

☐ Ceritakan apa itu OOP, object oriented programming

OOP(Object Oriented Programming) adalah contoh pemrograman berisi data dan metode yang konsepnya didasarkan pada “objek” dengan tujuan untuk meningkatkan fleksibilitas dan perawatan program. OOP mengelompokkan data dan metodenya dalam satu lokasi membuat suatu program lebih mudah untuk dipahami

☐ Di java 8 ada dua komponen untuk implementasi abstraksi lebih baik, yaitu stream dan lambda. Ceritakan apa itu. Pada saat apakah kita membutuhkan stream atau lambda.

Lambda adalah fungsi anonim atau tidak memiliki nama dan tidak terikat pada kelas apapun

Stream adalah operasi sekuensial yang mendukung paralelisme pada itemnya dan memuat waktu eksekusi program lebih cepat cara klasik

Lamda dan stream digunakan untuk mempersingkat fungsi yang memiliki operasi yang sama(paralel)

☐ Apa perbedaan antara interface, abstract kelas.

Interface

- Hanya boleh memiliki satu metode abstrak(statis atau default)
- Mendukung multiple-inheritance
- Hanya memiliki statis dan final variable
- Tidak menyediakan implementasi kelas abstrak
- Membernya secara default bersifat publik

Abstract kelas

- Bisa memiliki metode abstrak dan non-abstark
- Tidak mendukung multiple-inheritance
- Dapat memiliki variabel final, non-final, statis, dan non-statis.
- Dapat menyediakan implementasi antarmuka.
- Dapat memiliki member private, protected dll

☐ Apa itu functional interface, dibutuhkan pada saat apakah?

Functional interface adalah suatu interface yang memilii hanya satu metode abstrak tetapi bisa memiliki beberapa metode default dan mendeklarasikan satu metode kelas objek. Dibutuhkan saat suatu interface hanya menggunakan satu metode abstrak

☐ Terangkan apa itu SOLID principle. Apakah berguna?

SOLID principle adalah prinsip/pola desain yang merupakan kumpulan dari 5 prinsip desain, antara lain: Single Responsibility Principle, Open-Closed Principle, Liskov Substitution Principle, Interface Segregation Principle, and Dependency Inversion Principle, membuat isi program lebih mudah diakses dan dipahami, membantu menghindari code/script yang bertumpuk(busuk), dan menjaga kualitas software yang baik.

Sangat berguna untuk pemeliharaan program

☐ Jika saya mempunyai integer : 1234567 , buat lah fungsi yang keluarannya 7654321 tanpa menggunakan library, atau string proseding.

Jawabannya ada di file sorting.java

☐ ☐ Saya memiliki matrix sebagai berikut:

Buat matrix class yang mempunyai behavior untuk perkalian matrix. Coba untuk menyelesaikan perkalian matrix diatas.

Jawabannya ada di file matrix.java

☐ ☐ Apa itu unit testing. Bedakah dengan integration testing? Apa bedanya?

Unit testing adalah pengujian modul secara terpisah atau individu untuk memastikan code berjalan dengan benar

Integration testing adalah pengujian beberapa modul yang berbeda untuk memastikan akan berfungsi dengan baik saat digabungkan

☐ ☐ Terangkan apa itu TDD.

TDD(Test-Driven Development) adalah proses pengembangan software yang mencakup pengembangan test-first, artinya melakukan testing/pengujian terlebih dahulu sebelum melakukan implementasi. Langkah-langkahnya dapat dilakukan sebagai berikut:

- 1. Memahami persyaratan untuk suatu fitur tertentu**
- 2. Memuat serangkaian tes untuk menguji fitur tersebut tanpa terlalu memperhatikan code(error)**
- 3. Mengembangkan fitur tersebut sampai berjalan dengan baik(tidak error)**
- 4. Memperbaiki code agar lebih mudah dibaca dan dipelihara**