Sarah Griffiths - Middle layer user guide

Follow these instructions to use the Business Logic Layer - All code is written in python.

Front End instructions

1. You must have all the files saved in 1 directory – this should already be set up.

2. Import the module BusinessLogicMainFile.py in to your code

3. Use following functions as required

   Functions To return required lists of records.

   getAllGenes() – returns A-Z geneList

   getAllProteins() – returns A-Z proteinList

   getAllAccessions() – returns A-Z accessionList

   getAllChromosomeLocations() – returns ordered list of chromosomeLocationList

4. Call this function to retrieve all associated record data.

   getAllEntryData(keyword, type)

   keyword – Any accession/gene/protein/NCBI ID/Chromosomal location
   type - accession number, NCBI ID, protein name, gene name

   This will return the following python variables

Variable name (literal)	Explanation of what variable contains	Variable type
accession_number	Returns accession no from data access layer	string
NCBI_identifier	NCBI ID	string
Chromosome_location	Chromosome location	string
Protein_product_name	Protein product name	string
parsedSequence	Entire nucleic acid sequence	String
codingRegion	Coding region of nucleic acid sequence	String
mrnaSequence	Mrna sequence – (t's replaced with u's)	String
splitSequence	Mrna sequence split in to codons	List
translatedAndAligned	Sequence translated in to AA and aligned with Nucleic acid sequence	2Aligned Strings
justAminoAcids	Just AA sequence	String
codonFrequency	Codon frequencies	Dictionary
RestrictionEnzymes	Dictionary of associated RE and their respective sequences	Dictionary

Example –

getAllEntryData(U49845)

Returns -

parsedSequence = “actgtgggggtcacgtcgta…”

codingRegion = “gtcacgtgta…”

mrnaSequence = “gugcacgugaaaguggucgugugugggugucgugggggcaaaa….”

splitSequence = gug,cac,gug,aaa,gug,guc,gug,ugu,ggg,ugu,cgu,ggg,gca,aaa….”

translatedAndAligned = ‘gugcacgugaaaguggucgugugugggugucgugggggcaaaa..’, ‘T--B--A--G--T--N--R..’

JustAminoAcids = 'TBAGTNR...'

codonFrequencies = codons = {"uuu": 20% , "uuc": 30%, "uua": 20%, "uug": 30% …}

RestrictionEnzymes = ‘EcoRI’,’BamHI’,’BsuMI…’

You can then use these variables however you choose

5. Call this function to find cut sites

   restrictionEnzymeCutSites(bases) –

   bases – input the sequence you are looking for eg. ‘tcgaa’ - must include string quotations(‘)

   This will return a dictionary of the format

   (start cut point, end cut point : ‘within coding region/not within coding region’)

   Example
   restrictionEnzymeCutSites(‘tcgaa’)

   Returns

   {(127, 340 : “not in coding region”), (500, 1500: “in coding region”), (1600, 2100: “not in coding region”

Back end instructions

1. Import module main_file.py
2. To calculate total codon frequency use function

   Totalcodonfrequency() – This will return a dictionary containing the 64 codons and their respective frequencies. This can then be stored in the DB and called upon again by the middle layer.