

```

1 def load_dataset(path_to_segments_file):
2 # part of DatasetHandler which creates the Dataset object
3     dataset = Dataset()
4     dataset.init_from_segments(path_to_segments_file):
5         # loads 4 important lists:
6         Segments = LoadDataFile(path_to_segments_file)
7         list_of_images, labels, osm, segment_ids = self.load_data_from_segments(
8             Segments)
9         self.init_from_lists(list_of_images, labels, osm, segment_ids)
10
11     dataset.shuffle_by_segments()
12     # shuffles data while respecting that the data from one segment should be next
13     to each other
14
15     return dataset
16
17 def load_data_from_segments(Segments):
18     for Segment in Segments:
19         for i_th_image in Segment.number_of_images:
20             index = Segment.location_index[i_th_image]
21
22             list_of_images <- Segment.get_image_filename(i_th_image)
23             labels <- Segment.get_score()
24             osm <- Segment.distinct_nearby_vector[index]
25             segment_ids <- Segment.get_id()
26     return list_of_images, labels, osm, segment_ids

```