**LATIHAN/TUGAS**

1. Jelaskan apa itu Jre ? Apa kegunaannya!
2. Jelaskan apa itu JDK?
3. Jelaskan apa itu Java Virtual Machine?
4. Jelaskan tahap membuat projek aplikasi baru dan class baru di netbeans!
5. Berdasarkan jenis , aplikasi apa saja yang dapat dibuat dengan menggunakan pemrograman Java?
6. Berdasarkan platform , platform apa saja pada Java?

**JAWABAN**

1. **JRE (Java Runtime Environment)** adalah lingkungan yang memungkinkan program Java untuk dijalankan di perangkat pengguna. JRE berisi semua komponen yang diperlukan untuk menjalankan aplikasi yang telah dikompilasi menggunakan bahasa pemrograman Java

**Kegunaan JRE:**

* **Menjalankan Aplikasi Java**: JRE digunakan oleh pengguna akhir untuk menjalankan aplikasi yang ditulis dalam bahasa Java. Ini adalah komponen yang diperlukan untuk menjalankan aplikasi Java di perangkat tanpa harus menginstal seluruh kit pengembangan (JDK).
* **Mendukung Pengembangan**: Sementara JDK digunakan oleh pengembang untuk membuat aplikasi, JRE memungkinkan pengujian dan debugging aplikasi dalam lingkungan yang sama dengan yang akan digunakan oleh pengguna akhir.

1. **JDK (Java Development Kit)** adalah alat pengembangan perangkat lunak yang digunakan oleh para pengembang untuk menulis, mengkompilasi, dan menjalankan program Java. JDK adalah paket lengkap yang mencakup segala sesuatu yang dibutuhkan untuk mengembangkan aplikasi Java.
2. **Java Virtual Machine (JVM)** adalah sebuah mesin virtual yang memungkinkan komputer menjalankan program yang ditulis dalam bahasa pemrograman Java. JVM bertanggung jawab untuk mengeksekusi bytecode Java, yang merupakan hasil kompilasi dari kode sumber Java.
3. **Membuat Projek Aplikasi Baru:**
4. **Buka NetBeans**: Jalankan NetBeans di komputer Anda.
5. **Pilih Menu "File" > "New Project..."**: Di menu utama, pilih "File", kemudian klik "New Project...".
6. **Pilih Jenis Proyek**: Di jendela "New Project", pilih kategori "Java" dan jenis proyek "Java Application". Klik "Next".
7. **Nama dan Lokasi Proyek**: Beri nama proyek Anda pada kolom "Project Name". Tentukan lokasi penyimpanan proyek pada "Project Location".
8. **Main Class (Opsional)**: Jika Anda ingin langsung membuat kelas utama, pastikan kotak "Create Main Class" dicentang, dan beri nama kelas utama (misalnya, Main). Klik "Finish" untuk membuat proyek.

**Membuat Class Baru di Dalam Proyek:**

1. **Pilih Package**: Di "Projects" window (biasanya terletak di sebelah kiri), buka folder "Source Packages", lalu klik kanan pada package tempat Anda ingin membuat kelas baru (misalnya, myproject).
2. **Buat Kelas Baru**: Klik kanan pada package yang dipilih, lalu pilih "New" > "Java Class...".
3. **Nama Kelas**: Di jendela "New Java Class", masukkan nama kelas yang diinginkan pada kolom "Class Name".
4. **Klik "Finish"**: Setelah nama kelas diisi, klik "Finish" untuk membuat kelas baru.
5. **Aplikasi Desktop**:

* **GUI Applications**: Java memungkinkan pembuatan aplikasi desktop dengan antarmuka pengguna grafis (GUI) menggunakan toolkit seperti **Swing** dan **JavaFX**. Contoh: aplikasi perkantoran, alat desain, dan editor teks.

**Aplikasi Web**:

* **Server-side Applications**: Java digunakan secara luas dalam pengembangan aplikasi web berbasis server menggunakan teknologi seperti **Java Servlets**, **JSP (JavaServer Pages)**, dan **Spring Framework**. Contoh: sistem manajemen konten (CMS), portal web, dan layanan e-commerce.

**Aplikasi Mobile**:

* **Android Applications**: Java adalah bahasa utama untuk pengembangan aplikasi Android, menggunakan **Android SDK**. Contoh: aplikasi sosial media, game mobile, dan aplikasi utilitas.

**Aplikasi Game**:

* **Game Development**: Java dapat digunakan untuk membuat game, baik untuk desktop maupun mobile. Engine seperti **LibGDX** memfasilitasi pengembangan game 2D dan 3D dengan Java. Contoh: game puzzle, game platformer, dan game edukasi.

**Aplikasi Keamanan**:

* **Security Applications**: Java juga digunakan untuk membuat aplikasi yang memerlukan tingkat keamanan tinggi, seperti sistem enkripsi, autentikasi, dan aplikasi perbankan.

1. **Java SE (Standard Edition)**:

* **Deskripsi**: Java SE menyediakan lingkungan inti untuk pengembangan aplikasi desktop dan server. Ini mencakup berbagai API (Application Programming Interface) dasar seperti Collections, I/O, Networking, GUI (Swing dan JavaFX), dan lainnya.
* **Penggunaan**: Digunakan untuk membangun aplikasi desktop, alat pengembangan, dan aplikasi server non-web.

**Java EE (Enterprise Edition)**:

* **Deskripsi**: Java EE, kini dikenal sebagai Jakarta EE, adalah platform yang diperluas dari Java SE yang mencakup API dan runtime untuk pengembangan aplikasi berskala besar dan kompleks, khususnya aplikasi berbasis web dan enterprise
* **Penggunaan**: Digunakan untuk membangun aplikasi enterprise yang kuat seperti aplikasi bisnis, sistem manajemen pelanggan (CRM), dan layanan web.

**Java ME (Micro Edition)**:

* **Deskripsi**: Java ME adalah platform yang disesuaikan untuk perangkat dengan sumber daya terbatas, seperti perangkat mobile, perangkat embedded, dan IoT. Java ME menyediakan subset API yang lebih ringan dari Java SE, dengan tambahan API khusus untuk perangkat embedded.
* **Penggunaan**: Digunakan dalam pengembangan aplikasi untuk perangkat mobile, perangkat elektronik konsumen, dan sistem embedded.