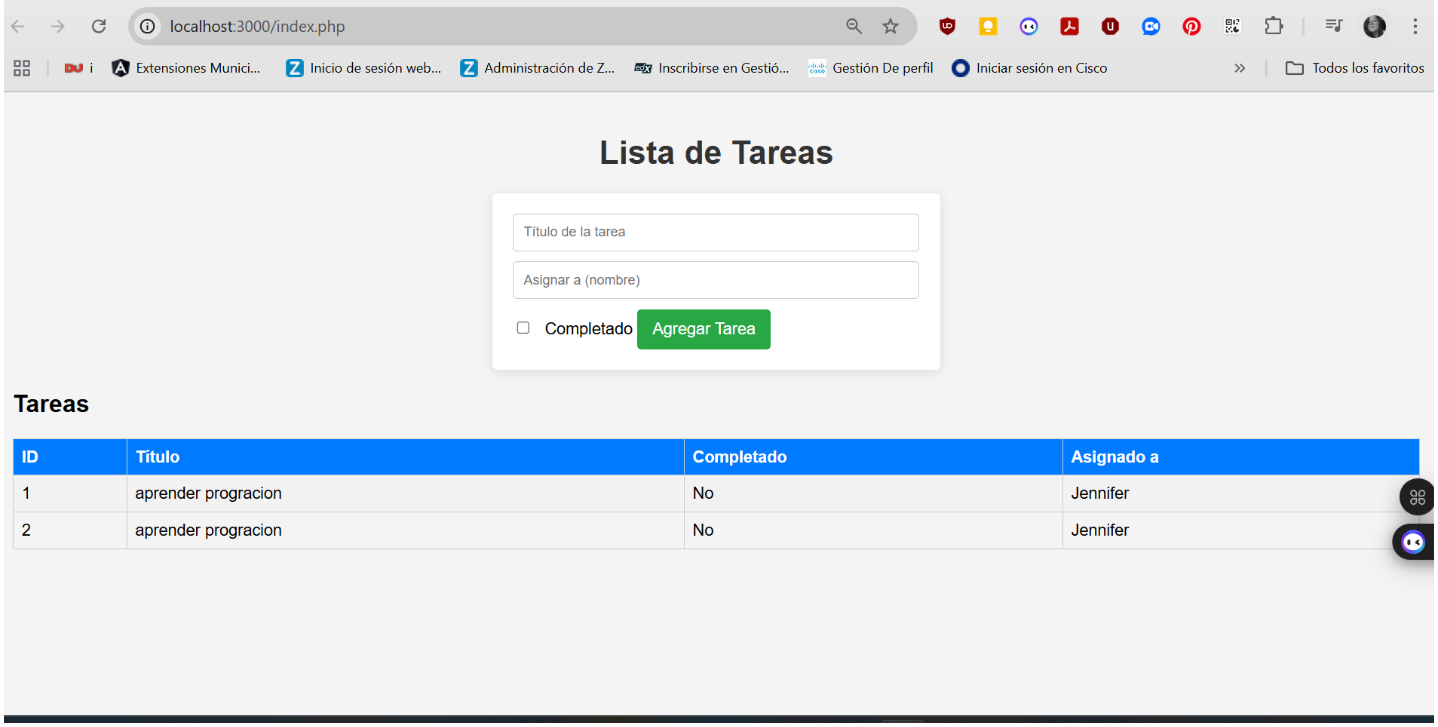
Funcionamiento



Lista de Tareas

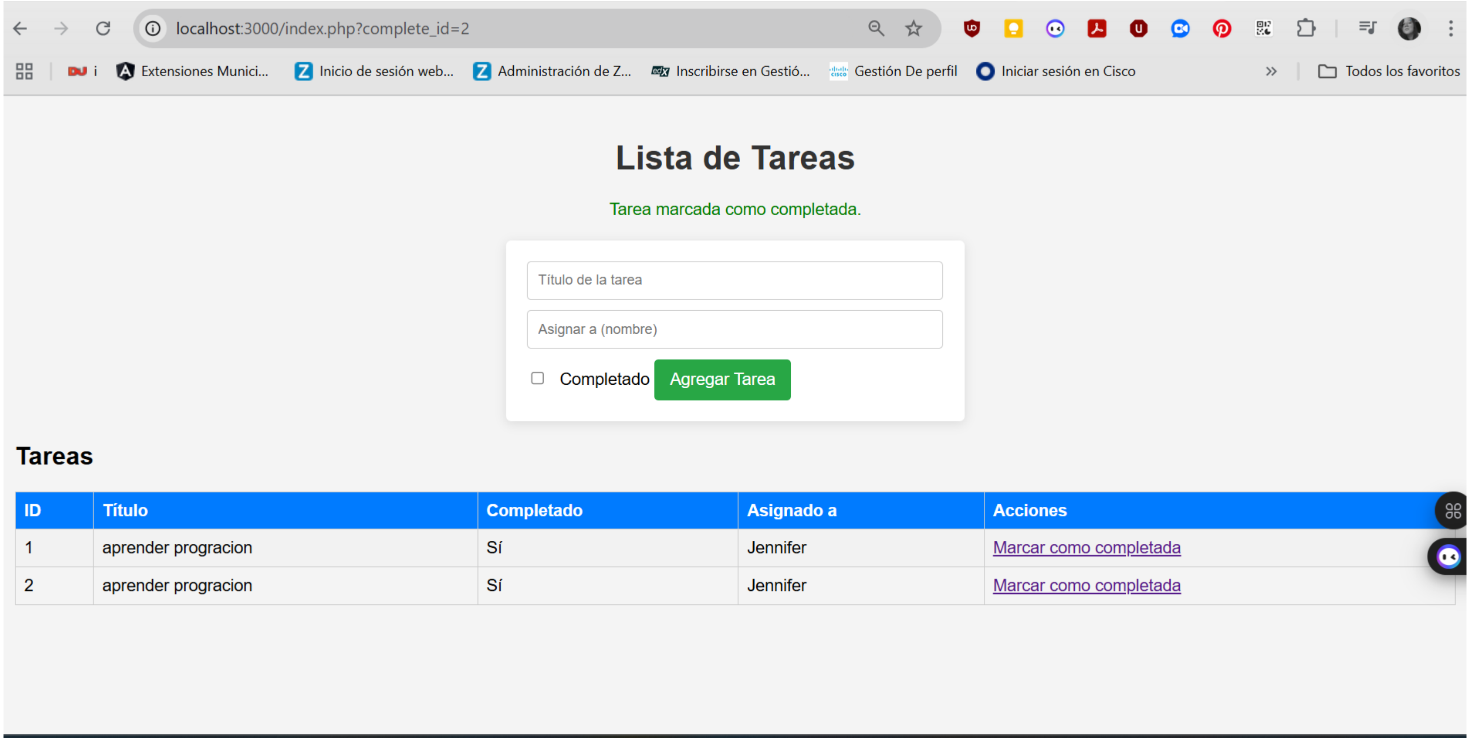
Título de la tarea

Asignar a (nombre)

□ Completado **Agregar Tarea**

Tareas

ID	Título	Completado	Asignado a
1	aprender progracion	No	Jennifer
2	aprender progracion	No	Jennifer



Lista de Tareas

Tarea marcada como completada.

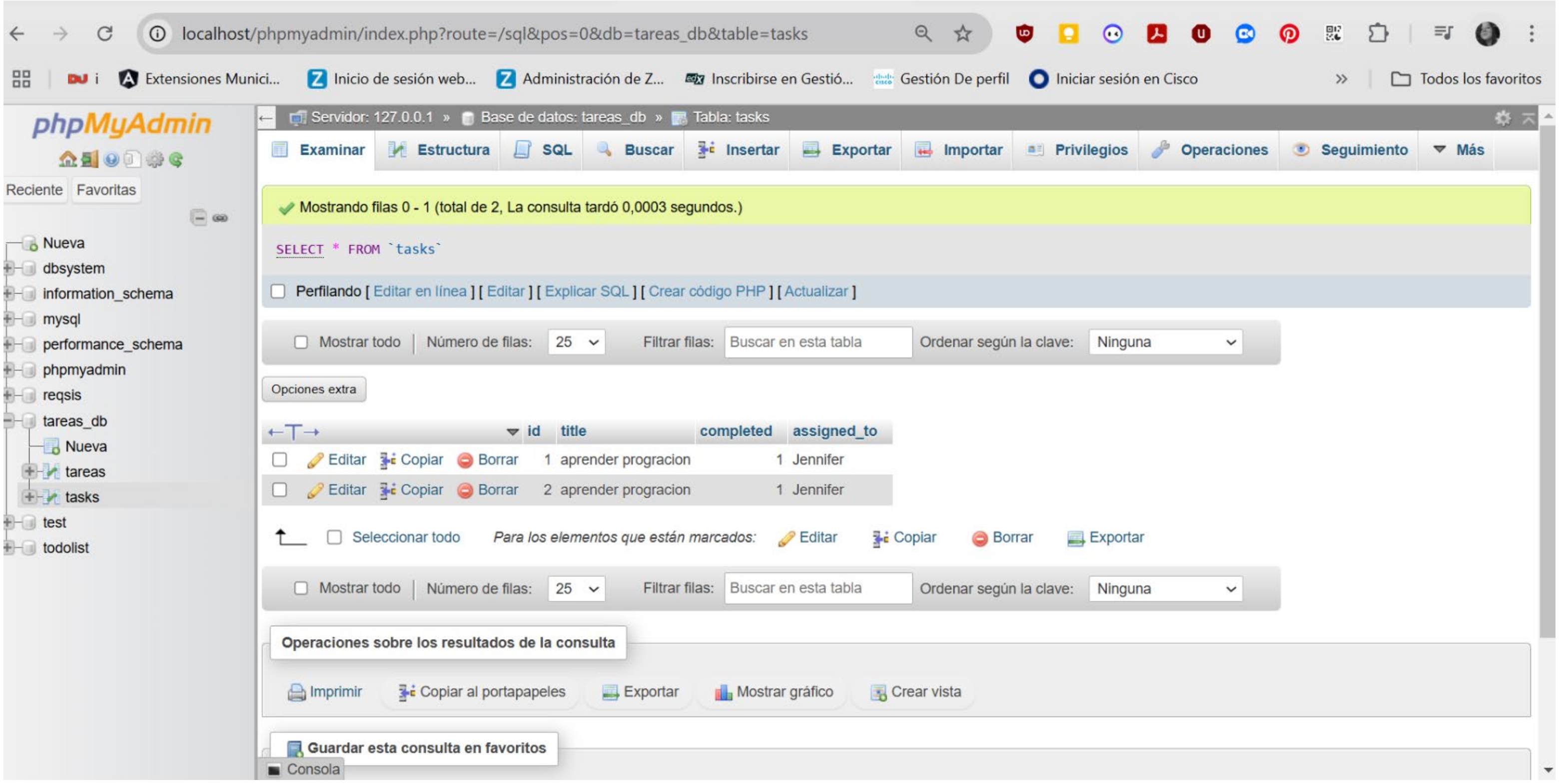
Título de la tarea

Asignar a (nombre)

□ Completado **Agregar Tarea**

Tareas

ID	Título	Completado	Asignado a	Acciones
1	aprender progracion	Sí	Jennifer	Marcar como completada
2	aprender progracion	Sí	Jennifer	Marcar como completada



Mostrando filas 0 - 1 (total de 2, La consulta tardó 0,0003 segundos.)

```
SELECT * FROM `tasks`
```

	id	title	completed	assigned_to
Editar Copiar Borrar	1	aprender progracion	1	Jennifer
Editar Copiar Borrar	2	aprender progracion	1	Jennifer

Index.php

```
<?php
// index.php

include 'db.php'; // Incluye el archivo de conexión

// Variable para almacenar mensajes
$message = '';

// Verifica si se envió el formulario
if ($_SERVER["REQUEST_METHOD"] == "POST") {
  $title = $_POST['title']; // Obtiene el título de la tarea
  $assigned_to = $_POST['assigned_to']; // Obtiene la persona asignada
  $completed = isset($_POST['completed']) ? 1 : 0; // Obtiene el estado de completado

  // Consulta para insertar la tarea
  $sql = "INSERT INTO tasks (title, completed, assigned_to) VALUES (?, ?, ?)";
  $stmt = $conn->prepare($sql);

  if ($stmt === false) {
    $message = "Error en la preparación de la consulta: " . $conn->error;
  } else {
    $stmt->bind_param("sis", $title, $completed, $assigned_to); // "sis" significa string,
integer, string

    if ($stmt->execute()) {
      $message = "Nueva tarea agregada con éxito.";
    } else {
      $message = "Error al agregar tarea: " . $stmt->error;
    }

    $stmt->close();
  }
}

// Marca la tarea como completada
if (isset($_GET['complete_id'])) {
  $complete_id = $_GET['complete_id'];
  $sql = "UPDATE tasks SET completed = 1 WHERE id = ?";
  $stmt = $conn->prepare($sql);
  $stmt->bind_param("i", $complete_id);

  if ($stmt->execute()) {
    $message = "Tarea marcada como completada.";
  } else {
    $message = "Error al marcar tarea: " . $stmt->error;
  }

  $stmt->close();
}

// Consulta para obtener todas las tareas
$sql = "SELECT * FROM tasks";
$result = $conn->query($sql);

// Verifica si la consulta fue exitosa
if ($result === false) {
  die("Error en la consulta: " . $conn->error);
}
?>

<!DOCTYPE html>
<html lang="es">
<head>
  <meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Lista de Tareas</title>
<style>
  /* Estilos básicos */
  body {
    font-family: Arial, sans-serif;
    margin: 0;
    padding: 20px;
    background-color: #f4f4f4;
  }

  h1 {
    color: #333;
    text-align: center;
  }

  form {
    background: #fff;
    padding: 20px;
    border-radius: 5px;
    box-shadow: 0 2px 10px rgba(0, 0, 0, 0.1);
    margin-bottom: 20px;
    max-width: 400px;
    margin-left: auto;
    margin-right: auto;
  }

  input[type="text"] {
    width: calc(100% - 22px);
    padding: 10px;
    margin-bottom: 10px;
    border: 1px solid #ccc;
    border-radius: 4px;
  }

  input[type="checkbox"] {
    margin-right: 10px;
  }

  button {
    background-color: #28a745;
    color: white;
    padding: 10px 15px;
    border: none;
```

```
      border-radius: 4px;
      cursor: pointer;
      font-size: 16px;
    }

    button:hover {
      background-color: #218838;
    }

    table {
      width: 100%;
      border-collapse: collapse;
      margin-top: 20px;
    }

    th, td {
      border: 1px solid #ccc;
      padding: 8px;
      text-align: left;
    }

    th {
      background-color: #007bff;
      color: white;
    }

    tr:nth-child(even) {
      background-color: #f2f2f2;
    }

    tr:hover {
      background-color: #e9ecef;
    }

    .message {
      color: green;
      text-align: center;
      margin-bottom: 20px;
    }
  </style>
</head>
<body>

<h1>Lista de Tareas</h1>
```

```
<?php if ($message): ?>
  <div class="message"><?php echo htmlspecialchars($message); ?></div>
<?php endif; ?>

<form method="POST" action="">
  <input type="text" name="title" placeholder="Título de la tarea" required>
  <input type="text" name="assigned_to" placeholder="Asignar a (nombre)" required>
  <label>
    <input type="checkbox" name="completed"> Completado
  </label>
  <button type="submit">Agregar Tarea</button>
</form>

<h2>Tareas</h2>
<table>
  <tr>
    <th>ID</th>
    <th>Título</th>
    <th>Completado</th>
    <th>Asignado a</th>
    <th>Acciones</th>
  </tr>
  <?php
  if ($result->num_rows > 0) {
    // Salida de cada fila
    while($row = $result->fetch_assoc()) {
      echo "<tr>
            <td>" . $row["id"] . "</td>
            <td>" . $row["title"] . "</td>
            <td>" . ($row["completed"] ? 'Sí' : 'No') . "</td>
            <td>" . htmlspecialchars($row["assigned_to"]) . "</td>
            <td>
              <a href='?complete_id=" . $row["id"] . "' onclick='return confirm(\"¿Estás seguro de
que quieres marcar esta tarea como completada?\");'>Marcar como completada</a>
            </td>
           </tr>";
    }
  } else {
    echo "<tr><td colspan='5'>No hay tareas.</td></tr>";
  }
  ?>
</table>
```

```
<?php
// Cierra la conexión a la base de datos
if (isset($conn) && $conn) {
   $conn->close(); // Asegúrate de que $conn esté definido
}
?>

</body>
</html>
```

// insert_task.php

```
<?php

include 'db.php'; // Incluye el archivo de conexión

$title = "Aprender PHP"; // Título de la tarea
$completed = false; // Estado de completado

// Consulta para insertar la tarea
$sql = "INSERT INTO tasks (title, completed) VALUES (?, ?)";

$stmt = $conn->prepare($sql);
$stmt->bind_param("si", $title, $completed); // "si" significa string, integer

if ($stmt->execute()) {
   echo "Nueva tarea agregada con éxito. ID: " . $conn->insert_id;
} else {
   echo "Error: " . $stmt->error;
}

$stmt->close();
$conn->close();
?>
```

Db.php

```
<?php

$servername = "localhost"; // Cambia esto si tu servidor de base de datos es diferente
$username = "root"; // Tu usuario de MySQL
$password = ""; // Tu contraseña de MySQL (deja vacío si no tienes)
$dbname = "tareas_db"; // Nombre de la base de datos
```

```
// Crear conexión
$conn = new mysqli($servername, $username, $password, $dbname);

// Verificar conexión
if ($conn->connect_error) {
  die("Conexión fallida: " . $conn->connect_error);
}
//echo "Conectado exitosamente a la base de datos";
?>
```

Api.php

```
<?php
$host = 'localhost';
$user = 'root'; // Cambia esto a tu usuario de MySQL
$password = ''; // Cambia esto a tu contraseña de MySQL
$dbname = 'todolist';

$conn = new mysqli($host, $user, $password, $dbname);

if ($conn->connect_error) {
  die("Conexión fallida: " . $conn->connect_error);
}

header("Content-Type: application/json");

$request_method = $_SERVER['REQUEST_METHOD'];

switch ($request_method) {
  case 'GET':
    // Listar todas las tareas
    $result = $conn->query("SELECT * FROM tasks");
    $tasks = $result->fetch_all(MYSQLI_ASSOC);
    echo json_encode($tasks);
    break;

  case 'POST':
    // Crear una nueva tarea
    $data = json_decode(file_get_contents("php://input"), true);
    $name = $conn->real_escape_string($data['name']);
    $assignee = $conn->real_escape_string($data['assignee']);
    $conn->query("INSERT INTO tasks (name, assignee) VALUES ('$name', '$assignee')");
```

```
        echo json_encode(['message' => 'Tarea creada exitosamente', 'task' => ['id' => $conn->insert_id, 'name' => $name, 'assignee' => $assignee]]);
        break;

    case 'PUT':
        // Actualizar una tarea existente
        $data = json_decode(file_get_contents("php://input"), true);
        $id = $data['id'];
        $name = $conn->real_escape_string($data['name']);
        $assignee = $conn->real_escape_string($data['assignee']);
        $conn->query("UPDATE tasks SET name='$name', assignee='$assignee' WHERE id=$id");
        echo json_encode(['message' => 'Tarea actualizada exitosamente', 'task' => ['id' => $id, 'name' => $name, 'assignee' => $assignee]]);
        break;

    case 'DELETE':
        // Eliminar una tarea
        $id = intval(basename($_SERVER['REQUEST_URI']));
        $conn->query("DELETE FROM tasks WHERE id=$id");
        echo json_encode(['message' => 'Tarea eliminada exitosamente']);
        break;

    default:
        echo json_encode(['message' => 'Método no permitido']);
        break;
}

$conn->close();
?>
```

// view_tasks.php

```php
<?php

include 'db.php'; // Incluye el archivo de conexión

$sql = "SELECT * FROM tasks";
$result = $conn->query($sql);

if ($result->num_rows > 0) {
  // Salida de cada fila
  while($row = $result->fetch_assoc()) {
    echo "ID: " . $row["id"]. " - Título: " . $row["title"]. " - Completado: " . ($row["completed"] ? 'Sí' : 'No') . "<br>";
  }
} else {
  echo "No hay tareas.";
}

$conn->close();
?>
```

Package.json

```json
{
 "dependencies": {
  "mysql": "^2.18.1"
 }
}
```

Package-lock.json

```json
{
 "name": "Examen final",
 "lockfileVersion": 3,
 "requires": true,
 "packages": {
  "": {
   "dependencies": {
    "mysql": "^2.18.1"
   }
  },
  "node_modules/bignumber.js": {
   "version": "9.0.0",
   "resolved": "https://registry.npmjs.org/bignumber.js/-/bignumber.js-9.0.0.tgz",
   "integrity": "sha512-
t/OYhhJ2SD+YGBQcjY8GzzDHEk9f3nerxjtfa6tlMXfe7frs/WozhvCNoGvpM0P3bNf3Gq5ZRMlGr
5f3r4/N8A==",
   "engines": {
    "node": "*"
   }
  },
  "node_modules/core-util-is": {
   "version": "1.0.3",
   "resolved": "https://registry.npmjs.org/core-util-is/-/core-util-is-1.0.3.tgz",
   "integrity": "sha512-
ZQBvi1DcpJ4GDqanjucZ2Hj3wEO5pZDS89BWbkcrvdxksJorwUDDZamX9ldFkp9aw2lmBDLgk
ObEA4DWNJ9FYQ=="
  },
  "node_modules/inherits": {
   "version": "2.0.4",
   "resolved": "https://registry.npmjs.org/inherits/-/inherits-2.0.4.tgz",
   "integrity": "sha512-
k/vGaX4/Yla3WzyMCvTQOXYeIHvqOKtnqBduzTHpzpQZzAskKMhZ2K+EnBiSM9zGSoIFeMpXKx
a4dYeZIQqewQ=="
  },
  "node_modules/isarray": {
   "version": "1.0.0",
   "resolved": "https://registry.npmjs.org/isarray/-/isarray-1.0.0.tgz",
   "integrity": "sha512-
VLghIWNM6ELQzo7zwmcg0NmTVyWKYjvIeM83yjp0wRDTmUnrM678fQbcKBo6n2CJEF0szoG/
/ytg+TKla89ALQ=="
  },
  "node_modules/mysql": {
   "version": "2.18.1",
   "resolved": "https://registry.npmjs.org/mysql/-/mysql-2.18.1.tgz",
   "integrity": "sha512-
Bca+gk2YWmqp2Uf6k5NFEurwY/0td0cpebAucFpY/3jhrwrVGuxU2uQFCHjU19SJfje0yQvi+rV
Wdq78hR5lig==",
   "dependencies": {
    "bignumber.js": "9.0.0",
    "readable-stream": "2.3.7",
    "safe-buffer": "5.1.2",
      "sqlstring": "2.3.1"
    },
    "engines": {
      "node": ">= 0.6"
    }
  },
  "node_modules/process-nextick-args": {
    "version": "2.0.1",
    "resolved": "https://registry.npmjs.org/process-nextick-args/-/process-nextick-args-
2.0.1.tgz",
    "integrity": "sha512-
3ouUOpQhtgrbOa17J7+uxOTpITYWaGP7/AhoR3+A+/1e9skrzelGi/dXzEYyvbxubEF6Wn2ypscT
KiKJFFn1ag=="
  },
  "node_modules/readable-stream": {
    "version": "2.3.7",
    "resolved": "https://registry.npmjs.org/readable-stream/-/readable-stream-2.3.7.tgz",
    "integrity": "sha512-
Ebho8K4jIbHAxnuxi7o42OrZgF/ZTNcsZj6nRKyUmkhLFq8CHItp/fy6hQZuZmP/n3yZ9VBUbp4zz
/mX8hmYPw==",
    "dependencies": {
      "core-util-is": "~1.0.0",
      "inherits": "~2.0.3",
      "isarray": "~1.0.0",
      "process-nextick-args": "~2.0.0",
      "safe-buffer": "~5.1.1",
      "string_decoder": "~1.1.1",
      "util-deprecate": "~1.0.1"
    }
  },
  "node_modules/safe-buffer": {
    "version": "5.1.2",
    "resolved": "https://registry.npmjs.org/safe-buffer/-/safe-buffer-5.1.2.tgz",
    "integrity": "sha512-
Gd2UZBJDkXlY7GbJxfsE8/nvKkUEU1G38c1siN6QP6a9PT9MmHB8GnpscSmMJSoF8LOIrt8ud/
wPtojys4G6+g=="
  },
  "node_modules/sqlstring": {
    "version": "2.3.1",
    "resolved": "https://registry.npmjs.org/sqlstring/-/sqlstring-2.3.1.tgz",
    "integrity": "sha512-
ooAzh/7dxIG5+uDik1z/Rd1vli0+38izZhGzSa34FwR7IbelPWCCKSNIl8jlL/F7ERvy8CB2jNeM1E
9i9mXMAQ==",
    "engines": {
      "node": ">= 0.6"
    }
  },
  "node_modules/string_decoder": {
    "version": "1.1.1",
    "resolved": "https://registry.npmjs.org/string_decoder/-/string_decoder-1.1.1.tgz",
    "integrity": "sha512-n/ShnvDi6FHbbVfviro+WojiFzv+s8MPMHBczVePfUpDJLwoLT0ht1l4YwBCbi8pJAveEEdnkHyPyTP/mzRfwg==",
    "dependencies": {
      "safe-buffer": "~5.1.0"
    }
  },
  "node_modules/util-deprecate": {
    "version": "1.0.2",
    "resolved": "https://registry.npmjs.org/util-deprecate/-/util-deprecate-1.0.2.tgz",
    "integrity": "sha512-EPD5q1uXyFxJpCrLnCc1nHnq3gOa6DZBocAIiI2TaSCA7VCJ1UJDMagCzIkXNsUYfD1daK//LTEQ8xiIbrHtcw=="
  }
}
}
```