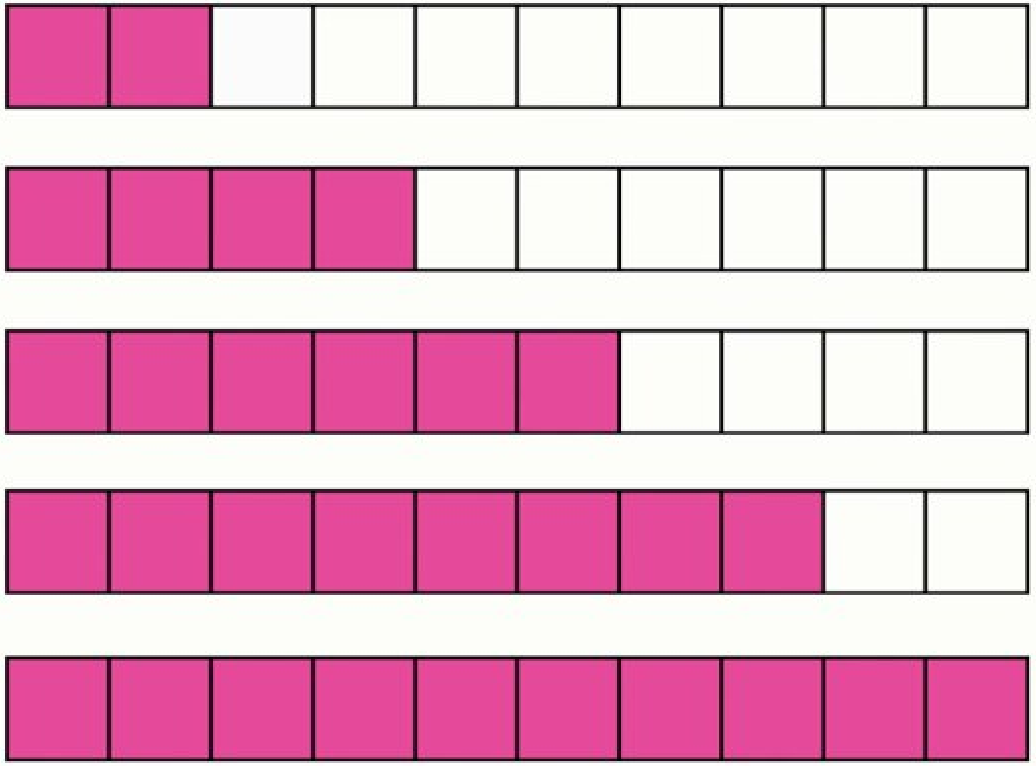
PyTorch

BASICS OF PYTORCH

DEEP LEARNING

LIST VS ARRAY VS TENSOR



	List	Array	Tensor
Type	Heterogeneous (elements can be of different types).	Homogeneous (elements are of the same type).	Homogeneous (elements are of the same type).
Dimensionality	Typically one-dimensional, can contain nested lists for higher dimensions.	Typically one-dimensional or multi-dimensional (2D, 3D, etc.).	Multi-dimensional, can have any number of dimensions (scalars, vectors, matrices, and higher).
Mutability	Mutable (elements can be changed).	Mutable (elements can be changed).	Mutable (elements can be changed).
Indexing	Zero-based indexing.	Zero-based indexing, often supports multi-dimensional indexing.	Zero-based indexing, supports multi-dimensional indexing.

	List	Array	Tensor
Performance	Generally slower for numerical operations due to heterogeneity.	Faster for numerical operations due to homogeneity and optimized implementations.	Optimized for numerical operations, especially with hardware acceleration (e.g., GPUs).
Libraries	Built-in Python data structure.	Provided by libraries like NumPy.	Provided by libraries like TensorFlow, **PyTorch**.
Use Cases	General-purpose data storage, especially for mixed data types.	Efficient numerical computing, handling arrays, and matrices.	Machine learning, deep learning, and complex numerical computations involving high-dimensional data.
AutoGrad	Not Available	Not Available	**Available**

AISCIENCES