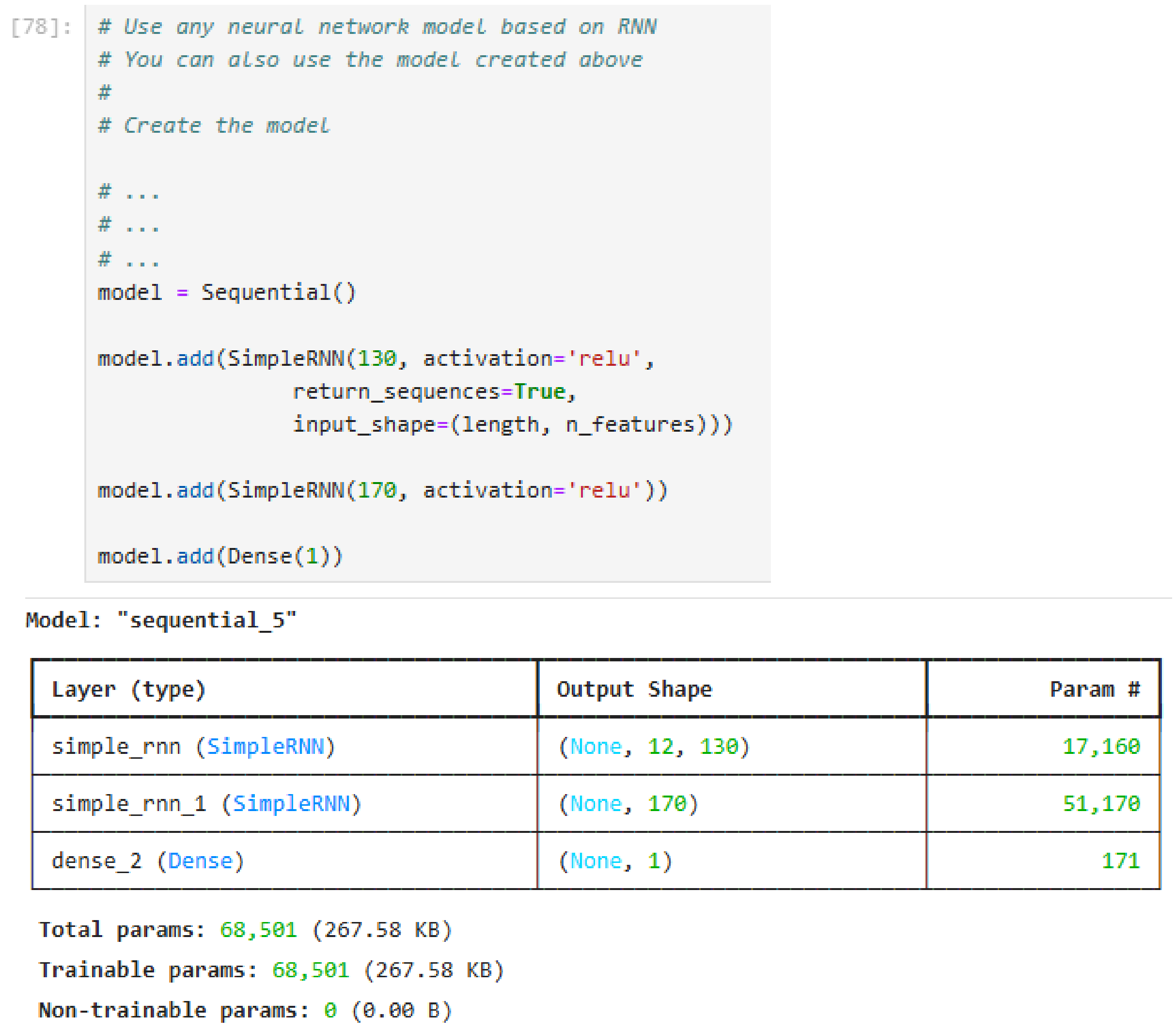
WEEK -6

LOB LOGBOOK

NAME - LIKHITHA MAHADEVA

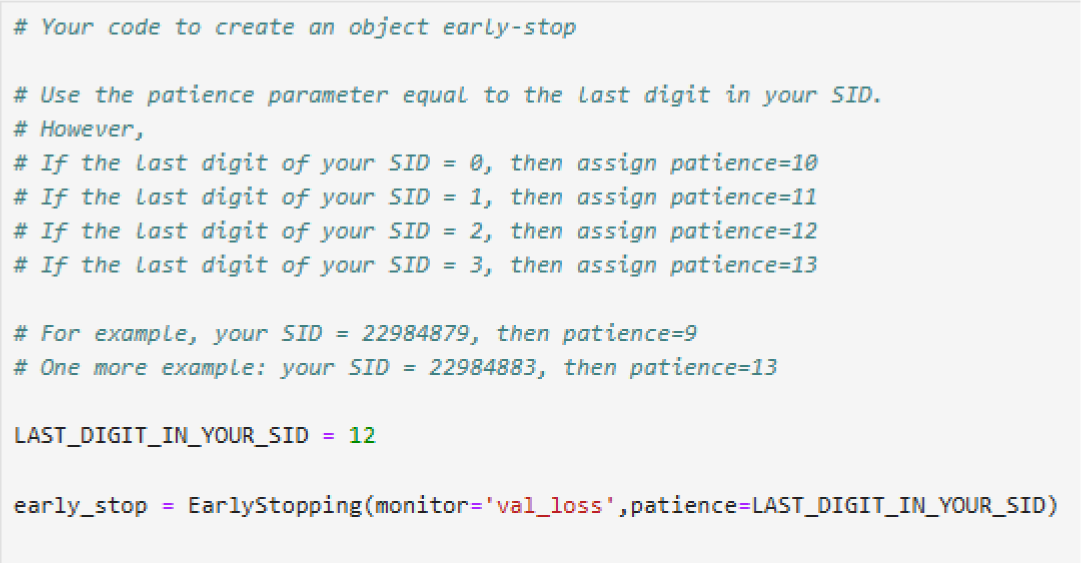
SID NUMBER - 2458682

ASSIGNMENT - WEEK 6

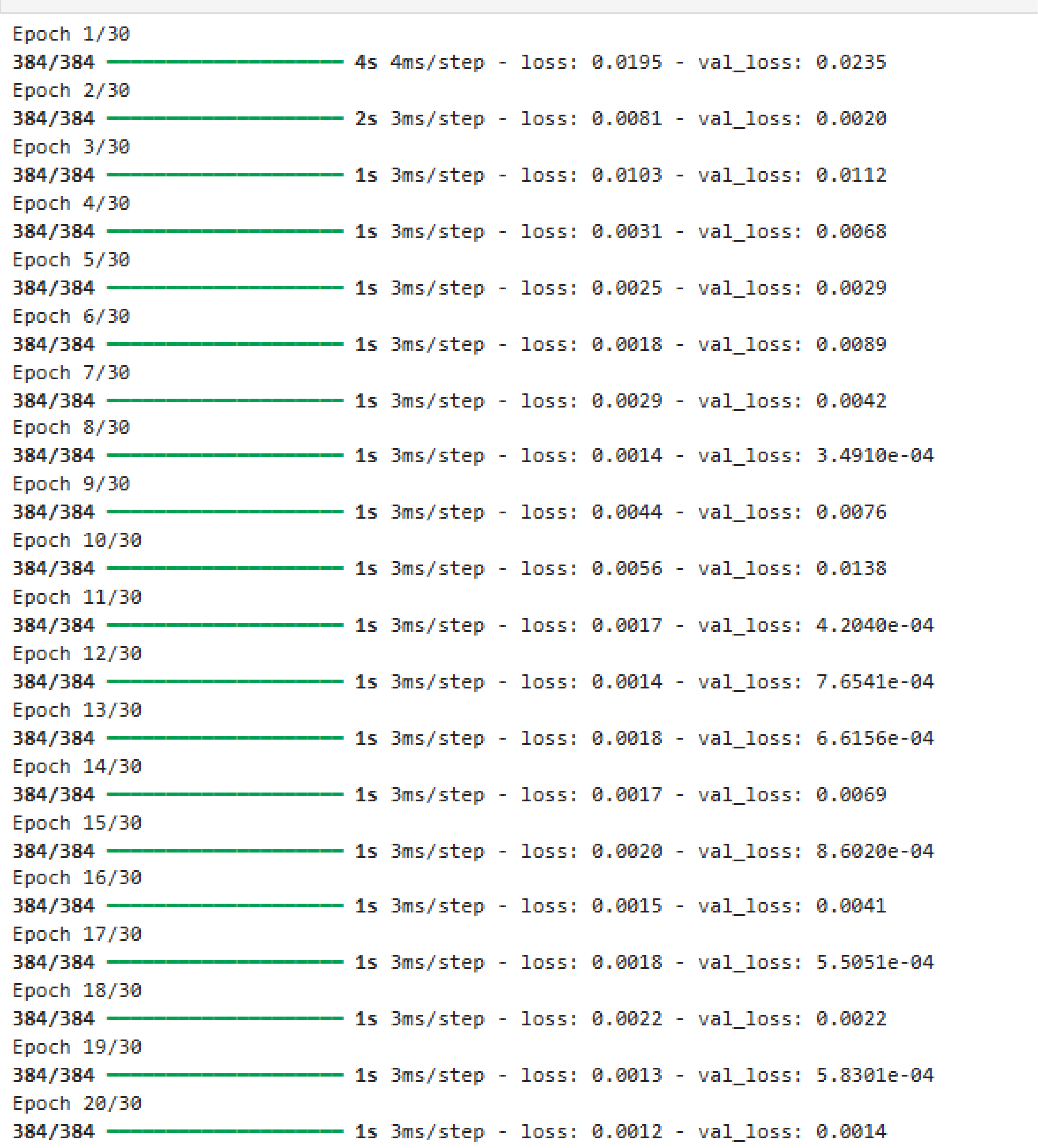


```
[78]: # Use any neural network model based on RNN
      # You can also use the model created above
      #
      # Create the model

      # ...
      # ...
      # ...
      model = Sequential()

      model.add(SimpleRNN(130, activation='relu',
                     return_sequences=True,
                     input_shape=(length, n_features)))

      model.add(SimpleRNN(170, activation='relu'))

      model.add(Dense(1))
```

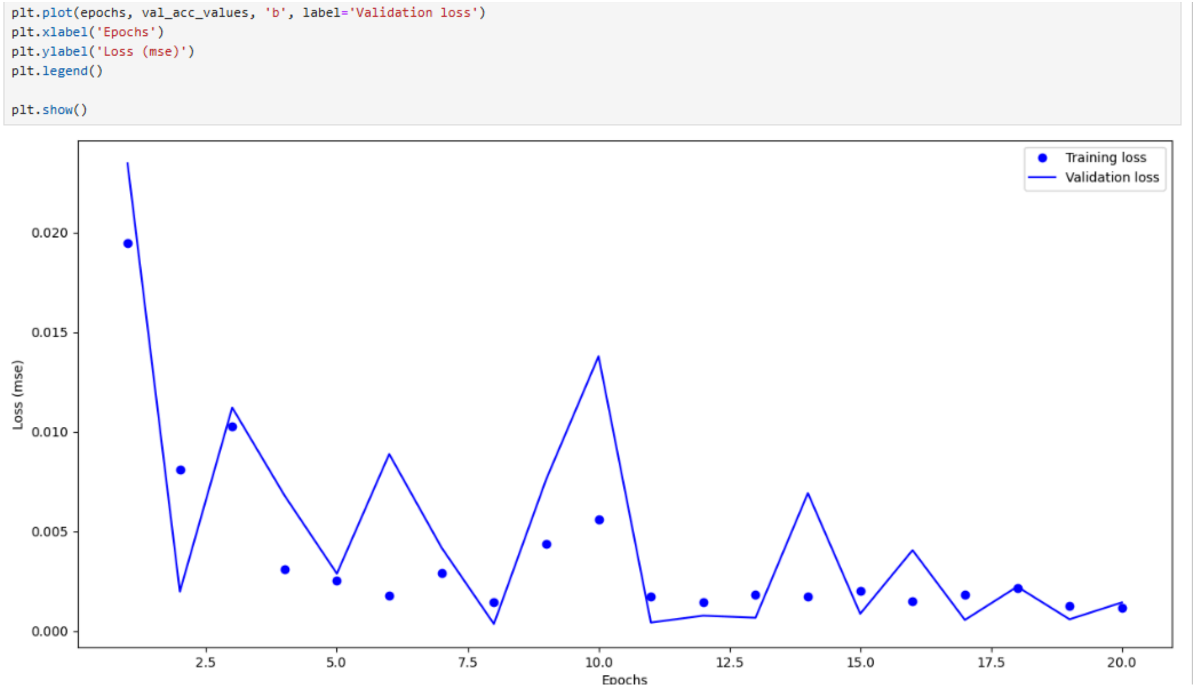
```
Model: "sequential_5"

┏━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┳━━━━━━━━━━━━━━━━━━━━━━━━┳━━━━━━━━━━━━━━━┓
┃ Layer (type)                  ┃ Output Shape           ┃       Param # ┃
┡━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━╇━━━━━━━━━━━━━━━━━━━━━━━━╇━━━━━━━━━━━━━━━┩
│ simple_rnn (SimpleRNN)        │ (None, 12, 130)        │        17,160 │
├───────────────────────────────┼────────────────────────┼───────────────┤
│ simple_rnn_1 (SimpleRNN)      │ (None, 170)            │        51,170 │
├───────────────────────────────┼────────────────────────┼───────────────┤
│ dense_2 (Dense)               │ (None, 1)              │           171 │
└───────────────────────────────┴────────────────────────┴───────────────┘

 Total params: 68,501 (267.58 KB)
 Trainable params: 68,501 (267.58 KB)
 Non-trainable params: 0 (0.00 B)
```



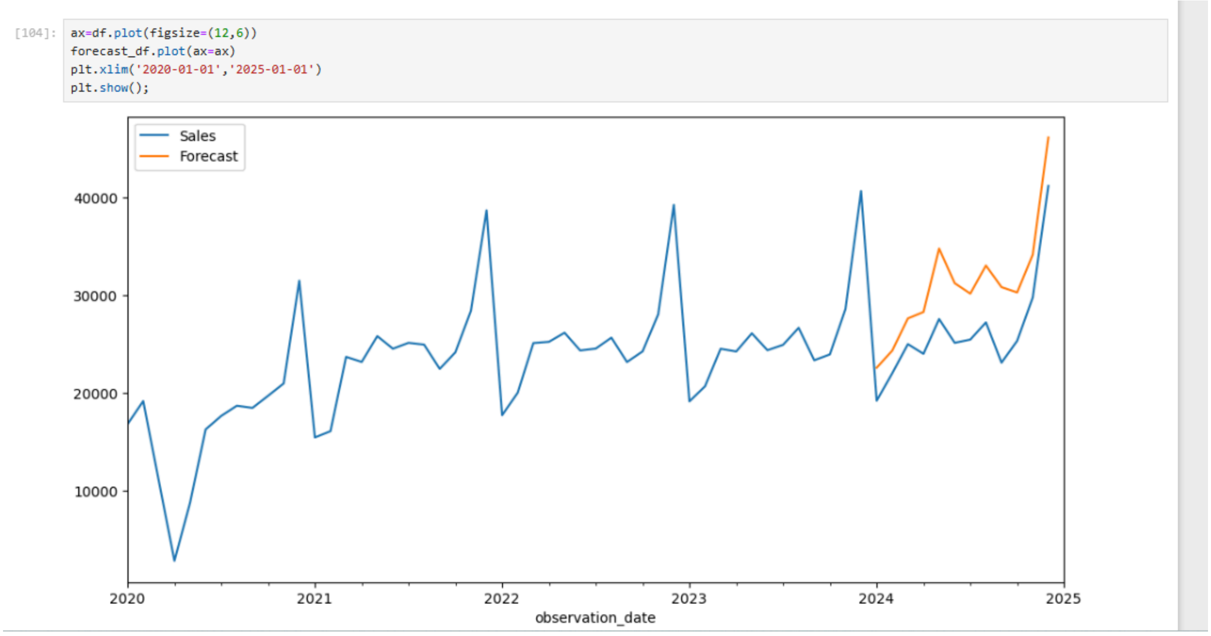
```python
# Your code to create an object early-stop

# Use the patience parameter equal to the last digit in your SID.
# However,
# If the last digit of your SID = 0, then assign patience=10
# If the last digit of your SID = 1, then assign patience=11
# If the last digit of your SID = 2, then assign patience=12
# If the last digit of your SID = 3, then assign patience=13

# For example, your SID = 22984879, then patience=9
# One more example: your SID = 22984883, then patience=13

LAST_DIGIT_IN_YOUR_SID = 12

early_stop = EarlyStopping(monitor='val_loss',patience=LAST_DIGIT_IN_YOUR_SID)
```



```
Epoch 1/30
384/384 ━━━━━━━━━━━━━━━━━━━━ 4s 4ms/step - loss: 0.0195 - val_loss: 0.0235
Epoch 2/30
384/384 ━━━━━━━━━━━━━━━━━━━━ 2s 3ms/step - loss: 0.0081 - val_loss: 0.0020
Epoch 3/30
384/384 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0103 - val_loss: 0.0112
Epoch 4/30
384/384 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0031 - val_loss: 0.0068
Epoch 5/30
384/384 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0025 - val_loss: 0.0029
Epoch 6/30
384/384 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0018 - val_loss: 0.0089
Epoch 7/30
384/384 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0029 - val_loss: 0.0042
Epoch 8/30
384/384 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0014 - val_loss: 3.4910e-04
Epoch 9/30
384/384 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0044 - val_loss: 0.0076
Epoch 10/30
384/384 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0056 - val_loss: 0.0138
Epoch 11/30
384/384 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0017 - val_loss: 4.2040e-04
Epoch 12/30
384/384 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0014 - val_loss: 7.6541e-04
Epoch 13/30
384/384 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0018 - val_loss: 6.6156e-04
Epoch 14/30
384/384 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0017 - val_loss: 0.0069
Epoch 15/30
384/384 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0020 - val_loss: 8.6020e-04
Epoch 16/30
384/384 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0015 - val_loss: 0.0041
Epoch 17/30
384/384 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0018 - val_loss: 5.5051e-04
Epoch 18/30
384/384 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0022 - val_loss: 0.0022
Epoch 19/30
384/384 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0013 - val_loss: 5.8301e-04
Epoch 20/30
384/384 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - loss: 0.0012 - val_loss: 0.0014
```



```
plt.plot(epochs, val_acc_values, 'b', label='Validation loss')
plt.xlabel('Epochs')
plt.ylabel('Loss (mse)')
plt.legend()

plt.show()
```



```
[104]: ax=df.plot(figsize=(12,6))
       forecast_df.plot(ax=ax)
       plt.xlim('2020-01-01','2025-01-01')
       plt.show();
```