

```

<?php
// MODUL 1: Pengenalan Pemrograman Berorientasi Objek (PBO)
abstract class Pasien {
    protected $id;
    private $nama;
    private $umur;

    public function setNama($nama) {
        if(empty($nama)) throw new Exception("Nama tidak boleh kosong!");
        $this->nama = $nama;
    }

    public function getNama() {
        return $this->nama;
    }

    public function setUmur($umur) {
        if($umur < 0) throw new Exception("Umur tidak valid!");
        $this->umur = $umur;
    }

    public function getUmur() {
        return $this->umur;
    }

    public function getId() {
        return $this->id;
    }

// MODUL 2: Dasar Pemrograman Berbasis Objek PHP
    public function __construct($nama, $umur, $id = null) {
        $this->nama = $nama;
        $this->umur = $umur;
        $this->id = $id;
    }

    abstract public function getTipePasien();

    public function __destruct() {
    }
}

// MODUL 3: Pewarisan (Inheritance)
class PasienUmum extends Pasien {
    private $penyakit;
}

```

```

public function __construct($nama, $umur, $penyakit, $id = null) {
    parent::__construct($nama, $umur, $id);
    $this->penyakit = $penyakit;
}

public function getPenyakit() {
    return $this->penyakit;
}

// MODUL 6: Polimorfisme
public function getTipePasien() {
    return "Pasien Umum (Rawat Jalan)";
}
}

// MODUL 4: Modifier dalam PHP
final class Config {
    public static $dsn = "mysql:host=localhost;dbname=rs_manajemen";
    public static $user = "root";
    public static $pass = "";
}

// MODUL 7: Abstraksi dalam PHP
interface OperasiCRUD {
    public function store($nama, $umur, $penyakit);
    public function index();
    public function destroy($id);
}

// MODUL 8: Namespace dan Autoloading dalam PHP
spl_autoload_register(function ($className) {
    $file = str_replace('\\', DIRECTORY_SEPARATOR, $className) . '.php';
    if (file_exists($file)) {
        require_once $file;
    }
});

// MODUL 9: Penanganan Kesalahan dengan Exception
class Connection {
    protected $conn;

    public function __construct() {
        try {
            // MODUL 11: Menggunakan Driver PDO untuk koneksi ke database MySQL

```

```

        $this->conn = new PDO(Config::$dsn, Config::$user, Config::$pass);
        $this->conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    } catch (PDOException $e) {
        die("<div style='color:red; font-weight:bold; padding:10px;
background:#fff3f3;'>Koneksi Database Gagal: " . $e->getMessage() . "</div>");
    }
}
}

```

// MODUL 10: Advanced OOP Features

```
class PasienController extends Connection implements OperasiCRUD {
```

// MODUL 11: Operasi Database CRUD dengan PDO

// 1. CREATE: Menyimpan Data Baru ke Database

```

public function store($nama, $umur, $penyakit) {
    try {
        $pasien = new PasienUmum($nama, $umur, $penyakit); // Instansiasi
Objek Baru (Modul 1)
        $tipe = $pasien->getTipePasien(); // Memanfaatkan Polimorfisme (Modul
6)

```

```

        $stmt = $this->conn->prepare("INSERT INTO pasien (nama, umur,
penyakit, tipe_pasien) VALUES (:nama, :umur, :penyakit, :tipe)");

```

```

        $stmt->bindParam(':nama', $nama);
        $stmt->bindParam(':umur', $umur);
        $stmt->bindParam(':penyakit', $penyakit);
        $stmt->bindParam(':tipe', $tipe);

```

```

        return $stmt->execute();

```

```

    } catch (PDOException $e) {
        echo "Gagal menyimpan data: " . $e->getMessage();
    }
}

```

// 2. READ: Mengambil Seluruh Data dari Database

```

public function index() {
    try {
        $stmt = $this->conn->prepare("SELECT * FROM pasien ORDER BY id
DESC");
        $stmt->execute();

        $results = $stmt->fetchAll(PDO::FETCH_ASSOC);
        $data = [];

```

```

        foreach ($results as $row) {
            $data[] = new PasienUmum($row['nama'], $row['umur'],
$row['penyakit'], $row['id']);
        }
        return $data;
    } catch (PDOException $e) {
        return [];
    }
}

// 3. DELETE: Menghapus Data Berdasarkan ID Pasien
public function destroy($id) {
    try {
        $stmt = $this->conn->prepare("DELETE FROM pasien WHERE id = :id");
        $stmt->bindParam(':id', $id);
        return $stmt->execute();
    } catch (PDOException $e) {
        echo "Gagal menghapus data: " . $e->getMessage();
    }
}
}

// LOGIKA OPERASIONAL ANTARMUKA PENGGUNA
$controller = new PasienController();

if (isset($_POST['tambah'])) {
    $controller->store($_POST['nama'], $_POST['umur'], $_POST['penyakit']);
    header("Location: index.php");
    exit();
}

if (isset($_GET['hapus'])) {
    $controller->destroy($_GET['hapus']);
    header("Location: index.php");
    exit();
}

$daftarPasien = $controller->index();
?>

<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">

```

```

<title>Sistem Manajemen Rumah Sakit</title>
<style>
  body { font-family: 'Segoe UI', Arial, sans-serif; margin: 0; padding:
30px; background-color: #f4f7f6; }
  .container { max-width: 1000px; margin: auto; background: white; padding:
25px; box-shadow: 0 4px 15px rgba(0,0,0,0.05); border-radius: 8px; }
  h2 { color: #2c3e50; text-align: center; margin-bottom: 25px; border-
bottom: 2px solid #3498db; padding-bottom: 10px; }
  .grid { display: flex; gap: 25px; flex-wrap: wrap; }
  .col-form { flex: 1; min-width: 300px; background: #f8f9fa; padding:
20px; border-radius: 6px; border: 1px solid #e9ecef; }
  .col-table { flex: 2; min-width: 500px; }
  h3 { margin-top: 0; color: #34495e; }
  label { font-weight: bold; display: block; margin-top: 10px; color: #555;
}
  input[type="text"], input[type="number"] { padding: 10px; margin: 5px 0
15px 0; width: 100%; display: block; box-sizing: border-box; border: 1px solid
#ccc; border-radius: 4px; }
  button { background: #2ecc71; color: white; border: none; padding: 12px;
width: 100%; cursor: pointer; font-weight: bold; font-size: 15px; border-radius:
4px; transition: 0.2s; }
  button:hover { background: #27ae60; }
  table { width: 100%; border-collapse: collapse; margin-top: 10px;
background: #fff; }
  table, th, td { border: 1px solid #dee2e6; }
  th, td { padding: 12px; text-align: left; }
  th { background-color: #3498db; color: white; }
  tr:nth-child(even) { background-color: #f9f9f9; }
  .btn-delete { background: #e74c3c; color: white; padding: 6px 12px; text-
decoration: none; border-radius: 3px; font-size: 13px; font-weight: bold; }
  .btn-delete:hover { background: #c0392b; }
  .badge { background: #e6f7ff; color: #1890ff; padding: 4px 8px; border-
radius: 4px; border: 1px solid #91d5ff; font-size: 12px; font-weight: bold; }
</style>
</head>
<body>

<div class="container">
  <h2>Sistem Manajemen Rumah Sakit</h2>

  <div class="grid">
    <div class="col-form">
      <h3>Pendaftaran Pasien</h3>
      <form action="" method="POST">
        <label>Nama Pasien</label>

```

```

                <input type="text" name="nama" required placeholder="Contoh: Rian
Hidayat">

                <label>Umur (Tahun)</label>
                <input type="number" name="umur" required placeholder="Contoh:
29">

                <label>Keluhan / Penyakit</label>
                <input type="text" name="penyakit" required placeholder="Contoh:
Asma / Sesak Nafas">

                <button type="submit" name="tambah">Daftarkan Pasien</button>
            </form>
        </div>

        <div class="col-table">
            <h3>Daftar Pasien Aktif</h3>
            <table>
                <thead>
                    <tr>
                        <th>ID</th>
                        <th>Nama Pasien</th>
                        <th>Umur</th>
                        <th>Diagnosa Penyakit</th>
                        <th>Kategori</th>
                        <th>Aksi</th>
                    </tr>
                </thead>
                <tbody>
                    <?php if(empty($daftarPasien)): ?>
                        <tr><td colspan="6" style="text-align:center; color:
#999;">Belum ada data pasien di dalam database.</td></tr>
                    <?php else: ?>
                        <?php foreach($daftarPasien as $p): ?>
                            <tr>
                                <td><?= $p->getId(); ?></td>
                                <td><strong><?= htmlspecialchars($p->getNama());
?></strong></td>
                                <td><?= $p->getUmur(); ?> Tahun</td>
                                <td><?= htmlspecialchars($p->getPenyakit()); ?></td>
                                <td><span class="badge"><?= $p->getTipePasien();
?></span></td>
                                <td>

```

```
                <a href="index.php?hapus=<?=$p->getId(); ?>"
class="btn-delete" onclick="return confirm('Apakah Anda yakin ingin menghapus
data pasien ini?')">Hapus</a>
                </td>
            </tr>
        <?php endforeach; ?>
    <?php endif; ?>
</tbody>
</table>
</div>
</div>
</div>
</div>
</body>
</html>
```