





```

Enum location_type {
    city
    attraction
}

//// -- LEVEL 1
//// -- Tables and References

// Creating tables
Table trips as T {
    id int [pk, increment, NOT NULL]
    user_id int [NOT NULL]
    start_date start_date
    end_date end_date
}

Table locations as L {
    id int [pk, increment, NOT NULL] // auto-increment
    user_id int [NOT NULL]
    name varchar [NOT NULL]
    type location_type [NOT NULL]
    address varchar
    city varchar [NOT NULL]
    state varchar [NOT NULL]
    country varchar [NOT NULL]
    visited boolean [NOT NULL]
}

Table trips_locations as TL {
    trip_id int
    location_id int
}

Table albums as A {
    album_id int [pk, increment, NOT NULL]
    link varchar
    location_id int // can be tied to a location
    trip_id int // OR a trip
}

Table notes as N {
    note_id int [pk, increment, NOT NULL]
    note_header varchar [NOT NULL]
    note_body varchar
    location_id int [NOT NULL]
}

Table journal as J {
    entry_id int [pk, increment, NOT NULL]
    entry_body varchar
    date datetime
    location_id int [NOT NULL]
}

Table users as U {
    id int [pk, increment, NOT NULL]
    username varchar [NOT NULL]
    password varchar [NOT NULL]
}

// Creating references
// You can also define relationship separately
// > many-to-one; < one-to-many; - one-to-one
Ref: L.id < A.location_id
Ref: T.id < A.trip_id
Ref: L.id < N.location_id
Ref: L.id < J.location_id
Ref: TL.trip_id < T.id // many to many
Ref: TL.location_id < L.id // many to many
Ref: U.id < T.user_id
Ref: U.id < L.user_id

/-----//

//// -- Level 3
//// -- Enum, Indexes

// Enum for 'locations' table
Enum location_type {
    city
    attraction
}

```