Matt White - Class 6 project

Quarto

Quarto enables you to weave together content and executable code into a finished document. To learn more about Quarto see https://quarto.org.

Running Code

When you click the **Render** button a document will be generated that includes both content and the output of embedded code. You can embed code like this:

```r
1 + 1
```

```
[1] 2
```

You can add options to executable code like this

```
[1] 4
```

The `echo: false` option disables the printing of code (only output is displayed).

My first function !

```r
add <- function(x, y) {
  x + y
}

add(x = 13, y = 71)
```

```
[1] 84
```

```r
add(13, 71)
```

```
[1] 84
```

```r
add(5, 54)
```

```
[1] 59
```

```r
add(c(100,1,100),1)
```

```
[1] 101   2 101
```

Make a function "generate_DNA()" that makes a random nucleotide sequence of any length.

```r
#generate_DNA <- function(  ) { }


generate_DNA <- function(length) {
  bases <- c("A", "C", "G", "T")
  sequence <- sample(bases, size = length, replace = T)
  return(sequence)
}

generate_DNA(20)
```

```
 [1] "G" "T" "G" "A" "A" "C" "G" "G" "G" "T" "T" "A" "C" "C" "T" "C" "C" "G" "G"
[20] "T"
```

The emblem :: will give us just one element from the bio3d package

```r
bio3d::aa.table
```

```
    aa3 aa1    mass       formula                            name
ALA ALA   A  71.078    C3 H5 N O1                         Alanine
ARG ARG   R 157.194  C6 H13 N4 O1                        Arginine
ASN ASN   N 114.103   C4 H6 N2 O2                      Asparagine
ASP ASP   D 114.079    C4 H4 N O3                   Aspartic Acid
CYS CYS   C 103.143  C3 H5 N O1 S                         Cystein
```

GLN	GLN	Q	117.126	C4 H9 N2 O2	Glutamine
GLU	GLU	E	128.106	C5 H6 N O3	Glutamic Acid
GLY	GLY	G	57.051	C2 H3 N O1	Glycine
HIS	HIS	H	137.139	C6 H7 N3 O1	Histidine
ILE	ILE	I	113.158	C6 H11 N O1	Isoleucine
LEU	LEU	L	113.158	C6 H11 N O1	Leucine
LYS	LYS	K	129.180	C6 H13 N2 O1	Lysine
MET	MET	M	131.196	C5 H9 N O1 S	Methionine
PHE	PHE	F	147.174	C9 H9 N O1	Phenylalanine
PRO	PRO	P	97.115	C5 H7 N O1	Proline
SER	SER	S	87.077	C3 H5 N O2	Serine
THR	THR	T	101.104	C4 H7 N O2	Threonine
TRP	TRP	W	186.210	C11 H10 N2 O1	Tryptophan
TYR	TYR	Y	163.173	C9 H9 N O2	Tyrosine
VAL	VAL	V	99.131	C5 H9 N O1	Valine
ABA	ABA	X	85.104	C4 H7 N1 O1	alpha-aminobutyric acid
ASH	ASH	D	115.087	C4 H5 N O3	Aspartic acid Neutral
CIR	CIR	R	157.170	C6 H11 N3 O2	citrulline
CME	CME	C	179.260	C5 H9 N O2 S2	s,s-(2-hydroxyethyl)thiocysteine
CMT	CMT	C	115.154	C4 H5 N O1 S	o-methylcysteine
CSD	CSD	C	134.134	C3 H4 N O3 S	s-cysteinesulfinic acid
CSO	CSO	C	119.142	C3 H5 N O2 S	s-hydroxycysteine
CSW	CSW	C	135.142	C3 H5 N O3 S	cysteine-s-dioxide
CSX	CSX	C	119.142	C3 H5 N O2 S	s-oxy cysteine
CYM	CYM	C	102.135	C3 H4 N O1 S	Cystein Negative
CYX	CYX	C	102.135	C3 H4 N O1 S	Cystein SSbond
DDE	DDE	H	280.346	C13 H22 N5 O2	diphthamide
GLH	GLH	E	129.114	C5 H7 N O3	Glutatmic acid Neutral
HID	HID	H	137.139	C6 H7 N3 O1	Histidine
HIE	HIE	H	137.139	C6 H7 N3 O1	Histidine
HIP	HIP	H	138.147	C6 H8 N3 O1	Histidine Positive
HSD	HSD	H	137.139	C6 H7 N3 O1	Histidine
HSE	HSE	H	137.139	C6 H7 N3 O1	Histidine
HSP	HSP	H	138.147	C6 H8 N3 O1	Histidine Positive
IAS	IAS	D	115.087	C4 H5 N O3	beta-aspartyl
KCX	KCX	K	172.182	C7 H12 N2 O3	lysine nz-carboxylic acid
LYN	LYN	K	129.180	C6 H13 N2 O1	Lysine Neutral
MHO	MHO	M	147.195	C5 H9 N O2 S	s-oxymethionine
MLY	MLY	K	156.225	C8 H16 N2 O1	n-dimethyl-lysine
MSE	MSE	M	178.091	C5 H9 N O1 SE	selenomethionine
OCS	OCS	C	151.141	C3 H5 N O4 S	cysteinesulfonic acid
PFF	PFF	F	165.164	C9 H8 F N O1	4-fluoro-l-phenylalanine
PTR	PTR	Y	243.153	C9 H10 N O5 P	o-phosphotyrosine

```
SEP SEP   S 167.057  C3 H6 N O5 P                     phosphoserine
TPO TPO   T 181.084  C4 H8 N O5 P                   phosphothreonine
```

```
#Only want to access aa1 column

bio3d::aa.table$aa1
```

```
 [1] "A" "R" "N" "D" "C" "Q" "E" "G" "H" "I" "L" "K" "M" "F" "P" "S" "T" "W" "Y"
[20] "V" "X" "D" "R" "C" "C" "C" "C" "C" "C" "C" "C" "H" "E" "H" "H" "H" "H" "H"
[39] "H" "D" "K" "K" "M" "K" "M" "C" "F" "Y" "S" "T"
```

```
#get one entry per amino acid

unique(bio3d::aa.table$aa1)[1:20]
```

```
 [1] "A" "R" "N" "D" "C" "Q" "E" "G" "H" "I" "L" "K" "M" "F" "P" "S" "T" "W" "Y"
[20] "V"
```

Now write a function that generates protein sequence of any length

```
generate_protein <- function(length) {
  amino_acids <- unique(bio3d::aa.table$aa1)[1:20]
  sequence <- sample(amino_acids, size = length, replace = T)
  return(sequence)
}

generate_protein(20)
```

```
 [1] "W" "W" "P" "K" "Q" "M" "V" "M" "N" "A" "R" "Q" "T" "G" "C" "A" "I" "Y" "P"
[20] "A"
```

Generate a set of random protein sequences of incremental length 6 to 13

Could do `generate_protein(6)`, `generate_protein(7)`, etc..

Or we can use a different useful function, `apply()` or `sapply()`

```
sapply(6:12, generate_protein)
```

```
[[1]]
[1] "M" "D" "W" "S" "P" "A"

[[2]]
[1] "L" "A" "F" "H" "M" "M" "G"

[[3]]
[1] "C" "K" "H" "V" "Q" "K" "N" "N"

[[4]]
[1] "N" "M" "G" "F" "F" "S" "G" "Y" "R"

[[5]]
 [1] "W" "M" "S" "Y" "G" "K" "A" "G" "H" "V"

[[6]]
 [1] "W" "W" "C" "G" "L" "L" "S" "E" "C" "H" "I"

[[7]]
 [1] "T" "D" "H" "T" "E" "L" "L" "T" "Y" "N" "E" "W"
```

Get this function to paste with no spaces, in format we could put into blast

```r
generate_protein <- function(length) {
  amino_acids <- unique(bio3d::aa.table$aa1)[1:20]
  sequence <- sample(amino_acids, size = length, replace = T)
  sequence <- paste(sequence, collapse = "")
  #paste with the specification of "" for collapse will provide our sequence with no spaces b

  return(sequence)
  #good practice to tell what our code is returning
}

answer <- sapply(6:12, generate_protein) #apply our function over lengths of 6 to 12 amino ac
answer
```

```
[1] "AETKVK"       "WPDDEMS"      "LFAEKELF"     "RFVPYWNTP"    "YMMQPYRIDW"
[6] "SWTDGFYMNKT"  "PGIMLHHIVQTT"
```

Format the values as fasta

```r
cat(paste(">id.", 6:12, "\n", answer, sep = ""), sep = "\n")
```

```
>id.6
AETKVK
>id.7
WPDDEMS
>id.8
LFAEKELF
>id.9
RFVPYWNTP
>id.10
YMMQPYRIDW
>id.11
SWTDGFYMNKT
>id.12
PGIMLHHIVQTT
```

```r
#"\n" is a return carriage, gives us a new line
```