First need a blast database

- The sequence(s) you are going to blast to.
- Create this using the command:

```
makeblastdb -in <fasta file> -dbtype <nucl>
```

  - types: **`nucl`** or **`prot`**

- If you include the ".fasta" file extension when the command, it will become part of the db name:
  - **`-in reference.fasta`** versus **`-in reference`**

Then blast your seqs to your db

- blastn Search a **nucleotide** database using a **nucleotide** query
- blastp Search **protein** database using a **protein** query
- blastx Search **protein** database using a **translated nucleotide** query
- tblastn Search **translated nucleotide** database using a **protein** query
- tblastx Search **translated nucleotide** database using a **translated nucleotide** query

Then blast your seqs to your db

- **`blastn -query <input.fasta> -db <dbname> -out <output file name> -outfmt 6`**
  - blastn, blastp, blastx, tblastn, tblastx
- Various output formats are available
  - -outfmt 6 provides tab-delimited text
    - useful for batch processing of output
  - you can also specify which variables are reported
    - -outfmt "6 qseqid sseqid pident length mismatch gapopen qstart qend sstart send evalue bitscore btop"
    - btop produces a cigar-like field (details on the alignment)

Can specify blast parameters

- -evalue 0.001(only output alignments with evalues <= this value)
  - **-e** 1e10-4
  - **-G** Cost to open gap [Integer]: default = 5 for nucleotides/ 11 for proteins
  - **-E** Cost to extend gap [Integer]: default = 2 for nucleotides/ 1 for proteins
  - **-q** Penalty for nucleotide mismatch [Integer]: default = -3
  - **-r** reward for nucleotide match [Integer]: default = 1