Type:

- O type introduz um sinônimo para um tipo existente.
- Ele usa os mesmos construtores de dados do tipo original.
- Não cria uma nova estrutura de dados; apenas fornece um nome alternativo.
- Útil para criar nomes mais descritivos para tipos existentes.

Exemplo:

type Nome = String

Aqui, Nome é um sinônimo para o tipo String.

Newtype:

- O newtype introduz uma renomeação de um tipo existente.
- Requer que você forneça novos construtores.
- Garante que a representação do tipo seja exatamente a mesma do tipo que você está encapsulando.
- Útil quando você deseja criar um novo tipo com base em um tipo existente, mas com uma única construção.

Exemplo:

newtype Livro = Livro String

Aqui, Livro é uma renomeação de String, mas com um único construtor.

Data:

- O data declara uma nova estrutura de dados.
- Permite múltiplos construtores e campos.
- Pode ser usado para criar tipos algébricos complexos.

Exemplo:

data Opcao = Sim | Nao

Aqui, Opcao tem dois construtores: Sim e Nao.

Exemplo de Aplicação: Suponha que queremos representar um resultado de votação como “Sim” ou “Não”. Podemos usar newtype ou data:

Usando newtype:
newtype Voto = Voto String

Usando data:
data Voto = Sim | Nao

Ambas as abordagens permitem criar um valor do tipo Voto com os construtores Sim ou Nao. A diferença está na representação subjacente: o newtype garante que a representação seja a mesma que a de String, enquanto o data cria uma nova estrutura de dados.

Em resumo, escolha newtype quando você precisa de uma única construção com a mesma representação e data quando precisa de múltiplos construtores ou campos.