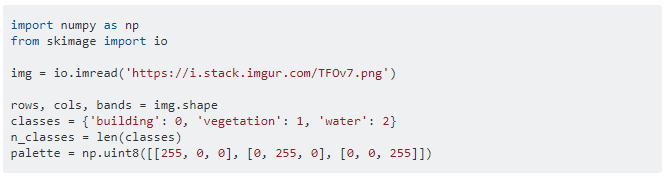
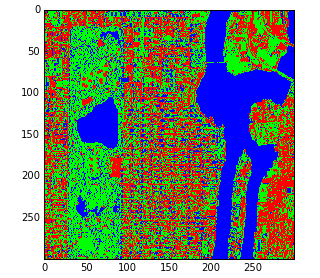
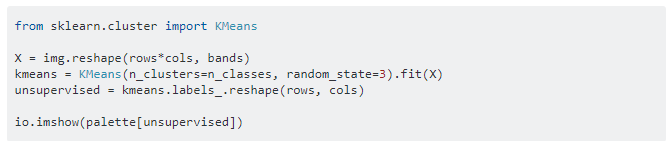
**EXPLANATION OF CODE**

We start off by reading the image and defining some variables that will be used later on.



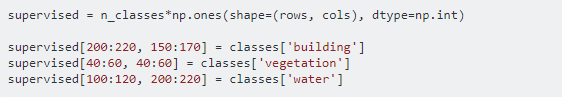
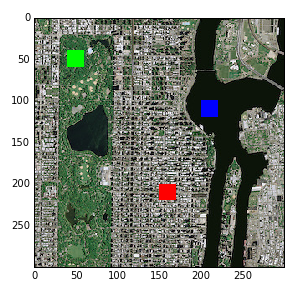
**Unsupervised classification:**

If you don't wish to manually label some pixels then you need to detect the underlying structure of your data, i.e. you have to split the image pixels into n\_classes partitions, for example through [k-means clustering](https://en.wikipedia.org/wiki/K-means_clustering):

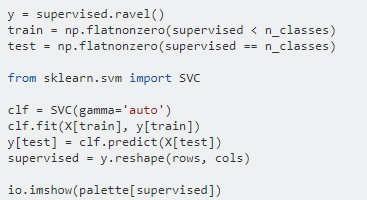


## Supervised classification:

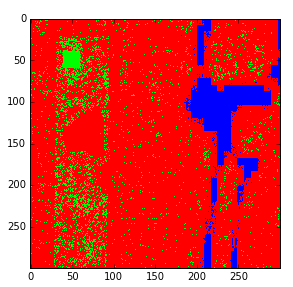
## Alternatively, you could assign labels to some pixels of known class (the set of labeled pixels is usually referred to as ground truth). In this toy example the ground truth is made up of three hardcoded square regions of 20×20 pixels shown in the following figure:



The pixels of the ground truth (training set) are used to fit a support vector machine.



After the training stage, the classifier assigns class labels to the remaining pixels (test set). The classification results look like this:



## Final remarks:

## Results seem to suggest that unsupervised classification is more accurate than its supervised counterpart. However, supervised classification generally outperforms unsupervised classification. It is important to note that in the analyzed example accuracy could be dramatically improved by adjusting the parameters of the SVM classifier. Further improvement could be achieved by enlarging and refining the ground truth, since the train/test ratio is very small and the red and green patches actually contain pixels of different classes. Finally, one can reasonably expect that utilizing more sophisticated features such as ratios or indices computed from the intensity levels (for instance [NDVI](https://en.wikipedia.org/wiki/Normalized_difference_vegetation_index)) would boost performance.