

Conceptos importantes en Kotlin relacionados con procesos, hilos, red y seguridad

Procesos

- `ProcessBuilder`: Clase que permite crear y controlar procesos en Kotlin.

[crayon-6a9d69707d1dc681395218/]

- `Runtime`: Clase que permite interactuar con el entorno de ejecución y controlar procesos en tiempo de ejecución.

[crayon-6a9d69707d1e4988983872/]

- `Process`: Representa un proceso en ejecución y proporciona métodos para controlarlo.

[crayon-6a9d69707d1e7137648516/]

- `ProcessHandle`: Obtener información sobre procesos en ejecución y controlarlos.

[crayon-6a9d69707d1e9174964886/]

Hilos

- `Thread`: Clase que permite crear y controlar hilos en

Kotlin.

[crayon-6a9d69707d1ef738956975/]

- **Runnable:** Interfaz funcional que define un hilo de ejecución en Kotlin.

[crayon-6a9d69707d1f1254166304/]

- **Coroutine:** Mecanismo de concurrencia que permite realizar tareas asíncronas y no bloqueantes en Kotlin.

[crayon-6a9d69707d1f3189636258/]

- **ExecutorService:** Esta interfaz permite administrar hilos y ejecutar tareas de manera concurrente.

[crayon-6a9d69707d1f5074151383/]

- **CompletableFuture:** Clase que facilita la programación asíncrona y la combinación de múltiples futuros en Kotlin.

[crayon-6a9d69707d1fa871103244/]

Red

- **URL:** Clase que permite trabajar con URL y realizar operaciones de red como abrir conexiones HTTP.

[crayon-6a9d69707d1fc977272033/]

- **URLConnection:** Clase abstracta que establece una conexión a un recurso a través de una URL.

[crayon-6a9d69707d1fe821862854/]

- **Socket**: Clase que representa un socket de red y permite la comunicación entre procesos a través de la red.

[crayon-6a9d69707d200727866826/]

- **SocketChannel**: Una clase para operaciones de socket no bloqueantes en Kotlin.

[crayon-6a9d69707d202911750667/]

- **DatagramSocket**: Para trabajar con datagramas y sockets UDP en Kotlin.

[crayon-6a9d69707d204082224475/]

Seguridad

- **Cipher**: Clase que proporciona funcionalidad de cifrado y descifrado para encriptar y desencriptar datos.

[crayon-6a9d69707d208766635511/]

- **SecureRandom**: Clase que genera números aleatorios criptográficamente seguros.

[crayon-6a9d69707d20a013507598/]

- **KeyStore**: Clase que almacena claves criptográficas y certificados de seguridad.

[crayon-6a9d69707d20c407523164/]

- `KeyPairGenerator`: Para generar pares de claves criptográficas (pública y privada) en Kotlin.

[crayon-6a9d69707d20e535750987/]

- `MessageDigest`: Clase que proporciona funciones de resumen criptográfico, como SHA-256 o MD5, para procesar datos.

[crayon-6a9d69707d210029547825/]

- `SSLContext`: Para configurar conexiones seguras con el protocolo SSL/TLS en Kotlin.

[crayon-6a9d69707d212637183535/]