const express = require('express')

const {open} = require('sqlite')

const sqlite3 = require('sqlite3')

const path = require('path')

const databasePath = path.join(\_\_dirname, 'covid19India.db')

const app = express()

app.use(express.json())

let database = null

const initializeDbAndServer = async () => {

  try {

    database = await open({

      filename: databasePath,

      driver: sqlite3.Database,

    })

    app.listen(3000, () =>

      console.log('Server Running at http://localhost:3000/'),

    )

  } catch (error) {

    console.log('DB Error:${error.message}')

    process.exit(1)

  }

}

initializeDbAndServer()

const convertStateDbObjectToResponseObject = dbObject => {

  return {

    stateId: dbObject.state\_id,

    stateName: dbObject.state\_name,

    population: dbObject.population,

  }

}

convertDistrictDbObjectToResponseObject = dbObject => {

  return {

    districtId: dbObject.district\_id,

    districtName: dbObject.district\_name,

    stateId: dbObject.state\_id,

    cases: dbObject.cases,

    cured: dbObject.cured,

    active: dbObject.active,

    deaths: dbObject.deaths,

  }

}

app.get('/states/', async (request, response) => {

  const getStateQuery = `

    SELECT

    \*

    FROM

    state

    ORDER BY

    state\_id;`

  const stateArray = await database.all(getStateQuery)

  response.send(

    stateArray.map(eachState =>

      convertStateDbObjectToResponseObject(eachState),

    ),

  )

})

app.get('/states/:stateId/', async (request, response) => {

  const {stateId} = request.params

  const getStateQuery = `

    SELECT

    \*

    FROM

    state

    WHERE

    state\_id = ${stateId};`

  const state = await database.get(getStateQuery)

  response.send(convertStateDbObjectToResponseObject(state))

})

app.post('/districts/', async (request, response) => {

  const {districtName, stateId, cases, cured, active, deaths} = request.body

  const postDistrictQuery = `

    INSERT INTO

        district(distrct\_name,state\_id,cases,cured,active,deaths)

    VALUES

    ('${districtName}',${stateId},${cases},${cured},${active},${deaths});`

  await database.run(postDistrictQuery)

  response.send('District Successfully Added')

})

app.get('/districts/:districtId/', async (request, response) => {

  const {districtId} = request.params

  const getDistrictQuery = `

  SELECT

  \*

  FROM

  district

  WHERE

  district\_id = ${districtId};`

  const district = await database.get(getDistrictQuery)

  response.send(convertDistrictDbObjectToResponseObject(district))

})

app.delete('/districts/:districtId', async (request, response) => {

  const {districtId} = request.params

  const deleteDistrictQuery = `

  DELETE

  FROM

  district

  WHERE

  district\_id = ${districtId};`

  await database.run(deleteDistrictQuery)

  response.send('District Removed')

})

app.put('/districts/:districtId', async (request, response) => {

  const {districtId} = request.params

  const {districtName, stateId, cases, cured, active, deaths} = request.body

  const updateDistrictQuery = `

  UPDATE

    district

  SET

    district\_name = "${districtName}";

    state\_id = ${stateId};

    cases = ${cases};

    cured = ${cured};

    active = ${active};

    deaths = ${deaths};

  WHERE

    district\_id = ${districtId};`

  await database.run(updateDistrictQuery)

  response.send('District Details Updated')

})

app.get('/states/:stateId/stats', async (request, response) => {

  const {stateId} = request.params

  const getDistrictStateQuery = `

  SELECT

  SUM(cases) as totalCases,

  SUM(cured) as totalCured,

  SUM(active) as totalActive,

  SUM(deaths) as totalDeaths

  FROM

  district

  WHERE

  state\_id = ${stateId};`

  const stateArray = await database.get(getDistrictStateQuery)

  response.send(stateArray)

})

app.get('/districts/:districtId/details/', async (request, response) => {

  const {districtId} = request.params

  const getDistrictIdQuery = `

  SELECT

  state\_name

  FROM

  state JOIN

  district ON state.state\_id = district.state\_id

  WHERE

  district.district\_id = ${districtId};`

  const getDistrictIdQueryResponse = await database.get(getDistrictIdQuery)

  response.send({stateName: getDistrictIdQueryResponse.state\_name})

})

module.exports = app