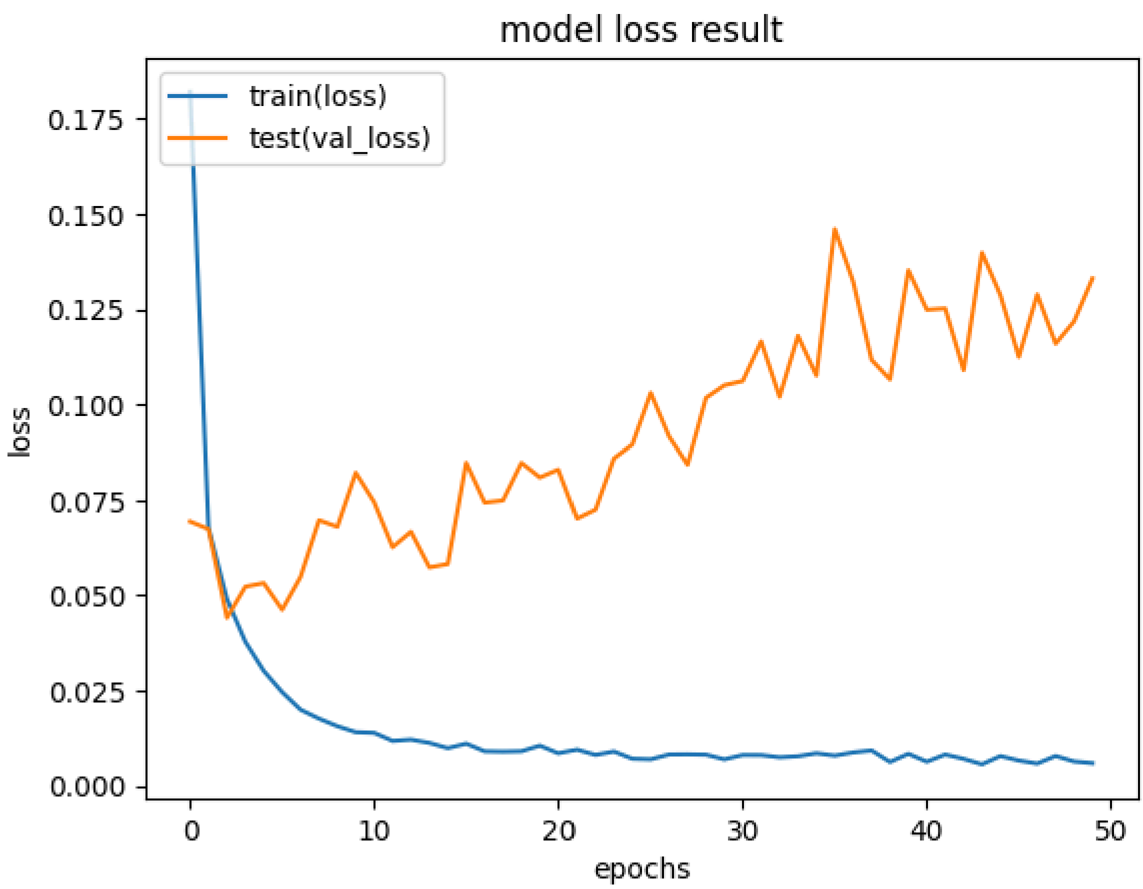
Out[27]:
```
dict_keys(['accuracy', 'loss', 'val_accuracy', 'val_loss'])
```

In [31]:
```
# 훈련결과 그래프 표시
plt.plot(history.history['accuracy'])
plt.plot(history.history['val_accuracy'])
plt.title('model accuracy result')
plt.ylabel('accuracy') # y축
plt.xlabel('epochs') # x축
plt.legend(['train(accuracy)', 'test(val_accuracy)'], loc = 'upper left')
# plt.legend(['accuracy', 'val_accuracy'], loc = 'upper left')
plt.show()
```



In [32]:

```
# 훈련결과 그래프 표시
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('model loss result')
plt.ylabel('loss') # y축
plt.xlabel('epochs') # x축
plt.legend(['train(loss)', 'test(val_loss)'], loc = 'upper left')
# plt.legend(['accuracy', 'val_accuracy'], loc = 'upper left')
plt.show()
```



5. 모델 적용하기(예측 : Predict)

In [33]:
```
# 모델 평가 - 문제와 답이 필요함
loss, acc = model.evaluate(test_images, test_labels, verbose=2) # verbose -> 1 :
```

```
313/313 - 1s - 3ms/step - accuracy: 0.9876 - loss: 0.1330
```

In [38]:
```
acc # 98.76%
```

Out[38]:
```
0.9876000285148621
```

In [40]:
```
# 모델 예측 - 문제만 있으면 됨
result = model.predict(test_images) # 10,000개
```

```
313/313 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step
```

In [43]:
```
result[5000]
```

Out[43]:
```
array([1.38281875e-29, 1.81292580e-19, 3.99213954e-22, 1.00000000e+00,
       7.19171071e-34, 1.32578095e-15, 5.60544591e-25, 1.27679835e-20,
       1.65464339e-16, 1.73610397e-30], dtype=float32)
```

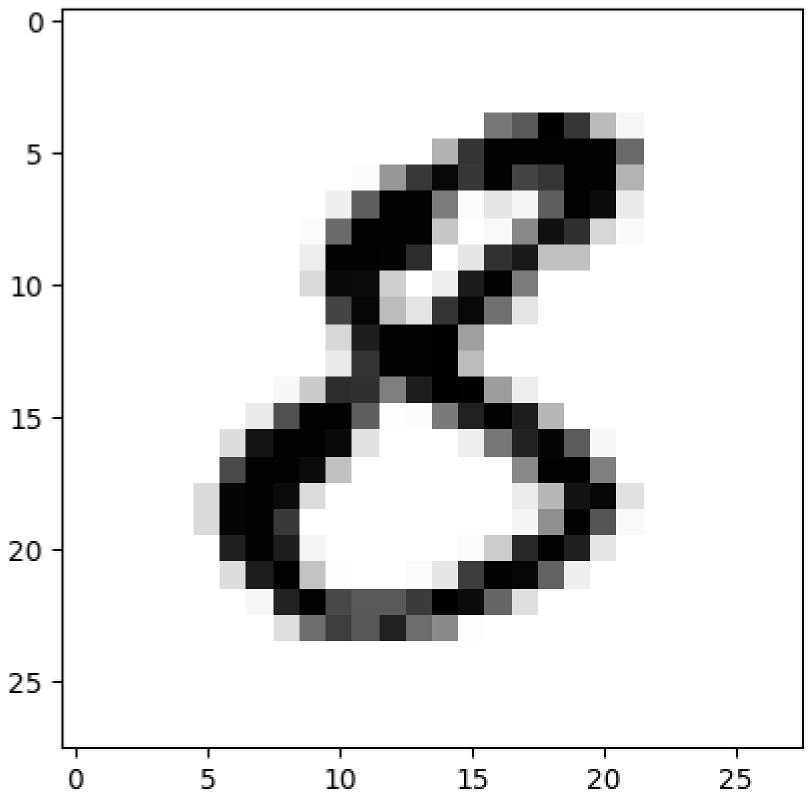
In [48]:
```
result_value = np.argmax(result[1234])
```

In [49]:
```
print("예측 결과값 = %d" % (result_value))
```

```
예측 결과값 = 8
```

In [50]:
```
plt.imshow(test_images[1234], cmap = 'Greys')
```

Out[50]: <matplotlib.image.AxesImage at 0x1a8329d29e0>



In [ ]:
```
# 실제로 사용할 때 모델 저장부타 하면 됨 -> 학습은 이미 끝났음
```

In [51]:
```
# 모델 저장
model.save('mnist_cnn_20240517_epochs_50_9876.h5') # FileName : dqte, epochs, ac
```

```
WARNING:absl:You are saving your model as an HDF5 file via `model.save()` or `ker
as.saving.save_model(model)`. This file format is considered legacy. We recommend
using instead the native Keras format, e.g. `model.save('my_model.keras')` or `ke
ras.saving.save_model(model, 'my_model.keras')`.
```

6. 실제 데이터 파일 검증

In [53]:
```
# 이미지 불러오기
image = Image.open('6.jpg')
```

In [55]:
```
# 이미지 흑백으로
image = image.convert('L') # 대문자 L -> 컬러에서 흑백으로
```

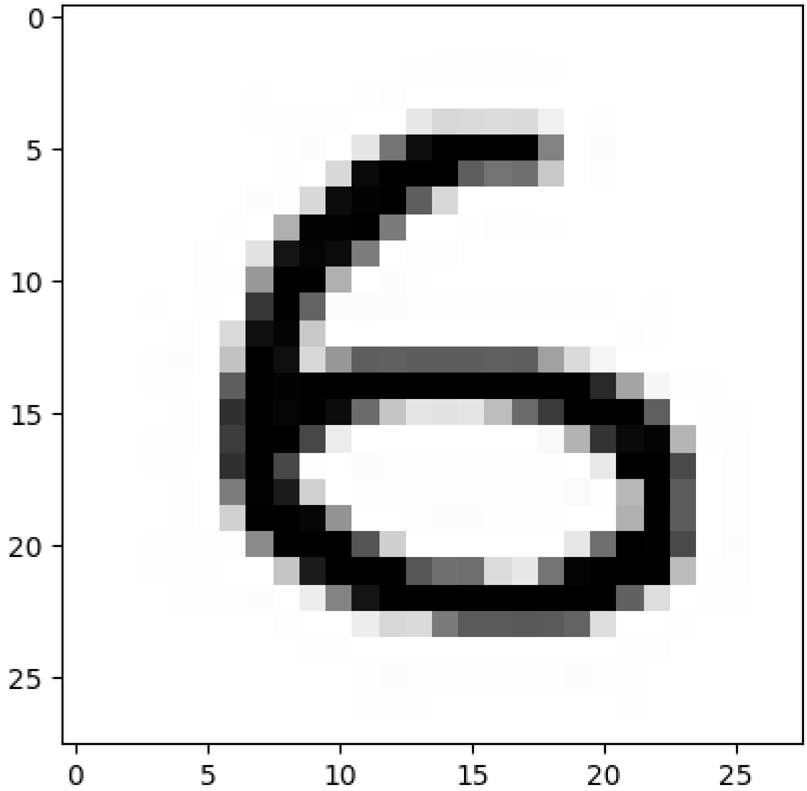
In [58]:
```
# 인공지능에 들어갈 수 있는 형태로 변환
data = np.ndarray(shape=(1, 28, 28, 1), dtype = np.float32)
size = (28, 28)
image = ImageOps.fit(image, size, Image.LANCZOS) # 사이즈 맞춤 : 이미지, 사이즈,

image_array = np.array(image)
# 전처리
nor_image_array = (image_array.astype(np.float32) / 255.) # or 127.) -1
data = nor_image_array
img_data = data.reshape(1, 28, 28, 1)
img_data.shape
```

Out[58]: (1, 28, 28, 1)

In [59]:
```
plt.imshow(image, cmap = 'Greys')
```

Out[59]: <matplotlib.image.AxesImage at 0x1a832bf6d70>



In [60]:
```
# 예측
result = model.predict(img_data)
```

```
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 17ms/step
```

In [61]:
```
result
```

Out[61]:
```
array([[3.5319815e-17, 8.4430869e-35, 0.0000000e+00, 0.0000000e+00,
        2.2788079e-26, 2.2093388e-13, 1.0000000e+00, 0.0000000e+00,
        1.2352133e-21, 1.5133417e-26]], dtype=float32)
```

In [62]:
```
rev_value = np.argmax(result)
```

In [63]:
```
print("실제 이미지 예측 결과 = %d" % (rev_value))
```

```
실제 이미지 예측 결과 = 6
```