**JAVA**

1. int i = 1, j = 10; do { if (i++ > --j){

continue;

}

}while(i<5);

System.out.println(“i = “+i+” and j = “+j);

**Berapa hasilnya dari script diatas?**

* + 1. i = 6 and j = 5
    2. i = 5 and j = 5
    3. i = 6 and j = 5
    4. **i = 5 and j = 6**
    5. i = 6 and j = 6

1. class Test { private Demo d; void start() {

d = new Demo(); this.takeDemo(d);

}

void takeDemo(Demo demo) {

demo = null; demo = new Demo();

}

}

**When is the Demo object, created on line 3, eligible for garbage collection?** A. After line 5.

* 1. After line 9.
  2. After the start() method completes.
  3. When the takeDemo() method completes.
  4. **When the instance running this code is made eligible for garbage collection.**

1. interface Animal { void soundOff();

}

class Elephant implements Animal {

public void soundOff() {

System.out.println(“Trumpet”);

}

}

class Lion implements Animal {

public void soundOff() {

System.out.println(“Roar”);

} } class Alpha1 {

static Animal get( String choice ) { if ( choice.equalsIgnoreCase( “meat eater” )) { return new Lion();

} else { return new Elephant();

}

}

}

**Which compiles?**

1. new Animal().soundOff();
2. Elephant e = new Alpha1();
3. Lion 1 = Alpha.get(“meat eater”);
4. **new Alpha1().get(“veggie”).soundOff();**
5. class A { A() { }

}

class B extends A {

}

**Which two statements are true? (Choose two)** A. A. Class B’s constructor is public.

* 1. **Class B’s constructor has no arguments.**
  2. Class B’s constructor includes a call to this().
  3. **Class B’s constructor includes a call to super().**

1. int x = 3; int y = 1; if (x = y) {

System.out.println(“x = “ + x);

}

**Apa hasil dari script diatas?**

* 1. x = 1
  2. x = 3
  3. **Compilation fails.**
  4. The code runs with no output.
  5. An exception is thrown at runtime.

1. Apa yang Anda ketahui tentang method dan attributedi java?

**Jawab:**

Yang saya ketahui tentang method di java yaiut, jika method adalah sebuah fungsi dalam sebuah kelas yang berisi logic untuk menjalankan suatu tugas, nah sedangkan atribut itu adalah sebuah variabel yang menyimpan data di dalam class

1. Apa yang Anda ketahui tentang encapsulation dan inheritance?

**Jawab:**

Yang saya ketahui untuk Encapsulation itu adalah membuat suatu data tidak bisa di akses dari luar kelas secara langsung, sedangkan untuk Inheritance seperti namanya yaitu mewarisi suatu data ke kelas lain

# MySQL

1. Apa yang Anda ketahui tentang right join, left join, inner join and outer join? Dan tuliskan contoh Query-nya.

**Tabel pelanggan**

|  |  |
| --- | --- |
| **id\_pelanggan** | **nama** |
| 1 | Muhamad |
| 2 | Faqih |
| 3 | Azhar |

**Tabel pesanan**

|  |  |  |
| --- | --- | --- |
| **id\_pesanan** | **id\_pelanggan** | **produk** |
| 1 | 1 | Anggur |
| 2 | 2 | Alpukat |

**Right Join**

Menggabungkan dan menampilkan semua data dari tabel kanan, misal tabel Pelanggan | Pesanan , jika Pelanggan right join maka akan menampilkan Pelanggan | Pesanan sesuai relasinya, jika tidak ada relasi maka akan tampil NULL

**SELECT pelanggan.nama, pesanan.produk FROM pelanggan RIGHT JOIN pesanan ON pelanggan.id\_pelanggan = pesanan.id\_pelanggan;**

|  |  |
| --- | --- |
| **name** | **produk** |
| Muhamad | Anggur |
| Faqih | Alpukat |

**Left Join**

Menggabungkan dan menampilkan semua data dari tabel kiri, misal tabel A | B, jika B left join maka akan menampilkan B | A sesuai relasinya, jika tidak ada relasi maka akan tampil NULL

**SELECT pelanggan.nama, pesanan.produk FROM pelanggan LEFT JOIN pesanan ON pelanggan.id\_pelanggan = pesanan.id\_pelanggan;**

|  |  |
| --- | --- |
| **name** | **produk** |
| Muhamad | Anggur |
| Faqih | Alpukat |
| Citra | NULL |

**Inner Join**

Menggabungkan dan menampilkan hanya data yang berelasi, jika tidak maka tidaka akn ditampilkan

**SELECT pelanggan.nama, pesanan.produk FROM pelanggan INNER JOIN pesanan ON pelanggan.id\_pelanggan = pesanan.id\_pelanggan;**

|  |  |
| --- | --- |
| **name** | **produk** |
| Muhamad | Anggur |
| Faqih | Alpukat |

**Outer Join**

Menggabungkan dan menampilkan semua data, baik ada relasi ataupun tidak, jika tidak akan ditampilkan NULL

**SELECT pelanggan.nama, pesanan.produk FROM pelanggan FULL OUTER JOIN pesanan ON pelanggan.id\_pelanggan = pesanan.id\_pelanggan;**

|  |  |
| --- | --- |
| **name** | **produk** |
| Muhamad | Anggur |
| Faqih | Alpukat |
| Citra | NULL |

1. Tuliskan Query untuk menampilkan data dari sebuah table.

**Jawab:**

**SELECT \* FROM table\_name;**

# PHP

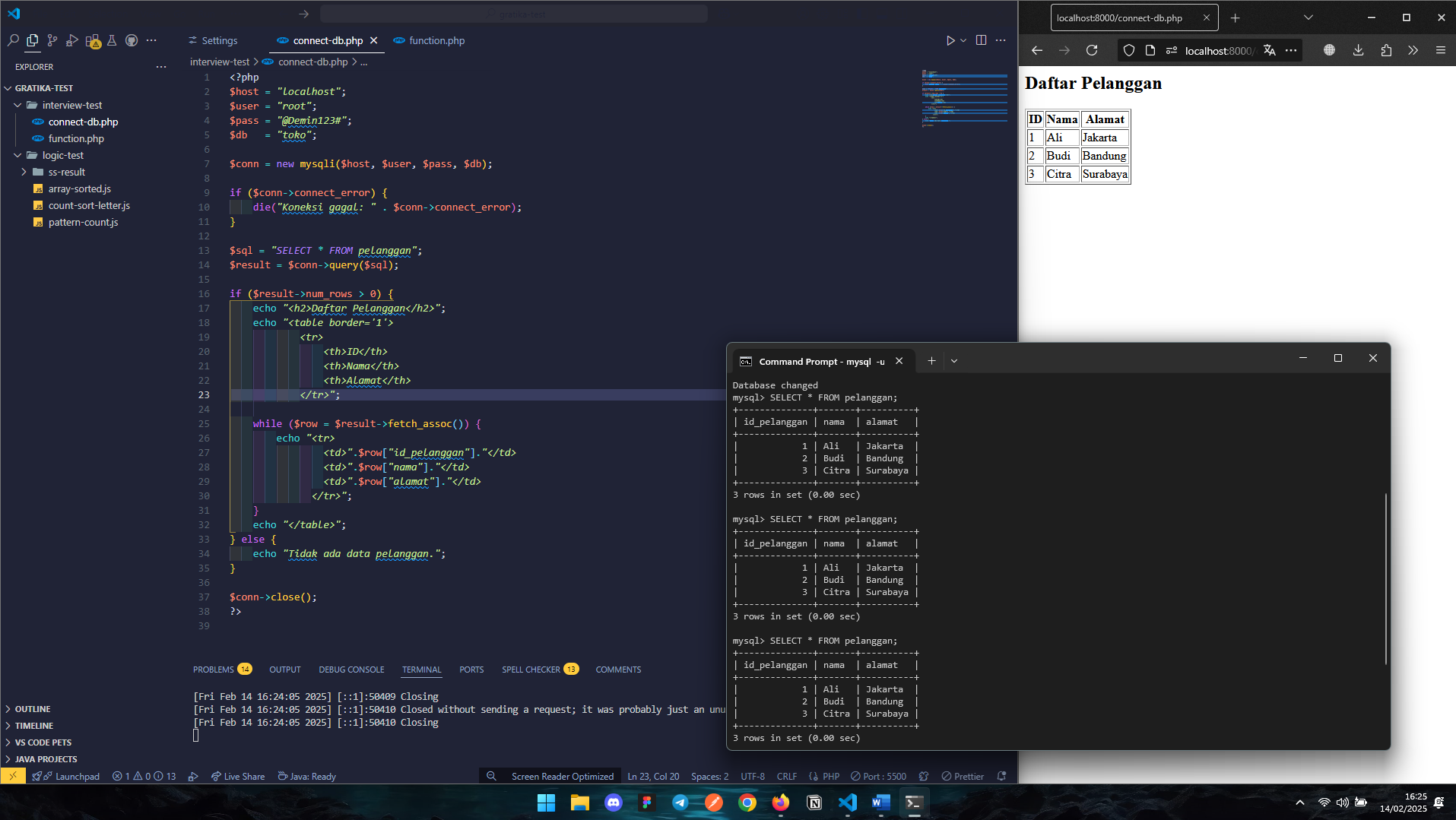
1. Bagaimana cara membuat fungsi di PHP dan bagaimana cara untuk memanggilnya?

**Jawab:**

Ketika membuat fungsi di PHP, bisa di awali dengan function kemudian di ikuti nama fungsinya, diikutin tanda kurung () dan kurung kurawal {}, kemduian ketika akan memanggilnya cukup sebutkan nama fungsinya saja, contohnya seperti ini



1. Tuliskan sebuah script untuk menghubungkan PHP dan MySQL dan tampilkan hasil Querynya di PHP



1. Framework PHP apa saja yang anda ketahui? dan apa kelebihannya setiap framework PHP yang anda ketahui?

**Jawab:**

Framework PHP yang saya ketahui dan pernah mencobanya adalah Laravel dan Code Igniter, untuk Laravel saya rasa sangat powerfull dan banyak fiturnya, kemudian komunitas dan dokumentasinya juga mudah ditemukan untuk dipelajari. Kemudian, untuk CI saya tidak lama mencobanya, namun dari yang say abaca CI disebut lebih ringan dan cepat. Selain itu, saya juga mengetahui YII, namun hanya sebatas nama saja belum pernah mencobanya.

1. Untuk mendukung tampilan Web selain php, pemograman/framework apa saja yang anda butuhkan? Berikan penjelasan kenapa anda menggunakan/membutuhkan nya?

**Jawab:**

Untuk mendukung tampilan web, saya rasa pastinya membutuhkan styling dengan CSS dan membuatnya interaktif dengan JavaScript, nah kemudian untuk frameworknya sendiri mungkin untuk saya sendiri lebih nyaman menggunakan TailwindCSS atau SaaS, meskipun ada Bootstrap juga, dan untuk JS saya prefer menggunakan React karena lebih flexible, atau jika ingin mengedepankan SEO bisa menggunakan NextJS. Nah kemudian untuk animasi jika memang lebih menarik, bisa menggunakan seperti framer-motion, GSAP, atau mungkin transisi sederhana dari CSS itu sendiri.

**Studi Kasus:**

Terdapat contoh nota dari gambar bawah berikut:

* 1. Buatlah LRS (Logical Record Structure) berdasarkan nota tersebut.

*\*Full Query in GitHub*

**LRS (Logical Record Structure)**

**faskes**

- id(PK)

- nama\_dokter

- alamat

- signature

**pasien**

- id(PK)

- nama\_pasien

- alamat

- penanggung\_jawab

- jenis\_kelamin

- umur

- tanggal\_lahir

- signature

**obat**

- id(PK)

- nama\_obat

- harga\_satuan

**transaksi**

- id(PK)

- tanggal

- id\_faskes(FK)

- id\_pasien (FK)

- total\_biaya

**detail\_transaksi**

- id(PK)

- id\_transaksi(FK)

- id\_obat(FK)

- qty

- subtotal

- total

* 1. Buatlah tampilan program tersebut berdasarkan LRS yang telah dibuat.
  2. Tulis Query berdasarkan LRS yang telah dibuat
     1. Tampilkan data pelanggan yang mempunya kelamin perempuan, umur antara 19 sampai 30, dan pemeriksaan pada bulan Agustus 2015.

**SELECT DISTINCT p.id, p.nama\_pasien, p.jenis\_kelamin, p.umur, t.tanggal**

**FROM pasien p**

**JOIN transaksi t ON p.id = t.id\_pasien**

**WHERE p.jenis\_kelamin = 'P'**

**AND p.umur BETWEEN 19 AND 30**

**AND t.tanggal BETWEEN '2015-08-01' AND '2015-08-31';**

* + 1. Tampilkan data semua dokter yang mempunyai transaksi dengan pasien ataupun tidak selama setahun di 2015

**SELECT f.id, f.nama\_dokter, f.alamat**

**FROM faskes f**

**LEFT JOIN transaksi t ON f.id = t.id\_faskes AND YEAR(t.tanggal) = 2015;**

* + 1. Hitung jumlah obat dan total uang per-obat selama bulan Agustus sampai Desember 2015

**SELECT**

**o.nama\_obat,**

**SUM(d.qty) AS total\_jumlah\_obat,**

**SUM(d.subtotal) AS total\_pendapatan**

**FROM detail\_transaksi d**

**JOIN transaksi t ON d.id\_transaksi = t.id**

**JOIN obat o ON d.id\_obat = o.id**

**WHERE t.tanggal BETWEEN '2015-08-01' AND '2015-12-31'**

**GROUP BY o.nama\_obat;**

* + 1. Tampilkan 10 Jenis obat apa saja yang paling banyak digunakan selama tahun 2015

**SELECT**

**o.nama\_obat,**

**SUM(d.qty) AS total\_jumlah\_obat**

**FROM detail\_transaksi d**

**JOIN transaksi t ON d.id\_transaksi = t.id**

**JOIN obat o ON d.id\_obat = o.id**

**WHERE YEAR(t.tanggal) = 2015**

**GROUP BY o.nama\_obat**

**ORDER BY total\_jumlah\_obat DESC**

**LIMIT 10;**

* + 1. Tampilkan data pelanggan Jika berumur di bawah 18 tahun “Anak-anak”, jika berumur 18 tahun sampai 30 “Dewasa”, dan jika lebih dari 30 “Orang tua”.

**SELECT**

**p.id,**

**p.nama\_pasien,**

**p.umur,**

**CASE**

**WHEN p.umur < 18 THEN 'Anak-anak'**

**WHEN p.umur BETWEEN 18 AND 30 THEN 'Dewasa'**

**ELSE 'Orang tua'**

**END AS kategori\_umur**

**FROM pasien p;**

