Software needed:
bowtie2 or stampy
samtools

NOTE: I include separate tutorials for two different mappers: bowtie2 and stampy. If you are mapping to a genome of the same species use bowtie2, it's much faster. For mapping to divergent genomes, stampy will get you a higher mapping rate and you can specify expected divergence (see stampy website: http://www.well.ox.ac.uk/~gerton/README.txt). If you're unsure, you can try mapping a single sample with each.

MAPPING WITH BOWITE2

1. Build database for reference genome

You only need to do this once per reference genome. I put all my assemblies into a folder called "references" so I can easily access them with minimal change to the script. Here, "YWARv0" is the name I'm giving to the database.

```
bowtie2-build YWAR_min1000_sm.fasta YWARv0
```

2. Map duplicate filtered samples to genome assembly

We convert the files directly to bam and sort them with sam tools in order to save space. While mapping, we also add read groups that identify each sample as well as the library and lane ("YWAR1" in this example where I sequenced 1 library per lane). This information is used by GATK later.

```
for sample in `ls *.1.1.fq.gz | cut -f1 -d'.'`
do
     bowtie2 -x ~/nobackup-klohmuel/References/YWARv0/YWARv0 \
     --threads 8 -1 $sample.1.1.fq.gz -2 $sample.2.2.fq.gz \
     --rg-id $sample --rg SM:$sample --rg ID:$sample \
     --rg LB:YWAR1 --rg PU:YWAR1 --rg PL:illumina | \
     samtools view -bhS - | \
     samtools sort - $sample
done
```

Alignment rates will be output as stdout. It's good to pay attention to these at this step, especially if you are using a genome from another species. You may want to fine tune the parameters in bowtie a bit.

My script for this is RADmap.sh

MAPPING WITH STAMPY

The basic steps are the same as above, so see notes in the bowtie2 section for explanation of each step.

1. Build database for reference genome

```
stampy.py –G YWAR YWAR_min1000_sm.fasta

stampy.py –g YWAR –H YWAR
```

2. Map duplicate filtered samples to genome assembly

```
for sample in `ls *.1.1.fq.gz | cut –f1 –d'.'`
do
     stampy.py –g YWAR –h YWAR –t 8 \
     –M $sample.1.1.fq.gz,$sample.2.2.fq.gz \
     ––readgroup="SM:$sample","ID:$sample","LB:YWAR1",\
     "PU:YWAR1","PL:illumina" | \
     samtools view –bhS – | \
     samtools sort – $sample
done
```