**INTEL IMAGE CLASSIFICATION**

This project helps you to apply **Python, Numpy, Pandas, Matplotlib, OpenCV, Tensorflow, Keras, CNN and its variations depending upon various sprints you attempt.**

**Modules utilized:**

* Numpy
* Pandas
* OpenCV
* sklearn
* Tensorflow
* Keras

**Recommended Tools:** Use Google Colab to perform the sprints, and while unzipping the data **AVOID UNZIPPING IN NESTED FOLDERS.**

**ABOUT DATASET**

**Context**

This is image data of Natural Scenes around the world.

**Content**

This Data contains around 17k images of size 150x150 distributed under 6 categories.

{'buildings' -> 0,

'forest' -> 1,

'glacier' -> 2,

'mountain' -> 3,

'sea' -> 4,

'street' -> 5 }

The Train, Test and Prediction data is separated in each zip file.

**There are around 14k images in Train, 3k in Test**

[**DATASET DOWNLOAD LINK**](https://drive.google.com/file/d/12oGJWnBYd8WpXnHXxPMoXCFm5CaFPPOi/view?usp=share_link)

**SPRINT -1**

**Task 1**: Load the dataset into a data frame

**Task 2:** Analyze what data you have in what proportions (Graph Visualization: Bar Graph, Pie Charts etc.)

**SPRINT - 2 (For ML only)**

**TASK 1:** Perform Image Pre-Processing Steps

**TASK 2:** Build a Machine Learning model to perform classification

**TASK 3:** Perform Hyperparameter tuning if required for the ***best performing model only.***

**TASK 4:** Perform the predictions in the following manner:

* Create an csv file with the following columns: Image name and Predicted class label
* Pass the training data through your best performing model and whatever predictions are made on each image store it in the csv and submit the CSV file as the final submission.
* The csv file should have 2 columns: image name and predicted label

**TASK 5:** Evaluate the model performance for the best model before and after hyperparameter tuning.

**SPRINT -3 (For Deep Learning Students)**

**TASK 1:** Perform Image Pre-Processing steps and use ImageDataGenerator if you feel the need for it.

**TASK 2:** Build two deep learning models:

* CNN Model
* VGG-16 by **using transfer learning only**

**TASK 3:** Fine tune VGG-16 using keras tuner if required

**TASK 4:** Perform the predictions in the following manner:

* Create an csv file with the following columns: Image name and Predicted class label
* Pass the training data through your best performing model and whatever predictions are made on each image store it in the csv and submit the CSV file as the final submission.
* The csv file should have 2 columns: image name and predicted label

**TASK 4:** Evaluate both CNN and VGG-16 performance (in case of VGG-16 pre and post tuning if performed).

**📍NOTE: ALL THE PARTICIPANTS ARE REQUESTED TO WRITE DOWN THEIR OBSERVATIONS IN A PROPER MANNER TO INDICATE THEIR FINDINGS AND DECISIONS MADE.**

**Submission form link 👇**

[**https://forms.gle/NNrzoRxPf1BpD72d7**](https://forms.gle/NNrzoRxPf1BpD72d7)

**Note: Zip the Jupyter notebook and CSV File and Zip file should contain your Full Name 📁**

**All the Best !!! ✌👍  
Team Innomatics Research Labs**