AddTRavel

import { MigrationInterface, QueryRunner, Table } from 'typeorm';

export default class AddTravels1593744863677 implements *MigrationInterface* {

  public async up(*queryRunner*: *QueryRunner*): *Promise*<*void*> {

    await *queryRunner*.createTable(

**new** Table({

        name: 'travels',

        columns: [

          {

            name: 'id',

            type: 'uuid',

            isPrimary: true,

            generationStrategy: 'uuid',

            default: 'uuid\_generate\_v4()',

          },

          {

            name: 'originAdress',

            type: 'varchar',

          },

          {

            name: 'destinationAdress',

            type: 'varchar',

          },

          {

            name: 'departureDate',

            type: 'timestamp with time zone',

          },

          {

            name: 'arrivalDate',

            type: 'timestamp with time zone',

            isNullable: true,

          },

          {

            name: 'kilometersTraveled',

            type: 'decimal',

            isNullable: true,

            precision: 5,

            scale: 2,

          },

          {

            name: 'createdAt',

            type: 'timestamp with time zone',

            default: 'now()',

          },

          {

            name: 'updatedAt',

            type: 'timestamp with time zone',

            default: 'now()',

          },

        ],

      }),

    );

  }

  public async down(*queryRunner*: *QueryRunner*): *Promise*<*void*> {

    await *queryRunner*.dropTable('travels');

  }

}

AddVehicleToTravel

import {

  MigrationInterface,

  QueryRunner,

  TableColumn,

  TableForeignKey,

} from 'typeorm';

export default class AddVehicleToTravel1593744916233

  implements *MigrationInterface* {

  public async up(*queryRunner*: *QueryRunner*): *Promise*<*void*> {

    await *queryRunner*.addColumn(

      'travels',

**new** TableColumn({

        name: 'vehicleId',

        type: 'uuid',

      }),

    );

    await *queryRunner*.createForeignKey(

      'travels',

**new** TableForeignKey({

        name: 'vehicleOfTravel',

        columnNames: ['vehicleId'],

        referencedColumnNames: ['id'],

        referencedTableName: 'vehicles',

        onDelete: 'RESTRICT',

        onUpdate: 'CASCADE',

      }),

    );

  }

  public async down(*queryRunner*: *QueryRunner*): *Promise*<*void*> {

    await *queryRunner*.dropForeignKey('travels', 'vehicleOfTravel');

    await *queryRunner*.dropColumn('travels', 'vehicleId');

  }

}

AddAdress

import { MigrationInterface, QueryRunner, Table } from "typeorm";

export default class AddAdress1595296932106 implements *MigrationInterface* {

  public async up(*queryRunner*: *QueryRunner*): *Promise*<*void*> {

    await *queryRunner*.createTable(

**new** Table({

        name: 'adresses',

        columns: [

          {

            name: 'id',

            type: 'uuid',

            isPrimary: true,

            generationStrategy: 'uuid',

            default: 'uuid\_generate\_v4()',

          },

          {

            name: 'adressName',

            type: 'varchar',

          },

          {

            name: 'neighborhood',

            type: 'varchar'

          },

          {

            name: 'city',

            type: 'varchar'

          },

          {

            name: 'state',

            type: 'varchar'

          },

          {

            name: 'complement',

            type: 'varchar',

            isNullable: true,

          },

          {

            name: 'adressNumber',

            type: 'varchar'

          },

          {

            name: 'createdAt',

            type: 'timestamp with time zone',

            default: 'now()',

          },

          {

            name: 'updatedAt',

            type: 'timestamp with time zone',

            default: 'now()',

          },

        ],

      }),

    );

  }

  public async down(*queryRunner*: *QueryRunner*): *Promise*<*void*> {

    await queryRunner.dropTable('adresses');

  }

}

AddAdressToTravel

import { MigrationInterface, QueryRunner, TableColumn, TableForeignKey } from "typeorm";

export default class AddAdressToTravel1595298809865 implements *MigrationInterface* {

  public async up(*queryRunner*: *QueryRunner*): *Promise*<*void*> {

    await *queryRunner*.dropColumn('travels', 'originAdress');

    await *queryRunner*.dropColumn('travels', 'destinationAdress');

    await *queryRunner*.addColumn(

      'travels',

**new** TableColumn({

        name: 'originAdressId',

        type: 'uuid',

      }),

    );

    await *queryRunner*.addColumn(

      'travels',

**new** TableColumn({

        name: 'destinationAdressId',

        type: 'uuid',

      }),

    );

    await *queryRunner*.createForeignKey(

      'travels',

**new** TableForeignKey({

        name: 'OriginAdressTravel',

        columnNames: ['originAdressId'],

        referencedColumnNames: ['id'],

        referencedTableName: 'adresses',

      })

    )

    await *queryRunner*.createForeignKey(

      'travels',

**new** TableForeignKey({

        name: 'DestinationAdressTravel',

        columnNames: ['destinationAdressId'],

        referencedColumnNames: ['id'],

        referencedTableName: 'adresses',

      })

    )

  }

  public async down(*queryRunner*: *QueryRunner*): *Promise*<*void*> {

    await *queryRunner*.dropForeignKey('travels', 'OriginAdressTravel');

    await *queryRunner*.dropForeignKey('travels', 'DestinationAdressTravel');

    await *queryRunner*.addColumn(

      'travels',

**new** TableColumn({

        name: 'destinationAdress',

        type: 'varchar',

      }),

    );

    await queryRunner.addColumn(

      'travels',

**new** *TableColumn*({

        name: 'originAdress',

        type: 'varchar',

      }),

    );

  }

}