



FPT POLYTECHNIC



THỰC HỌC – THỰC NGHIỆP



Conceive Design Implement Operate

NODEJS & RESFUL WEB SERVICE

LÀM VIỆC VỚI MYSQL

- ⊙ Biết sử dụng module thao tác với cơ sở dữ liệu
- ⊙ Truy vấn dữ liệu từ database mysql
- ⊙ Thực hiện quản trị cơ sở dữ liệu mysql hoàn chỉnh



📖 MySQL?

📖 Cài đặt và sử dụng trong nodejs

📖 Truy vấn dữ liệu mySql với nodejs

📖 Thao tác trên mySQL



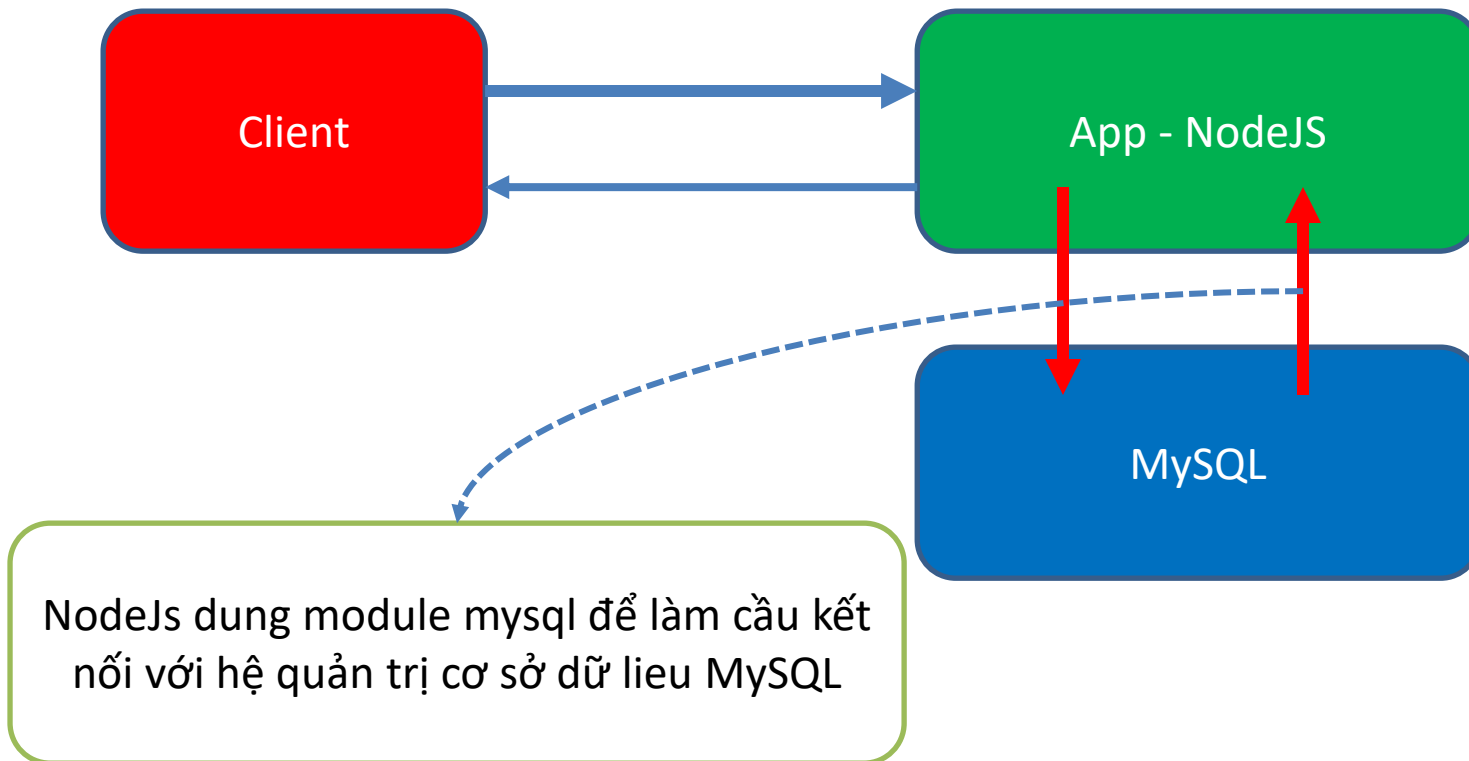


PHẦN 1: TỔNG QUAN VÀ TRUY VẤN DỮ LIỆU MYSQL

□ Là hệ quản trị cơ sở dữ liệu, hỗ trợ:

- ❖ Tổ chức cơ sở dữ liệu quan hệ
- ❖ Thao tác trên dữ liệu
- ❖ Kết nối với các ngôn ngữ lập trình server side và thao tác

❑ Mô hình kết nối giữa nodejs với MySql



Cài đặt module mysql

```
Npm install mysql
```

Tạo kết nối tới server mysql

```
//sử dụng module database  
var mysql = require('mysql');  
  
// create connection to database  
const db = mysql.createConnection ({  
  host: 'localhost',  
  user: 'root',  
  password: '',  
  database: 'bookstore'  
});
```

Sử dụng hàm query

Khai báo chuỗi truy vấn

Thực hiện truy vấn với hàm query

```
app.get('/shop',(req,res)=>{  
  let sql = `SELECT * FROM products`;  
  db.query(sql, function(err, data) {  
    if (err) throw err;  
    res.render('shop',{products:data});  
  })  
})
```


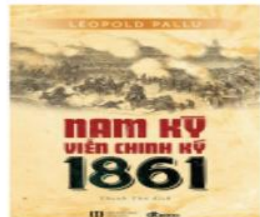

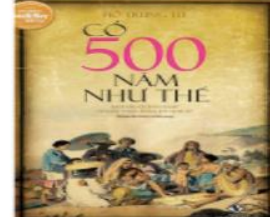
Biến chứa kết quả thực hiện truy vấn nếu thành công

Hiển thị kết quả ra giao diện

```
<div class="grid">
  <% for (let product of products) { %>
    <article class="card product-item">
      <header class="card_header">
        <h1 class="product_title"><%= product.nameProduct %></h1>
      </header>
      <div class="card_image">
```

← → ↻ ⓘ localhost:3000/shop ☆ ⓘ

Trang chủ Sản phẩm Thêm sản phẩm

<p>5 Múi Giờ, 10 Tiếng Bay Và Một Cái Khép Mi</p>  <p>\$300000</p> <p>5 Múi Giờ, 10 Tiếng Bay Và Một Cái Khép Mi</p> <p>Add to Cart</p>	<p>5 Múi Giờ, 10 Tiếng Bay Và Một Cái Khép Mi</p>  <p>\$300000</p> <p>Gói Nỗi Buồn Lại Và Ném Đi Thật Xa</p> <p>Add to Cart</p>	<p>Gói Nỗi Buồn Lại Và Ném Đi Thật Xa</p>  <p>\$200000</p> <p>Gói Nỗi Buồn Lại Và Ném Đi Thật Xa</p> <p>Add to Cart</p>	<p>5 Múi Giờ, 10 Tiếng Bay Và Một Cái Khép Mi</p>  <p>\$300000</p> <p>Gói Nỗi Buồn Lại Và Ném Đi Thật Xa</p> <p>Add to Cart</p>
---	---	--	---

Truy vấn với điều kiện là tham số: url/route/value

//lấy chi tiết sản phẩm theo mã

```
app.get("/shop/:id",(req,res)=>{
```

```
  let id=req.params.id;
```

```
  let sql = `SELECT * FROM products where idProduct=${id}`;
```

```
  db.query(sql, function(err, data) {
```

```
    if (err) throw err;
```

```
    res.send(data)
```

```
  })
```

```
})
```

Lấy tham số

Điều kiện với truy
vấn với tham số

Thực hiện truy vấn Thống kê

//thống kê số mặt hàng của từng loại hàng

```
app.get("/shop/:id",(req,res)=>{  
  let sql =  
    `SELECT nameCategory, COUNT(*) as numOfProduct FROM products, catalog WHERE products.idCategory=catalog.idCategory GROUP BY nameCategory`;  
  db.query(sql, function(err, data) {  
    if (err) throw err;  
    res.send(data)  
  })  
})
```

Truy vấn thống kê

Nhận kết quả trả về

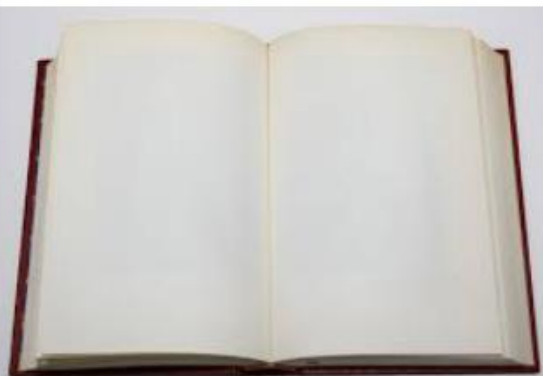
Sản phẩm Demo

Trang chủ

Sản phẩm

Thêm sản phẩm

Apple Book

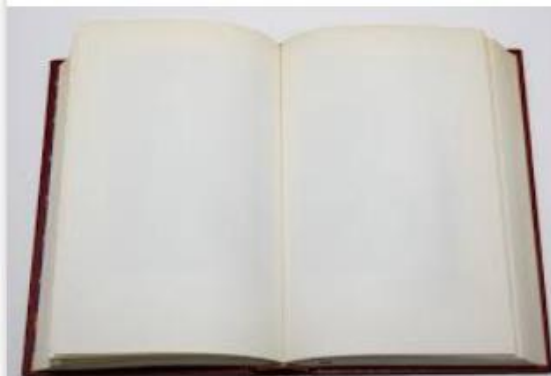


\$2000

A very interesting book about so many even more interesting things!

Add to Cart

Microsoft Book

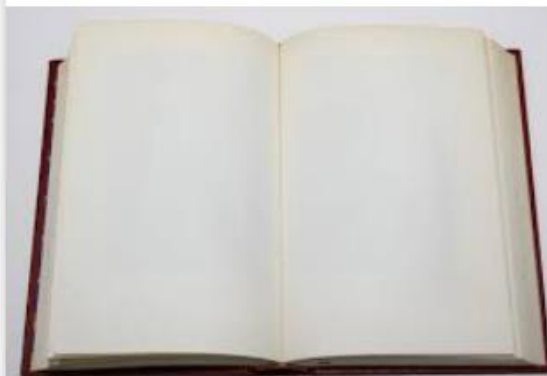


\$2000

A very interesting book about so many even more interesting things!

Add to Cart

VFast Book



\$1000

A very interesting book about so many even more interesting things!

Add to Cart

demo



PHẦN 2: THAO TÁC TRÊN CƠ SỞ DỮ LIỆU

Thêm dữ liệu vào table



//Cách thứ nhất

```
let sql = `insert into catalog(name) values ("Thiếu  
nhi")`;  
db.query(sql, function(err, data) {  
    if (err) throw err;  
    res.send(data)  
})
```

Thêm dữ liệu vào table



//Cách thứ 2

```
var cate= {name:"Thiếu nhi"}
```

```
db.query('insert into catalog SET ?',cate function(err, data) {
```

```
  if (err) throw err;
```

```
  res.send(data)
```

```
})
```


Thêm dữ liệu vào table từ form

```
const file = req.file  
let title=req.body.productName;  
let namelImage=file.filename;
```

Lấy dữ liệu từ form

```
product={nameProduct:title,priceProduct:price,  
sortDescription:description,  
images:namelImage,  
}
```

Tạo biến json data thêm vào csdl

Thêm dữ liệu vào table từ form

```
product={nameProduct:title,priceProduct:price,  
sortDescription:description,  
images:nameImage,  
}
```

Tạo biến json data thêm vào csdl

```
db.query('insert into products SET ?',product, function(err, data) {  
if (err) throw err;  
res.redirect('/shop');  
}}});
```

Tạo database với nodeJS

```
//sử dụng module database  
var mysql = require('mysql');  
  
// create connection to database  
const db = mysql.createConnection ({  
  host: 'localhost',  
  user: 'root',  
  password: '',  
});
```

Tạo database với nodeJS

```
//tiến hành kết nối
db.connect(function(err){
  if(err) throw err;
  db.query("CREATE DATABASE tintuc",function(err, result){
    if(err) throw err;
    console.log("Tạo thành công database!");
  })
})
```

Thêm dữ liệu mẫu với thư viện Faker

Cài đặt thư viện: `npm install faker`

```
var faker = require('faker');
```

```
var person = {  
  email: faker.internet.email(),  
  created_at: faker.date.past()  
};
```

```
var end_result = db.query('INSERT INTO users SET ?', person, function(err,  
result) {  
  if (err) throw err;  
  console.log(result);  
});
```

Xoá dữ liệu

```
let id=req.params.idProduct;
```

Lấy điều kiện xoá

```
db.query("DELETE FROM products WHERE id = ?", [id], function(err, data) {  
  if (err) throw err;  
  res.redirect('/shop');  
}});
```

[id] là điều kiện xoá
Nếu điều kiện xoá là nhiều thì [value1, values2,...]

Sửa dữ liệu

```
const file = req.file  
let title=req.body.productName;  
let nameImage=file.filename;
```

Lấy dữ liệu update từ form

```
db.query("UPDATE products SET nameProduct=?,priceProduct=?,sortDescription =?  
WHERE id = ?", [title, price, description, id], function(err, data) {  
}}});
```

Thực hiện update

Tạo Function hoặc Procedure

```
Create procedure addProduct(var1 type, var2 type,...)
Begin
    code insert, update, delete
end
```


Gọi Procedure trong NodeJS

```
Var sql="SET @var1=?;SET var2=?;...; CALL  
addProduct(@var1,@var2,...);"  
db.query(sql, [value1, value2, ...])
```

Cài đặt

```
npm install express --save
```

```
npm install body-parser --save
```

```
npm install mysql --save
```

```
npm install --save-dev nodemon
```

Create và store

```
app.get('/users/create',(req, res) => {  
  res.render('users/create');  
});  
app.post('/users',(req, res) => {  
  let name = req.body.name;  
  let data = {name: name, email: req.body.email};  
  let sql = "INSERT INTO nodejs.users SET ?";  
  let query = db.query(sql, data, (err, results) => {  
    if(err) throw err;  
    // res.redirect(`/users`);  
    console.log('Create success.');  });  
});
```

demo

- ☑ MySQL?
- ☑ Cài đặt và sử dụng trong nodejs
- ☑ Truy vấn dữ liệu mySql với nodejs
- ☑ Thao tác trên mySQL



thank
you!