* Fundamentos de Java
  + Sintaxe Básica:
    - Tipos de dados (int, char, float, boolean), variáveis, operadores, controle de fluxo (if, switch, for, while, do-while).
  + Classes e Objetos:
    - Definição de classes, criação de objetos, métodos, construtores, encapsulamento.
  + Herança:
    - Superclasse, subclasse, extends keyword, método super().
  + Polimorfismo:
    - Sobrecarga de métodos, sobrescrita de métodos, classes abstratas, interfaces.
  + Encapsulamento:
    - Modificadores de acesso (private, protected, public), getters e setters.
  + Pacotes e Importação:
    - Organização de classes em pacotes, uso da palavra-chave import.
* Coleções e Generics
  + **List**: Uma lista é uma coleção ordenada que pode conter elementos duplicados. A posição dos elementos é mantida.
    - **ArrayList**: Usa um array dinâmico internamente. Bom para acesso aleatório (get) rápido, mas operações de inserção e remoção são mais lentas (exceto no fim da lista).

**A computer screen with colorful text

Description automatically generated with medium confidence**

* + - **LinkedList**: Usa uma lista duplamente ligada internamente. Bom para inserções e remoções rápidas em ambas as extremidades da lista, mas o acesso aleatório é mais lento.

A black background with white text

Description automatically generated

* + **Set**: Um conjunto é uma coleção que não permite elementos duplicados
    - **HashSet**: Usa uma tabela hash para armazenar elementos. Não garante nenhuma ordem dos elementos. Operações são geralmente em tempo constante.

A black background with white text

Description automatically generated

* + - **TreeSet:** Usa uma árvore de busca binária (árvore Red-Black). Mantém os elementos ordenados. Operações são em tempo logarítmico

**A black background with white text

Description automatically generated**

* + - **LinkedHashSet: Mantém uma tabela hash e uma lista duplamente ligada. Mantém a ordem de inserção.**

**A black background with white text

Description automatically generated**

* + **Map**: Um mapa é uma coleção que mapeia chaves para valores. Não permite chaves duplicadas.
    - **HashMap**: Usa uma tabela hash para armazenar pares chave-valor. Não garante nenhuma ordem das chaves.

A black background with white text

Description automatically generated

* + - **TreeMap**: Usa uma árvore de busca binária (árvore Red-Black). Mantém as chaves ordenadas

A black background with white text

Description automatically generated

* + **Generics**: Tipos genéricos para assegurar a segurança de tipo em tempo de compilação.