Issues in the “old” ML project:

**Block 1**:

1. He does not assess whether this data is actually suitable for a forecasting ML application
2. ‘datetime' and ‘atemp’ are actually possibly quite useful, while ‘index’ is not.
3. we need to check if the missing values are random. It’s better to impute them rather than dropping
4. **weathersit** is used as a continious variable, while it is categorical

**Block 2**:

1. mean values of the target are determined based on the original data rather than after dropping the missing values
2. The outliers are determined really arbitrarily
3. Importantly, ‘registered’ is a part of the target, including it among the features would be a serious mistake

**Block 3**:

1. Rather arbitrary features are added, and why only a square of the temperature?
2. Syntaxis of adding new columns is suboptimal (better use .assign inside pipelines)
3. Naming for the new squared temp. variable is very unfortunate and it is created with a wrong operator: \* rather than \*\*
4. Sum of all row-values is a strange and dangerous variable that has no meaningful interpretation and does not scale well when we may add new features in the future

**Block 4**:

1. Scaling is done on all data, so the training data would be contaminated with some test data information (data leakage)
2. Scaling should rather be done with sklearn
3. train-test split is done here very strangely, sampling like this does not actually ensure that no rows used for test are not used in train data. Besides 35% for test may be a little too much
4. CNT is scaled first and then a walkaround is used to not used these scaled values. Why not avoid scaling it in the first place?

**Block 5**:

1. Not mentioned that Decision Trees are very likely to overfit and may be a problematic choice
2. RMAE is not really a metric, MAE is already measured on the right scale, here it is clearly confused with RMSE where a sq. root is needed.
3. The best model is selected incorrectly: higher MAE is WORSE rather than better. Besides one should rather compare two models based on their best parameter specifications rather than the default ones
4. The parameter order is confused in the second MAE evaluation for model 2
5. MAE should be compared with the test set average rather than said to be “good” in abstraction

**Block 6**:

1. Why not use Sklearn’s grid search?
2. CV scores on the training data rather than the scores on the test set should be used for such evaluations
3. Why these parameters? Why would you try to find the “best” random state?
4. Other coding issues: e.g. rounding scores before comparing etc

**Block 7**:

1. Clearly the widget is very confusing and hard to use for our target audience
2. The predictions that it provides also seem to be misleading and arbitrary
3. Why focus only on 3 features? that’s not what an MVP is
4. Why is a Pandas rows displayed? For debugging?
5. Questions are unclear, no information about prediction certainty is given, and many other such issues

**Style issues**:

1. variable names, should be lowercase with underscores
2. spaces should separate operators
3. f-strings are prefered to “text” + str(non-text)
4. columns are referenced with df.col rather than df[‘col’]

…. many more