Answer the Questions:

You can either complete the code in the Jupyter Notebook first, and then answer the homework questions here in the course shell (recommended);  
Or develop your code in the Notebook and answer the questions here at the same time

No matter which approach, please **read very carefully** every **description and requirement** **in the juypter Notebook** descriptions & code comments, **as well as in the question** asked here in the course shell.

Submission:

You can download the Jupyter Notebook base file file [DSCI471\_HW2.ipynb here](https://learn.dcollege.net/bbcswebdav/pid-12969402-dt-content-rid-286425468_1/xid-286425468_1), and then click "File" > "Save as" in the menu after opening it with your Jupyter Notebook, and give it a new name "**NAME**" to continue develop your code upon it.

**QUESTION 1**

1. For **##-Q1** in the Jupyter Notebook DSCI471\_HW2.ipynb:

What line of code you would develop to fulfill the requirement in the code comments "## import the integrated keras from tensorflow platform"?

Copy your executed error-free code line to fill in the blank here.

\* Be careful for all the blanks in this homework:  
\* for example not to include any extra spaces in front or in the end of your code line when you fill it in the blank here.  
\* reminder: python is case sensitive, so is your answer here to fill in the blank.



**QUESTION 2**

1. For **##-Q2**:

What is the name of NumPy attribute that you can use on both of the two NumPy arrays **images\_train**and **images\_test**to get their dimension?

images\_train\_dimension = images\_train.shape\_\_\_\_\_

images\_test\_dimension = images\_test.shape\_\_\_\_\_



### QUESTION 3

1. for **##-Q3**:

The numbers of the available images in the dataset loaded from Keras in previous code,

for train dataset is: 

for test dataset is: 

\* fill the pure number value without comma or spaces.

**QUESTION 4**

1. For **##-Q4**:

What is the dimension of an image from the MNIST dataset?

For example if an image has 800 pixel as its height and 600 pixel as its width, then its dimension can be expressed as:

800X600

Please fill in the blank with your answer to the same pattern as above (i.e. no space involved anywhere in your answer and use a capitalized letter X to represent the cross mark.



### QUESTION 5

1. for **##-Q5**:

The value range of all image pixels in the dataset is [ ,  ]

\* fill the pure integer number value without comma or spaces.

**QUESTION 6**

1. For **##-Q6**:

From the code result of the label and its frequency count, how many unique value of labels for the images in the MNIST dataset?  \_\_\_10\_\_\_

Please fill in the blank a pure number value without any comma or space.



**QUESTION 7**

1. For **##-Q7**:

The index of the 6th image to retrieve from the train dataset for displaying is  \_\_5\_\_\_

Please fill in the blank a pure number value without any comma or space.



**QUESTION 8**

1. For **##-Q8**:

The number turned out from both image display and the provided label associated with the the 6th image in the train dataset is \_2\_

Please fill in the blank a pure number value without any comma or space.

**QUESTION 9**

1. For **##-Q9**:

Based on the range of pixel values of the images we explored in previous code for ##-Q5, what is the value you should fill in the blank to have following code to rescale the range to [0, 1]?

image = image / \_255\_

Please fill in the blank a pure number value without any comma or space.



**QUESTION 10**

1. For **##-Q10**:

What is the value to fill in the code below that reshape to explicitly express the color channel for grayscale images in the MNIST dataset?

images = images.reshape(images\_dimension[0], images\_dimension[1], images\_dimension[2], \_\_\_\_1\_\_\_ )

Please fill in the blank a pure number value without any comma or space.



### QUESTION 11

1. for **##-Q11**:

What are the values you should fill in the blank for the below code to fulfill the requirements of configuring a Convolutional layer that can:

receive image **input** with tensors of shape (**image\_height**, **image\_width**, **color\_channels**)  for the grayscale image of MNIST dataset,  
has **32 filters**,  
**5X5**as the **kernel size**,  
**relu** as the **activation function,** and  
**padding with zeros evenly**to the surroundings of the input.

layers.Conv2D(  , (,  ), input\_shape=( ,  ,  ), activation='', padding=' ') ,

\* fill the pure **number value**and/or **string characters** without comma or spaces.

\* refer to the clickable link in section description of the Notebook above the code cell for more on the **parameter**syntax from the **API documentaions**

**QUESTION 12**

1. For **##-Q12**:

What is the value you should put in both of the blanks below to fulfill the requirements of configuring a Max Pooling layer that has pooling window size as 2 and stride step as 2? 

layers.MaxPool2D(pool\_size=2 , strides=2 ),

Please fill in the blank a pure number value without any comma or space.



**QUESTION 13**

1. For **##-Q13**:

What is the number of the output of the Flatten layer, as the outputs from previous layer are flattened into a vector of one dimension, in order to fully connect the subsequent layer?

\* Please fill in the blank a pure number value without any comma or space.



**QUESTION 14**

1. For **##-Q14**:

What is the number of the output of the below densely connectly layer ?

    layers.Dense(128, activation='tanh'),

\* Please fill in the blank a pure number value without any comma or space.



**QUESTION 15**

1. For **##-Q16**:

What is total number of ALL parameters in your model build that need to be learned during the model training?

\* Please fill in the blank a pure number value without any comma or space.



**QUESTION 16**

1. For **##-Q17**:

What is the function name to fill in the blank \_\_\_ in the below code to implement an optimizer of the Adam algorithm ?

    my\_optimizer = optimizers.Adam(learning\_rate=0.0001)

\* Please fill in the blank a string of characters without any comma or space.

\* Reminder: case sensitive answer



### QUESTION 17

1. For **##-Q18**:

What is the function name to fill in the blank \_\_\_\_ in the below code to compile your builded model ?

      model.compile(  
          optimizer=my\_optimizer,  
          loss=keras.losses. sparse\_categorical\_crossentropy(),  
          metrics=['accuracy', 'mse']  
      )

\* Please fill in the blank a string of characters without any comma or space.

### QUESTION 18

1. for **##-Q19**:

What is the function name for sparse categorical crossentropy as loss and one of the metric names for accuracy that you can fill in the blanks \_\_\_\_\_ in the below code ?

      model.compile(  
          optimizer=my\_optimizer,  
          loss=keras.losses. (),  
          metrics=['', 'mse']  
      )

\* Please fill in the blank with case-sensitive string of characters without any comma or space.   The blank for **metric**is expecting **8 characters** (although its equalent varient that has only 3 characters also works).

\* refer to the clickable link in section description of the Notebook above the code cell for more on the **parameter**syntax from the **API documentaions**

### QUESTION 19

1. For **##-Q20**:

What is value you should assign to the variable of sample\_count  
in order to randomly sample 20,000 images and labels form the **original train dataset**  
by using the our previously defined function preprocess\_dataset?

    train\_sample\_count = 20000

\* Please fill in the blank a pure number value without any comma or space.

**QUESTION 20**

1. For **##-Q21**:

What is the function name to fill in the blank \_\_\_\_\_ in the below code to feed the train dataset into the model for training?

    model.fit(dataset\_train, epochs=epochs\_count)

\* Please fill in the blank a string of characters without any comma or space.

\* Reminder, **you need to also tuning your model** in the Notebook by**adjusting the values of hyperparameters such as learning rate, batch size, and epochs** to reach an **accuracy more than 97%**



**QUESTION 21**

1. For **##-Q22**:

What is value you should assign to the variable of test\_sample\_count  
in order to obtain**ALLimages** and labels available form the **original test dataset**  
by using the our previously defined function preprocess\_dataset?

    test\_sample\_count = 10000

\* Please fill in the blank a pure number value without any comma or space.



**QUESTION 22**

1. For **##-Q23**:

What is the function name to fill in the blank \_\_\_\_\_ in the below code to evaluate the model after training using test dataset?

    model.evaluate(dataset\_test)

\* Please fill in the blank a string of characters without any comma or space.



**QUESTION 23**

1. For **##-Q24**:

The index value to assign in the blank of below code to pick the 2nd image from the original test images for prediction by trained model is ?

    idx\_img\_predict =1

Please fill in the blank a pure number value without any comma or space.



**QUESTION 24**

1. For **##-Q25**:

What is the function name to fill in the blank \_\_\_\_\_ in the below code to predict the 2nd image from the original test images using the trained model?

    prediction = model.predict(image\_predict)

\* Please fill in the blank a string of characters without any comma or space.



**QUESTION 25**

1. For **##-Q26**:

What is the function name to fill in the blank \_\_\_\_\_ in the below code to interpret the predicted results from the Softmax actication function in the trained model ?

    print("The predicted digit number from this image is: \n", np.argmax(prediction))

\* Please fill in the blank a string of characters without any comma or space.  You can refer to the hint and clickable link function documents



**QUESTION 26**

1. What is the predicted digit \_\_\_\_\_ for the 2nd image from the original test images?

Also, check if Does it match to the visual and label comparison of that image.



**QUESTION 27**

1. **Copy**your final settings of the **hyperparameters**in the code after tunning, for example:

Tuned Hyperparameters:

learning\_rate=0.0001

train\_batch\_size = 32

epochs\_count = 10

And also **copy**all the ouput **verbose** information during the **training**and **testing** procedure, for example:

Training verbose:

Epoch 1/???  
???/??? [==============================] - 29s 184ms/step - loss: 0.0976 - accuracy: 0.8723 - mse: 27.3944  
Epoch 2/???  
???/??? [==============================] - 3.........  
.........  
.........  
.........  
Epoch ???/???  
???/??? [==============================] - 37s 185ms/step - loss: 0.0589 - accuracy: 0.???? - mse: 27.3969

Testing verbose:

???/??? [==============================] - 0s 2ms/step - loss: 0.0569 - accuracy: 0.???? - mse: 27.3362

🡺.. Training verbose:

Epoch 1/10

625/625 [==============================] - 28s 43ms/step - loss: 0.5888 - accuracy: 0.8494 - mse: 27.3650

Epoch 2/10

625/625 [==============================] - 21s 31ms/step - loss: 0.1811 - accuracy: 0.9484 - mse: 27.3888

Epoch 3/10

625/625 [==============================] - 22s 34ms/step - loss: 0.1227 - accuracy: 0.9653 - mse: 27.3929

Epoch 4/10

625/625 [==============================] - 20s 31ms/step - loss: 0.0934 - accuracy: 0.9726 - mse: 27.3948

Epoch 5/10

625/625 [==============================] - 20s 31ms/step - loss: 0.0751 - accuracy: 0.9789 - mse: 27.3959

Epoch 6/10

625/625 [==============================] - 21s 32ms/step - loss: 0.0631 - accuracy: 0.9814 - mse: 27.3967

Epoch 7/10

625/625 [==============================] - 20s 32ms/step - loss: 0.0538 - accuracy: 0.9846 - mse: 27.3973

Epoch 8/10

625/625 [==============================] - 20s 31ms/step - loss: 0.0461 - accuracy: 0.9866 - mse: 27.3978

Epoch 9/10

625/625 [==============================] - 21s 32ms/step - loss: 0.0401 - accuracy: 0.9888 - mse: 27.3982

Epoch 10/10

625/625 [==============================] - 20s 31ms/step - loss: 0.0339 - accuracy: 0.9910 - mse: 27.3985

Testing verbose:

1/1 [==============================] - 4s 4s/step - loss: 0.0497 - accuracy: 0.9843 - mse: 27.3372

### QUESTION 28

1. **Submit here**

your **completed** and **executed**Notebook (**with all the outputs** remains) where the model trained to reach the required **accuracy**of more than**97%**, and

has the Notebook named as **DSCI471\_HW2\_yourDrexelID.ipynb**