

1. Jelaskan secara sederhana mengenai konsep pemrograman OOP!

Pemrograman Berorientasi Objek (Object-Oriented Programming atau OOP) adalah paradigma pemrograman yang berfokus pada pengorganisasian dan strukturisasi kode berdasarkan objek-objek yang merepresentasikan entitas dalam dunia nyata. Konsep OOP mencakup tiga pilar utama: encapsulation, inheritance, dan polymorphism.

2. . Jelaskan secara sederhana istilah-istilah berikut:

- a. Class
- b. Trait
- c. Interface
- d. Abstract Class
- e. Static Function

- A. Class (Kelas): Kelas adalah sebuah blueprint atau cetak biru untuk menciptakan objek. Ini mendefinisikan struktur dan perilaku yang dimiliki oleh objek-objek yang akan dibuat berdasarkan kelas tersebut. Dalam kelas, Anda dapat mendefinisikan atribut (variabel) dan metode (fungsi) yang berkaitan dengan objek yang akan diwujudkan dari kelas tersebut.
- B. Trait dalam pemrograman merujuk pada konsep atau fitur dalam beberapa bahasa pemrograman yang digunakan untuk menyimpan dan mengelompokkan fungsi atau metode bersama-sama, sehingga dapat digunakan oleh berbagai kelas atau objek tanpa harus diwariskan dari kelas lain.
- C. Interface (dalam bahasa pemrograman) adalah sebuah konsep yang mendefinisikan kumpulan metode yang harus diimplementasikan oleh kelas-kelas yang menggunakan interface tersebut. Interface menyediakan kontrak atau blueprint untuk kelas-kelas yang ingin mengikuti aturan yang ditetapkan oleh interface tersebut.
- D. Abstract Class adalah sebuah konsep dalam pemrograman berorientasi objek (OOP) yang berfungsi sebagai kerangka kerja untuk kelas-kelas turunannya. Abstract class tidak dapat diinstansiasi menjadi objek secara langsung, tetapi berperan sebagai "kelas dasar" yang menyediakan definisi metode-metode abstrak dan/atau metode dengan implementasi konkret yang dapat diwarisi oleh kelas-kelas turunannya (subclass).
- E. Static Function (fungsi statis) adalah sebuah fungsi dalam pemrograman yang terkait dengan kelas (disebut juga metode statis dalam beberapa bahasa pemrograman). Berbeda dengan metode biasa yang terkait dengan instance objek dari kelas, metode statis tidak memerlukan pembuatan objek dan bisa dipanggil langsung melalui kelas tanpa harus menginstansiasi objek terlebih dahulu.

3. Dengan perkembangan teknologi saat ini, masih bersediakah kamu mengelola dan melakukan maintenance terhadap aplikasi yang telah dibuat sejak lama? Tuliskan alasannya.

Iya Saya Bersedia untuk mengelola dan melakukan maintenance terhadap aplikasi yang telah di buat sejak lama , di karenakan

Keberlanjutan Bisnis: Aplikasi yang telah dibuat sejak lama mungkin masih digunakan oleh banyak pengguna atau perusahaan. Dengan melakukan maintenance, aplikasi dapat terus berfungsi dengan baik dan mendukung kelangsungan operasional bisnis.

Kepatuhan dan Keamanan: Dengan berkembangnya teknologi, standar keamanan dan peraturan yang berlaku juga dapat berubah. Dengan melakukan maintenance, aplikasi dapat diperbarui untuk tetap mematuhi aturan dan standar keamanan terbaru, sehingga risiko keamanan dapat dikelola dengan baik.

Peningkatan Kinerja: Dalam jangka waktu yang lama, aplikasi dapat mengalami performa yang menurun akibat penambahan data atau perubahan lingkungan. Dengan maintenance, kita dapat melakukan optimisasi untuk meningkatkan kinerja aplikasi.

4. Familiar dengan Unit Testing? Jelaskan secara sederhana tentang Unit Testing.

Ya, saya familiar dengan Unit Testing. Unit Testing adalah salah satu praktik dalam pengembangan perangkat lunak yang digunakan untuk menguji setiap unit terkecil dari kode secara terpisah dan otomatis. Unit dalam konteks ini merujuk pada fungsi, metode, atau bagian kecil dari kode yang dapat diuji secara terisolasi.

Tujuan dari Unit Testing adalah untuk memastikan bahwa setiap unit kode berfungsi sesuai dengan yang diharapkan dan tidak menyebabkan kesalahan (bug) ketika digunakan bersama dengan unit lain dalam aplikasi. Unit Testing membantu mengidentifikasi masalah atau bug pada tahap awal pengembangan sehingga memungkinkan pengembang untuk memperbaikinya sebelum integrasi kode dengan bagian lain dari system

5. Manakah diantara keempat role berikut yang mencerminkan kamu? Tuliskan alasannya.

- Front End Developer
- Back End Developer
- Fullstack Developer
- QA Tester

QA Tester , dikarenakan ingin Meningkatkan Reputasi Perusahaan dan Kualitas perangkat lunak yang tinggi adalah salah satu faktor penting untuk membangun reputasi baik bagi perusahaan. Dengan menguji secara menyeluruh dan memastikan aplikasi berfungsi dengan baik, perusahaan dapat membangun kepercayaan dan kepuasan pengguna.