*# 1. Import the library***import** numpy **as** np  
**import** pandas **as** pd  
**import** tensorflow **as** tf  
**from** sklearn.metrics **import** root\_mean\_squared\_error  
  
*# 2. Load data  
# 2.1 Target Data*target\_Train = pd.read\_csv(**"D:/MY LECTURE 2025/DL - Week 2/Target\_data.csv"**)  
  
*# 2.2 Convert data to array*target\_Train\_arr = np.array(target\_Train)  
  
*# 2.3 Transpose target data*target\_Train\_tran = np.transpose(target\_Train\_arr)  
  
*# 3.1 Load Test data*Test\_data = pd.read\_csv(**"D:/MY LECTURE 2025/DL - Week 6/Input\_Testin\_Data.csv"**)  
*# 3.2 Convert data to array*Test\_data\_arr = np.array(Test\_data)  
*# 3.3 Transpose target data*Test\_data\_Tra = np.transpose(Test\_data\_arr)  
  
*# 3.4 Normalize the data  
# 3.4.1 formula to normalize***def** norm(x):  
 **return** (x - x.min()) / (x.max() - x.min())  
  
Test\_data\_norm = norm(Test\_data\_Tra)  
*# 4. Recreate the exact same model, including its weights and the optimizer*ann\_Model = tf.keras.models.load\_model(**'w6\_ANN\_1.h5'**)  
  
*# 5. Prediction the test data with models*pre\_Ann = ann\_Model.predict(Test\_data\_norm)  
  
*# 6. Un-Label*res\_Ann = pre\_Ann + target\_Train\_tran.min()  
  
*# 7. Save data to execl field*df1 = pd.DataFrame(np.transpose(res\_Ann))  
df1.to\_csv(**'D:/MY LECTURE 2025/DL - Week 6/W6\_Pred\_ANN.csv'**, index=**None**)