

```
/*supprimer étudiant*/
```

```
const express = require('express');  
const mysql = require('mysql');  
const bodyParser = require('body-parser');
```

```
const app = express();  
const port = 3001;
```

```
app.use(bodyParser.json());  
app.use(bodyParser.urlencoded({ extended: true }));
```

```
// Définition de l'en-tête pour permettre l'accès depuis n'importe où  
app.use((req, res, next) => {  
  res.setHeader('Access-Control-Allow-Origin', '*');  
  res.setHeader('Access-Control-Allow-Methods', 'GET, POST, PUT, DELETE');  
  res.setHeader('Access-Control-Allow-Headers', 'Content-Type, Authorization');  
  next();  
});
```

```
const db = mysql.createConnection({  
  host: 'localhost',  
  user: 'root',  
  password: '',
```

```
    database: 'test',
  });

  db.connect((err) => {
    if (err) {
      throw err;
    }
    console.log('Connecté à la base de données MySQL');
  });

// Récupération de tous les étudiants
app.get('/etudiants', (req, res) => {
  const sql = 'SELECT * FROM etudiants';
  db.query(sql, (err, result) => {
    if (err) throw err;
    res.send(result);
  });
});

// Route pour ajouter un nouvel étudiant à la table 'etudiants'
app.post('/etudiants', (req, res) => {
  const nom = req.body.nom;
  const values = [[nom]];

  const sql = 'INSERT INTO etudiants (nom) VALUES ?';

  db.query(sql, [values], (err, result) => {
```

```
    if (err) {
      console.error("Erreur lors de l'ajout de l'étudiant :", err);
      return;
    }
    console.log('Étudiant ajouté avec succès à la base de données');
  });
});

// Mise à jour d'un étudiant par ID
app.put('/etudiants/:id', (req, res) => {
  const id = req.params.id;
  const nom = req.body.nom;
  const sql = 'UPDATE etudiants SET nom = ? WHERE id = ?';
  db.query(sql, [nom, id], (err, result) => {
    if (err) throw err;
    res.send(result);
  });
});

// Route pour supprimer un étudiant de la table 'etudiants'
app.delete('/etudiants/:id', (req, res) => {
  const id = req.params.id;
  const sql = `DELETE FROM etudiants WHERE id = ?`;

  db.query(sql, id, (err, result) => {
    if (err) throw err;
    res.send(result);
  });
});
```

```
});  
});
```

```
app.listen(port, () => {  
  console.log(`Serveur Express en cours d'exécution sur le port ${port}`);  
});
```