

Help document

1. **iPtgxDB_insilico.jar**: In silico ORF predictor by Ulrich Omasits. Generates a GFF file of open reading frames for a given genomic sequence. Alternative start codons and minimal length are configurable.

Usage: java -jar iPtgxDB_insilico.jar [-alt <codon(s)>] [-extend <e>] [-laa] [-min <length>] [-out <file>] [-seq <file>] [-tab <output>]

-alt <codon(s)>	Alternative start codons (default: GTG TTG CTG)
-extend <e>	Get extension for all features 3' and 5' by <e> nucleotides in tabular output file
-laa	Take only longest alternative anchor for regions without a main start codon
-min <length>	The minimum protein length in aa (default: 10)
-out <file>	The output gff file
-seq <file>	The input sequence
-tab <output>	A tab-separated output file with sequences per entry

2. **iPtgxDB_convert.jar**: Genome Annotation Converter by Ulrich Omasits. Converts genome annotations from various formats to the GFF file format.

Usage: java -jar iPtgxDB_convert.jar [-all] [-broad <file>] [-chemgenome <file>] [-cmr <file>] [-ensembl <file>] [-extend <e>] [-genbank <file>] [-genoscope <file>] [-id <sequenceid>] [-out <output>] [-repseek <file>] [-seq <input>] [-short <file>] [-tab <output>]

-all	Convert extensive, i.e. all features and attributes
-broad <file>	A BROAD genome summary per gene file
-chemgenome <file>	One or more fasta files generated by ChemGenome 2.0
-cmr <file>	A tab-separated tabular text file from CMR
-ensembl <file>	A plain-text Ensembl file
-extend <e>	Extend all features 3' by <e> nucleotides or automatically if e='auto'
-genbank <file>	A plain-text GenBank file
-genoscope <file>	A tab-separated tabular text file from Genoscope/MAGE
-id <sequenceid>	The sequence id for the GFF file [default: guessed from input file]
-out <output>	The output gff3 file
-repseek <file>	A repseek R-Table output file
-seq <input>	The genomic sequence
-short <file>	A tab-separated short ORF file
-tab <output>	A tab-separated output file with sequences per entry [requires seq]

- 3. iPtgxDB_combine.jar:** Gff genome annotation combiner by Ulrich Omasits. Combines annotations from different sources (GFF format) by clustering them using their stop codon coordinates. Clustered annotations are merged hierarchically and renamed to generate unique and information-rich iPtgxDB identifiers. The output is a detailed matching table (tsv format), the matched annotations (gff format) and a minimally redundant iPtgxDB protein database (fasta format).

Usage: java -jar iPtgxDB_combine.jar [-chemg <input>] [-cmr <input>] [-easy <input>] [-ens <input>] [-enz <pattern>] [-extend <e>] [-fasta <output>] [-fullseq] [-geno <input>] [-gff <output>] [-id <prefix>] [-min <length>] [-nopseudo] [-orf <input>] [-prod <input>] [-pseudoproteins] [-ref <input>] [-refseq <input>] [-seq <input>] [-short <input>] [-tab <output>] [-tsv <output>]

-chemg <input>	The input chemgenome annotation gff3 file
-cmr <input>	The input cmr annotation gff3 file
-easy <input>	The input Easygene annotation gff2 file
-ens <input>	The input ensembl annotation gff3 file
-enz <pattern>	Cleavage pattern of the restriction enzyme (default: '([KR])([AP])')
-extend <e>	Get extension for all features 3' and 5' by <e> nucleotides in tabular output file
-fasta <output>	The output protein database
-fullseq	Output full protein sequence for extension/reduction entries
-geno <input>	The input genoscope annotation gff3 file
-gff <output>	The output gff file
-id <prefix>	The ID prefix for the organism used
-min <length>	The minimum protein length in aa for export in protein database (default: 6)
-nopseudo	Ignore pseudo entries completely
-orf <input>	The input in-silico annotation gff3 file
-prod <input>	The input Prodigal annotation gff3 file
-pseudoproteins	Also write pseudo entries to the protein database
-ref <input>	A generic reference annotation gff3 file
-refseq <input>	The input refseq annotation gff3 file
-seq <input>	The genomic input sequence
-short <input>	The input shortORF annotation gff3 file
-tab <output>	A tab-separated output file with sequences per entry
-tsv <output>	The output matching table