

## TUGAS RESUME

### PBO

Nama = Muhamad Iqbal Ramadhan

NIM = 20210040139

Kelas = TI21G

### Jawab

#### 1. Versi Terbaru Java?

Pada Maret 2022, Java 18 adalah versi Java terbaru yang dirilis. Pada September 2022, Java 19 akan menyusul. Versi dukungan jangka panjang (LTS) terakhir dari Java adalah Java 17, dirilis pada September 2021.

#### Versi Java apa yang harus saya gunakan?

Versi Java yang lebih baru sekarang mengikuti setiap 6 bulan. Jadi, Java 20 dijadwalkan untuk Maret 2023, Java 21 untuk September 2023 dan seterusnya. Di masa lalu, siklus rilis Java jauh lebih lama, hingga 3-5 tahun.

#### Fitur – Fitur terbaru java 18 serta kelebihanannya?

##### - Deprecate finalization for removal

dalam rilis mendatang. Finalizer memiliki kekurangan yang menyebabkan masalah dunia nyata yang signifikan dalam keamanan, kinerja, keandalan, dan pemeliharaan. Ini juga memiliki model pemrograman yang sulit. Finalisasi diaktifkan secara default untuk saat ini tetapi dapat dinonaktifkan untuk memfasilitasi pengujian awal. Ini akan dinonaktifkan secara default dalam rilis fitur dan dihapus sama sekali di rilis berikutnya. Proposal meminta opsi baris perintah untuk menonaktifkan finalisasi dan penghentian semua finalizer dan metode finalisasi di Java API standar. Tujuan proposal termasuk membantu pengembang memahami bahaya finalisasi, mempersiapkan pengembang untuk penghapusan akhirnya, dan menyediakan alat sederhana untuk membantu mendeteksi ketergantungan pada finalisasi. Diperkenalkan di Java 1.0, finalisasi dimaksudkan untuk membantu menghindari kebocoran sumber daya. Kelas dapat mendeklarasikan finalizer — metodeprotected void finalize()— yang tubuhnya melepaskan sumber daya yang mendasarinya. Pengumpul sampah akan menjadwalkan finalizer dari objek yang tidak terjangkau untuk dipanggil sebelum mengambil kembali memori objek; pada gilirannya, finalizemetode ini dapat mengambil tindakan seperti memanggil objekclose. Ini tampak seperti jaring pengaman yang efektif untuk mencegah kebocoran sumber daya, tetapi ada kekurangan termasuk latensi yang tidak dapat diprediksi, dengan waktu yang lama antara saat objek tidak dapat dijangkau dan saat finalizer dipanggil; perilaku yang tidak dibatasi, dengan kode finalizer yang dapat melakukan tindakan apa

pun, termasuk menghidupkan kembali suatu objek dan membuatnya dapat dijangkau kembali; finalizer selalu diaktifkan, tanpa mekanisme pendaftaran eksplisit; dan finalizer dapat berjalan pada utas yang tidak ditentukan dalam urutan yang berubah-ubah. Mengingat masalah dengan finalisasi, pengembang disarankan untuk menggunakan teknik alternatif untuk menghindari kebocoran sumber daya, yaitu pernyataan try-with-resources dan pembersih.

- **For the Internet-address resolution SPI, the proposal is to define an SPI** untuk resolusi alamat host dan nama sehingga InetAddress dapat menggunakan resolver selain dari resolver bawaan platform. Motivasi untuk upaya ini mencakup pengaktifan Project Loom yang lebih baik, untuk konkurensi dan model pemrograman baru di Java, bersama dengan pengintegrasian protokol jaringan baru, penyesuaian, dan pengujian yang memungkinkan. Proposal tidak melibatkan pengembangan resolver alternatif untuk JDK.
- **The reimplementation of core reflection with method handles** diimplementasikan kembali lang.reflect.Method, Constructor, dan Field di atas java.lang.invoke pegangan metode. Memiliki pegangan metode yang berfungsi sebagai mekanisme dasar untuk refleksi akan mengurangi biaya pemeliharaan dan pengembangan baik API java.lang.reflect maupun java.lang.invoke API.
- **Java 19 Coming Soon**
- **Java 20 Coming Soon.**