# Dataset Annexure

## Columns

- category – A number representing the category to which the listing belongs
- title – The title of the listing
- subtitle – The subtitle of the listing
- gallery_url – The primary image URL for the listing
- picture_url – Semicolon ";" separated list of all associated image URLs for the listing
- attributes – Seller-provided attributes for the listing as a list of key-value pairs, see below for details
- description_html_base64 – Seller-provided listing description HTML which is base64 encoded
- index – A numeric identifier for each listing

## Attributes Parsing Logic

The attributes contain noisy data using the format key1:value1,key2:value2,… The suggested approach is to perform a 2-level parsing. First split on one or more colons ":+" using a regular expression split, then split on comma ",". The last term in each comma-split group is the name of the next group, the other terms in each group make up the values. This is not 100% correct because the data is noisy, but it does work most of the time. Here is an example:

(Colors:blue, white,Special Note::very nice,Style: Modern)

Step 1: Remove containing parenthesis and split by one or more colons:
["Colors", "blue, white,Special Note", "very nice,Style", "Modern"]

Step 2: Split by commas:
[[Non-last terms: N/A, last term: "Colors"], [Non-last terms: ["blue", " white"], last term: "Special Note"], [Non-last terms: ["very nice"], last term: "Style"], [Non-last terms: ["Modern"], last term: N/A]]

Step 3: Putting each last term as key followed by the non-last terms from the next group as value:
[ "Colors":["blue", " white"], "Special Note":["very nice"], "Style":["Modern"]]

Step 4: Join the value lists again by comma ",":
[ "Colors":"blue, white", "Special Note":"very nice", "Style":"Modern"]