



DỰ ÁN MẪU (TKTW)

BÀI 3: THỰC HIỆN DỰ ÁN 1

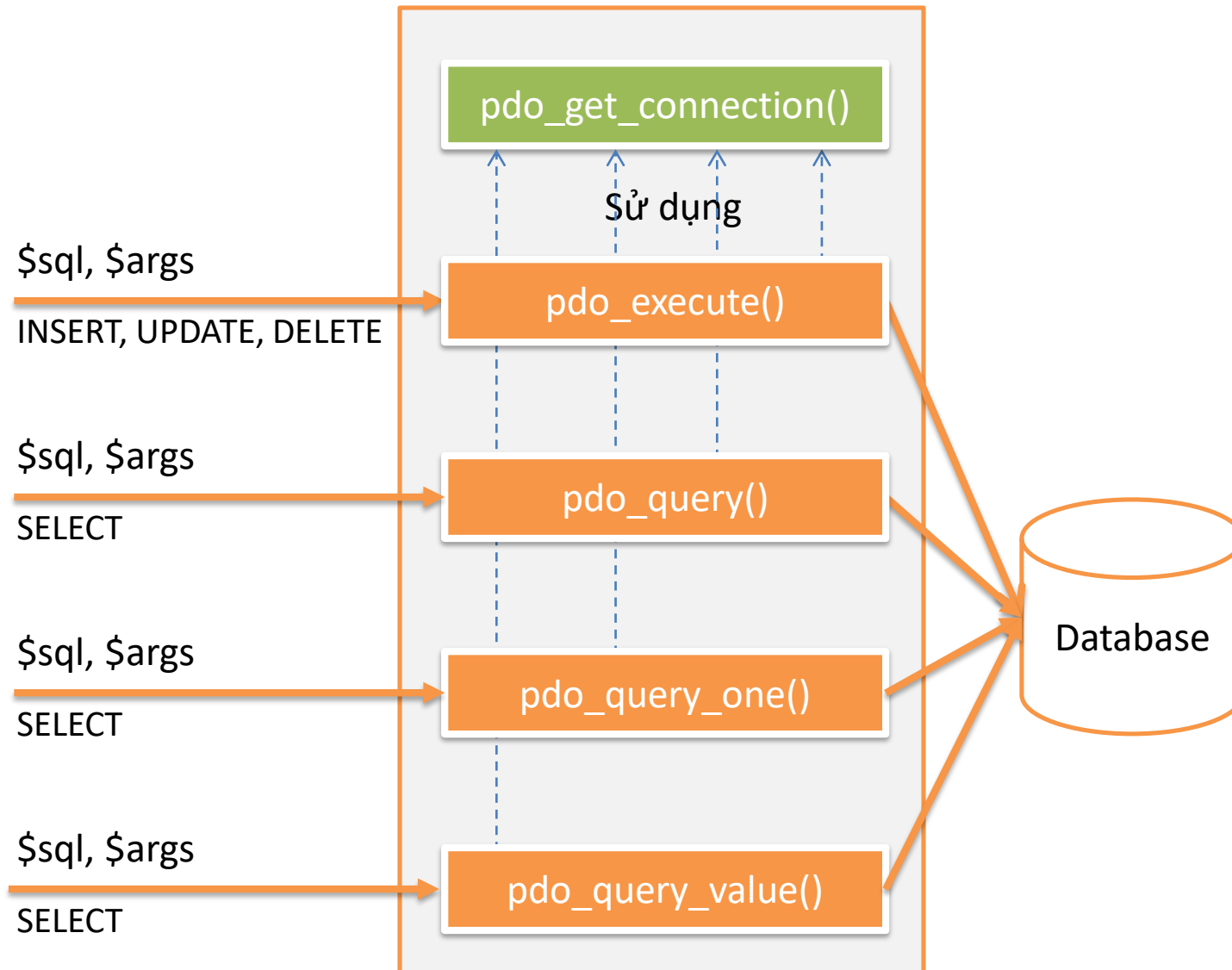
3.2. XÂY DỰNG THƯ VIỆN PDO

- 📖 Ôn lại kiến thức lập trình PDO
- 📖 **Xây dựng thư viện tiện ích PDO**
- 📖 Lập trình CSDL theo mô hình DAO
- 📖 Xây dựng DAO làm việc với loại
- 📖 Xây dựng DAO cho các bảng khác



Xây dựng thư viện PDO

- ❑ Khi làm việc với CSDL chúng ta thường thực hiện các hành động
 - ❖ Mở kết nối
 - ❖ Thao tác dữ liệu với câu lệnh INSERT, UPDATE, DELETE
 - ❖ Truy vấn nhiều bản ghi
 - ❖ Truy vấn 1 bản ghi
 - ❖ Truy vấn 1 giá trị
- ❑ Thư viện pdo chứa các phương thức nhằm hỗ trợ các hành động trên để đơn giản hóa công việc lập trình CSDL



```
<?php
```

```
// Mở kết nối
```

```
function pdo_get_connection(){...}
```

```
// Thực thi câu lệnh INSERT, UPDATE, DELETE
```

```
function pdo_execute($sql){...}
```

```
// Truy vấn nhiều bản ghi
```

```
function pdo_query($sql){...}
```

```
// Truy vấn 1 bản ghi
```

```
function pdo_query_one($sql){...}
```

```
// Truy vấn 1 giá trị
```

```
function pdo_query_value($sql){...}
```

```
function pdo_get_connection() {  
    $dburl = "mysql:host=localhost;dbname=xshop;charset=utf8";  
    $username = 'root';  
    $password = '';  
  
    $conn = new PDO($dburl, $username, $password);  
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);  
    return $conn;  
}
```

`$conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION)`

```
function pdo_execute($sql){  
    $sql_args = array_slice(func_get_args(), 1);  
    try{  
        $conn = pdo_get_connection();  
        $stmt = $conn->prepare($sql);  
        $stmt->execute($sql_args);  
    }  
    catch(PDOException $e){  
        throw $e;  
    }  
    finally{  
        unset($conn);  
    }  
}
```

array_slice(*func_get_args*(), 1)


```
function pdo_query($sql){  
    $sql_args = array_slice(func_get_args(), 1);  
    try{  
        $conn = pdo_get_connection();  
        $stmt = $conn->prepare($sql);  
        $stmt->execute($sql_args);  
        $rows = $stmt->fetchAll();  
        return $rows;  
    }  
    catch(PDOException $e){  
        throw $e;  
    }  
    finally{  
        unset($conn);  
    }  
}
```

\$stmt->fetchAll()

```
function pdo_query_one($sql){  
    $sql_args = array_slice(func_get_args(), 1);  
    try{  
        $conn = pdo_get_connection();  
        $stmt = $conn->prepare($sql);  
        $stmt->execute($sql_args);  
        $row = $stmt->fetch(PDO::FETCH_ASSOC);  
        return $row;  
    }  
    catch(PDOException $e){  
        throw $e;  
    }  
    finally{  
        unset($conn);  
    }  
}
```

`$stmt->fetch(PDO::FETCH_ASSOC)`

```
function pdo_query_value($sql){  
    $sql_args = array_slice(func_get_args(), 1);  
    try{  
        $conn = pdo_get_connection();  
        $stmt = $conn->prepare($sql);  
        $stmt->execute($sql_args);  
        $row = $stmt->fetch(PDO::FETCH_ASSOC);  
        return array_values($row)[0];  
    }  
    catch(PDOException $e){  
        throw $e;  
    }  
    finally{  
        unset($conn);  
    }  
}
```

array_values(\$row)[0]

❑ Thêm mới loại

```
$sql = "INSERT INTO loai(ten_loai) VALUES(?)";
```

```
pdo_execute($sql, $ten_loai);
```

❑ Cập nhật loại

```
$sql = "UPDATE loai SET ten_loai=? WHERE ma_loai=?";
```

```
pdo_execute($sql, $ten_loai, $ma_loai);
```

❑ Xóa loại

```
$sql = "DELETE FROM loai WHERE ma_loai=?";
```

```
pdo_execute($sql, $ma_loai);
```

❑ Truy vấn tất cả các loại

```
$sql = "SELECT * FROM loai";
```

```
$rows = pdo_query($sql);
```

❑ Truy vấn 1 loại

```
$sql = "SELECT * FROM loai WHERE ma_loai=?";
```

```
$row = pdo_query_one($sql, $ma_loai);
```

❑ Đếm số loại

```
$sql = "SELECT count(*) FROM loai WHERE ma_loai=?";
```

```
$count = pdo_query_value($sql, $ma_loai) ;
```

Demo 3.2. Xây dựng các hàm thư viện PDO