PyTorch workshop

### **11:00 a.m. - 1:00 p.m. *(Explanations)***

#### **1. Introduction to PyTorch (11:00 - 11:15 a.m., 15 mins)**

* Overview of PyTorch: Uses and applications in AI/ML
* Comparison with other frameworks (TensorFlow, Keras)
* Introduction to PyTorch’s core features (e.g., dynamic computation graphs)

#### **2. Tensor Basics and Matrix Operations (11:15 - 11:45 a.m., 30 mins)**

* Understanding tensors and differences from Numpy arrays
* Tensor creation, initialization, and basic properties
* Essential tensor operations (reshaping, slicing, mathematical operations)

#### **3. Data Handling in PyTorch (11:45 a.m. - 12:15 p.m., 30 mins)**

* Introduction to PyTorch datasets and DataLoader classes
* Preparing and loading custom datasets
* Importance of batching, shuffling, and parallel data loading

#### **4. Building Neural Networks (12:15 - 12:45 p.m., 30 mins)**

* PyTorch’s nn module for defining neural networks
* Creating a simple neural network architecture (fully connected layers)
* Explanation of activation functions and forward pass

#### **5. Training and Evaluation (12:45 - 1:00 p.m., 15 mins)**

* Basics of model training (loss functions, optimizers)
* Evaluation metrics for model performance (accuracy, loss, etc.)

### **1:00 - 2:00 p.m. *(Lunch Break)***

### **2:00 - 3:15 p.m. *(Hands-on Practice)***

#### **1. Tensor Basics and Matrix Operations Hands-On (2:00 - 2:15 p.m., 15 mins)**

* Creating and manipulating tensors in PyTorch
* Practice exercises: tensor reshaping, slicing, and basic arithmetic operations

#### **2. Data Handling in PyTorch Hands-On (2:15 - 2:30 p.m., 15 mins)**

* Implementing a DataLoader for a sample dataset
* Mini-exercise: Loading and previewing a dataset in batches

#### **3. Building a Neural Network Hands-On (2:30 - 2:50 p.m., 20 mins)**

* Defining a simple neural network using nn.Module
* Adding layers, activation functions, and customizing the architecture

#### **4. Training and Evaluation Hands-On (2:50 - 3:15 p.m., 25 mins)**

* Training the network on a sample dataset
* Implementing a basic evaluation loop
* Observing and interpreting loss and accuracy metrics