Short notes on:

1. **Pandas**

   Pandas is an open-source Python library, which is used to analyze, clean, explore, and manipulate the data. It consists of data structures and functions to perform on data. Some key features of Pandas are:

   - Data visualization
   - Cleaning, merging, and joining of dataset.
   - Handling the missing data.

   There are two types of data structures in Pandas library:

   - Series
     - ➢ It is a one-dimensional array holding data of any type.
     - ➢ It is similar to a column in an Excel sheet.
     - ➢ It can be created from lists, dictionaries, etc. by loading the datasets from existing storage.
     - ➢ To create series from list:
       pd.Series(list)
   - DataFrame
     - ➢ It is a two-dimensional array with labeled axes (rows and columns).
     - ➢ It defines a whole table.
     - ➢ Creation of dataframe is similar to series.
     - ➢ To load data into a DataFrame object:
       pd.DataFrame(data)

2. **NumPy**

   NumPy stands for Numerical python which is an open source Python library to operate numerical operations on large and multi-dimensional arrays. It is featured for fast operations on arrays with mathematical, logical, shape manipulation, sorting, basic linear algebra and statistical operations, and many more other such operations.

   - NumPy Array Creation: Using array(), arrange(), linspace() function.
   - NumPy Array Indexing: Slicing, Integer array indexing, Boolean array indexing
   - NumPy Basic Operations:
     - ➢ NumPy-Unary Operators: Includes sum, min, max, cumsum, etc.
     - ➢ NumPy-Binary Operators: Basic arithmetic operators like +, -, *, /, etc.
     - ➢ NumPy's ufuncs: Includes mathematical functions like sin, cos, exp, sqrt, etc.
   - NumPy Sorting Arrays: Using sort() function.

3. **Tensorflow**

   Tensorflow makes it easy to create ML models that can run in any environment which is an open-source machine learning library. Tensorflow works by creating a computational graph and then executing it by passing the input values i.e.

multi-dimensional vectors (tensors). These tensors flow through the computational graph to obtain output.

4. **Keras**

   Keras is a higher level, user friendly API which can be run on top of tensorflow. It is also used for building deep learning models efficiently and conveniently in various fields like image processing, NLP, etc. It has been integrated with TensorFlow as 'tf.keras'. There are two ways to build model in keras:

   - Sequential API: Used with the models with single i/p and o/p and a linear stack of layers.
   - Functional API: Used for models that require multiple i/ps and o/ps, or layers of multiple i/ps and o/ps.

5. **PyTorch**

   PyTorch is an open-source deep learning framework built for flexibility and easeness. It provides a Python package for high-level features like tensor computation. It supports a dynamic computational graph allowing models to be optimized at runtime by constructing a directed acyclic graph consisting of a function that keeps track of all the executed operations on the tensors allowing to change shape, size and operations after every iterations. Used in reinforcement learning, image classification, NLP etc.

6. **Sklearn**

   Also known as scikit-learn, is an open-source ML and data modeling library for Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering, dimensionality reduction, model selection, and preprocessing via a consistent interface in Python.