In [5]:
```python
import cv2

frozen_model = 'C:\\Users\\shars\\AppData\\Roaming\\Microsoft\\Windows\\Start Menu\\Programs\\Python 3.9\\frozen_inference_graph.pb'
config_file = 'C:\\Users\\shars\\AppData\\Roaming\\Microsoft\\Windows\\Start Menu\\Programs\\Python 3.9\\ssd_mobilenet_v3_large_coco_2020_01_14.pbtxt'

model = cv2.dnn_DetectionModel(frozen_model, config_file)

model.setInputSize(320, 320)
model.setInputScale(1.0/127.5) # 255 / 2 = 127.5
model.setInputMean((127.5, 127.5, 127.5)) # mobilenet => [-1, 1]
model.setInputSwapRB(True)
```

Out[5]: `<dnn_Model 00000182DA94D8D0>`

In [8]:
```python
classlabels = []
file_name = 'C:/Users/shars/Desktop/lables.txt'
with open(file_name, 'rt') as fpt:
    classlabels = fpt.read().rstrip('\n').split('\n')
```

In [17]:
```python
import matplotlib.pyplot as plt

img = cv2.imread('C:/Users/shars/Desktop/john-arano-LzxsSWAVMYs-unsplash.jpg')
plt.imshow(cv2.cvtColor(img, cv2.COLOR_BGR2RGB))
```

Out[17]: `<matplotlib.image.AxesImage at 0x182f04d16a0>`

In [18]:
```python
ClassIndex, confidence, bbox = model.detect(img, confThreshold=0.5)
```

In [19]:
```python
print(confidence)
```

```
[[0.63268983]
 [0.62289923]
 [0.61495084]
 [0.59208065]
 [0.58845043]
 [0.5691272 ]
 [0.56890094]
 [0.56597775]
 [0.5658475 ]
 [0.5581588 ]
 [0.55553865]
 [0.5544171 ]
 [0.5502028 ]
 [0.54977566]
 [0.54632646]
 [0.5440526 ]
 [0.5321235 ]
 [0.53126436]
 [0.52667063]]
```

In [20]:
```python
print(bbox)
```

```
[[186 649  48  68]
 [371 622  53  66]
 [223 622  33  42]
 [ 97 528  13  30]
 [395 620  69  80]
 [155 647  52  75]
 [ 65 649 117  89]
 [193 624  39  49]
 [560 625  42  53]
 [510 623  37  48]
 [262 617 120 121]
 [ 38 542 566 154]
 [398 638  59  81]
 [551 592  12  31]
 [548 617  27  59]
 [ 93 536  11  27]
 [431 647  84  77]
 [377 534   9  26]
 [495 374  52  67]]
```

In [21]:
```python
print(ClassIndex)
```

```
[[ 3]
 [ 3]
 [ 3]
 [10]
 [ 3]
 [ 3]
 [ 3]
 [ 3]
 [ 3]
 [ 3]
 [ 3]
 [ 3]
 [ 3]
```
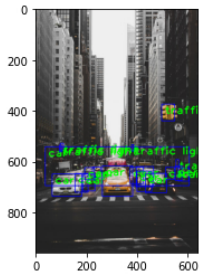
```
[ 3]
[10]
[ 1]
[10]
[ 3]
[10]
[10]]
```

In [22]:
```
print(classlabels)
```

```
['person', 'bicycle', 'car', 'motorbike', 'aeroplane', 'bus', 'train', 'truck', 'boat', 'traffic light', 'fire hydrant', 'stop sign', 'parking meter', 'bench', 'bird', 'ca
t', 'dog', 'horse', 'sheep', 'cow', 'elephant', 'bear', 'zebra', 'giraffe', 'backpack', 'umbrella', 'handbag', 'tie', 'suitcase', 'frisbee', 'skis', 'snowboard', 'sports bal
l', 'kite', 'baseball bat', 'baseball glove', 'skateboard', 'surfboard', 'tennis racket', 'bottle', 'wine glass', 'cup', 'fork', 'knife', 'spoon', 'bowl', 'banana', 'apple',
'sandwich', 'orange', 'broccoli', 'carrot', 'hot dog', 'pizza', 'donut', 'cake', 'chair', 'sofa', 'pottedplant', 'bed', 'diningtable', 'toilet', 'tvmonitor', 'laptop', 'mous
e', 'remote', 'keyboard', 'cell phone', 'microwave', 'oven', 'toaster', 'sink', 'refrigerator', 'book', 'clock', 'vase', 'scissors', 'teddy bear', 'hair drier', 'toothbrus
h']
```

In [23]:
```
font_scale = 3
font = cv2.FONT_HERSHEY_PLAIN
for ClassInd, conf, boxes in zip(ClassIndex.flatten(), confidence.flatten(), bbox):
    cv2.rectangle(img, boxes, (255, 0, 0), 2)
    cv2.putText(img, classlabels[ClassInd-1], (boxes[0]+10, boxes[1]+40), font, fontScale=font_scale, color=(0, 255, 0), thickness=3)
```

In [24]:
```
plt.imshow(cv2.cvtColor(img, cv2.COLOR_BGR2RGB))
```

Out[24]: <matplotlib.image.AxesImage at 0x182ef2038b0>

In [ ]: