ASSUMPTIONS

1. “All drivers must complete an initial 'Transport of VIPs' training module before they are permitted to drive PAT's vehicles.”

~ Therefore, we assumed that all instances of drivers must also be tied to at least one instance of Training. Hence the cardinality of 1 .. * from DRIVER to TRAINING.

2. “PAT records the languages that a driver speaks – some drivers speak several languages”

~ Therefore, we assumed PAT will record all languages even if the language is not spoken by a driver.

3. A vehicle at the exact pick up date and pick up time can be booked by only once.

4. We store a catalogue of languages even if no drivers speaks the language (as with industrial standards).

For the below relationship, we assumed 0 .. many cardinality because there is a possibility that the entity may not have an instance yet or similar case:

1. We assumed that one Official can book zero or many trips due to the nature of not having any planned trip yet.

2. One vehicle may be new and unassigned to any trip yet.

3. One language may have not been used in any trip yet.

4. One language may not have been any driver’s preferred language yet.

5. A driver may not have been assigned a trip yet.

6. A training may not have been in completed yet.

ASSUMPTIONS for assignment1: logical

1. We assumed that for every location ID, there will be a unique return of the location name, location type, location address.
2. We assumed that for some of the training modules there will be an expiry period and it is counted in months and some of the training modules will not have an expiry period and will be represented as a null.
3. We assumed that driver completed training count to be mandatory with a maximum of numeric size of 2.
4. We assumed that the maximum char for the odometer is 7.

COMPOSITE KEYS:

5. Since the composite key in DRIVER_TRAINING_MODULE has more than 2 attibutes, we chose to include a surrogate key dtm_no.
6. Since the feature name should not be the primary key due to the type being varchar, and lead to duplicate values, a lookup table was implemented with surrogate key as feature_id to uniquely identify feature name.
7. Similarly, for the vehicle model we could not have the model name as the primary key so we introduced surrogate key as vehicle_model_id as part of lookup table VEHICLE_MODEL
8. We assumed that many models were manufactured by the same make so we introduced VEHICLE_MAKE lookup table to avoid redundancy.