

Class 06

Abel Demoz

Quarto

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

```
1 + 1
```

```
[1] 2
```

My first function :-)

```
add <- function (x, y=1:10) {  
  x+y  
}
```

```
add(1,1)
```

```
[1] 2
```

```
add(x=1, y=100)
```

```
[1] 101
```

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

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

```
add(10)
```

```
[1] 11 12 13 14 15 16 17 18 19 20
```

```
add(10,10)
```

```
[1] 20
```

```
add(1,1)
```

```
[1] 2
```

Write a function to generate DNA that makes a random nucleotide of any seq
Nucleotide Function

```
## generate_DNA <- function() {  
bases <- c("A", "C", "G", "T")  
sample(bases, size=5, replace = TRUE)
```

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

Than is my wee working snipet, now I can make it into a function.

```
generate_dna <- function(lenght) {  
bases <- c("A", "C", "G", "T")  
sequence <- sample(bases, size=lenght, replace = TRUE)  
return(sequence)  
}
```

```
generate_dna()
```

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

bio3d::aa.table

aa3	aa1	mass	formula	name
ALA	ALA	A 71.078	C3 H5 N 01	Alanine
ARG	ARG	R 157.194	C6 H13 N4 01	Arginine
ASN	ASN	N 114.103	C4 H6 N2 02	Asparagine
ASP	ASP	D 114.079	C4 H4 N 03	Aspartic Acid
CYS	CYS	C 103.143	C3 H5 N 01 S	Cystein
GLN	GLN	Q 117.126	C4 H9 N2 02	Glutamine
GLU	GLU	E 128.106	C5 H6 N 03	Glutamic Acid
GLY	GLY	G 57.051	C2 H3 N 01	Glycine
HIS	HIS	H 137.139	C6 H7 N3 01	Histidine
ILE	ILE	I 113.158	C6 H11 N 01	Isoleucine
LEU	LEU	L 113.158	C6 H11 N 01	Leucine
LYS	LYS	K 129.180	C6 H13 N2 01	Lysine
MET	MET	M 131.196	C5 H9 N 01 S	Methionine
PHE	PHE	F 147.174	C9 H9 N 01	Phenylalanine
PRO	PRO	P 97.115	C5 H7 N 01	Proline
SER	SER	S 87.077	C3 H5 N 02	Serine
THR	THR	T 101.104	C4 H7 N 02	Threonine
TRP	TRP	W 186.210	C11 H10 N2 01	Tryptophan
TYR	TYR	Y 163.173	C9 H9 N 02	Tyrosine
VAL	VAL	V 99.131	C5 H9 N 01	Valine
ABA	ABA	X 85.104	C4 H7 N1 01	alpha-aminobutyric acid
ASH	ASH	D 115.087	C4 H5 N 03	Aspartic acid Neutral
CIR	CIR	R 157.170	C6 H11 N3 02	citrulline
CME	CME	C 179.260	C5 H9 N 02 S2	s,s-(2-hydroxyethyl)thiocysteine
CMT	CMT	C 115.154	C4 H5 N 01 S	o-methylcysteine
CSD	CSD	C 134.134	C3 H4 N 03 S	s-cysteinesulfinic acid
CSO	CSO	C 119.142	C3 H5 N 02 S	s-hydroxycysteine
CSW	CSW	C 135.142	C3 H5 N 03 S	cysteine-s-dioxide
CSX	CSX	C 119.142	C3 H5 N 02 S	s-oxy cysteine
CYM	CYM	C 102.135	C3 H4 N 01 S	Cystein Negative
CYX	CYX	C 102.135	C3 H4 N 01 S	Cystein SSbond
DDE	DDE	H 280.346	C13 H22 N5 02	diphthamide
GLH	GLH	E 129.114	C5 H7 N 03	Glutatmic acid Neutral
HID	HID	H 137.139	C6 H7 N3 01	Histidine
HIE	HIE	H 137.139	C6 H7 N3 01	Histidine
HIP	HIP	H 138.147	C6 H8 N3 01	Histidine Positive
HSD	HSD	H 137.139	C6 H7 N3 01	Histidine
HSE	HSE	H 137.139	C6 H7 N3 01	Histidine
HSP	HSP	H 138.147	C6 H8 N3 01	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

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

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

```
generate_protein(5)
```

```
[1] "F" "M" "T" "I" "A"
```

Generate random protein sequences of length 6 to 12

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

```
[[1]]
```

```
[1] "S" "I" "F" "G" "E" "K"
```

```
[[2]]
```

```
[1] "N" "P" "I" "Y" "A" "A" "H"
```

```
[[3]]
```

```
[1] "Y" "L" "A" "V" "N" "V" "L" "H"
```

```
[[4]]
```

```
[1] "W" "H" "V" "R" "I" "H" "S" "R" "V"
```

```
[[5]]  
[1] "D" "I" "I" "T" "H" "I" "V" "Q" "T" "Q"
```

```
[[6]]  
[1] "S" "H" "R" "G" "N" "Y" "R" "N" "I" "K" "G"
```

```
[[7]]  
[1] "Y" "M" "I" "N" "R" "N" "P" "E" "N" "D" "Y" "L"
```

```
paste(c("barry", "Claire", "Chandra"), "Loves R", sep="")
```

```
[1] "barryLoves R" "ClaireLoves R" "ChandraLoves R"
```

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

```
generate_protein(10)
```

```
[1] "WRQYMWYPRR"
```

```
answer <- sapply(6:12, generate_protein)  
answer
```

```
[1] "NFFDYI" "AEFWSGL" "RDVQRMNA" "VLGCTRENQ" "PSAHATWIEK"  
[6] "LVTVMYEHYEP" "LEPHQILMGVIS"
```

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

```
>id.6  
NFFDYI  
>id.7  
AEFWSGL  
>id.8
```

RDVQRMNA
>id.9
VLGCTRENQ
>id.10
PSAHATWIEK
>id.11
LVTVMYEHYEP
>id.12
LEPHQILMGVIS