Assignment #1

Instructions:

1. ALL DUE TIMES ARE IN EST
2. **Upload Python files to this BB Assignment.**
3. All answers must be in your own words, and copy-and-paste answers will receive no credit.
4. You must submit .py files to be graded.
5. You are limited to 2 submissions
6. If you cannot submit the assignment by the due date, you can still submit it by the two days with a 25%penalty.
7. Any submissions after two days of the assignment class will be given a zero.

What your programs have to do:

1. Given an average molecular weight of 110 Daltons per amino acid, write a Python program that calculates the estimated molecular weight of the Rattus norvegicus PKC Beta-1 protein (in **kilodaltons**) and writes the result on the screen. Hard-code in (copy and paste to a variable in your code) the protein **sequence**. Hard coding is not just related to the python language, it is a general practice in programming where you assign a constant value (here the protein sequence) and program refers to that value for processing. Call your program: **calc_daltons.py.**

   Hint:

   ```
   # to get rid of newline characters
   protein_sequence =
   protein_sequence.replace('\r','').replace('\n', '')
   ```

   Output Example:

   ```
   The length of "Protein kinase C beta type" is: 671
   The average weight of this protein sequence in kilodaltons
   is: 73.81
   ```

2. **Processing DNA**

   There is a sample of several DNA sequences, one per line. Each sequence starts with the same 14 base pair fragment – a sequencing adapter that should have been removed. Write a program that will (a) trim this adapter and write the cleaned sequences to the screen and (b) print the length of each sequence to the screen. Hard-code in the DNA sequences as a list of strings. You have to use a loop for answering this question.

   ATTCGATTATAAGCTCGATCGATCGATCGATCGATCGATCGATCGATCGATC

   ATTCGATTATAAGCACTGATCGATCGATCGATCGATCGATGCTATCGTCGT

ATTCGATTATAAGCATCGATCACGATCTATCGTACGTATGCATATCGATATCGATCGTAGTC

ATTCGATTATAAGCACTATCGATGATCTAGCTACGATCGTAGCTGTA

ATTCGATTATAAGCACTAGCTAGTCTCGATGCATGATCAGCTTAGCTGATGATGCTATGCA

3. **Generating random DNA sequences**:
   Generate a random DNA string of 1001 bases long and store it in a string variable. To do this, use a list of nucleotides and the random.randint() function from the random module of Python. Print out the percentages of the four nucleotides in your random DNA string. Run your program several times. Do you get the same random sequence? (Answer in a comment in your code)