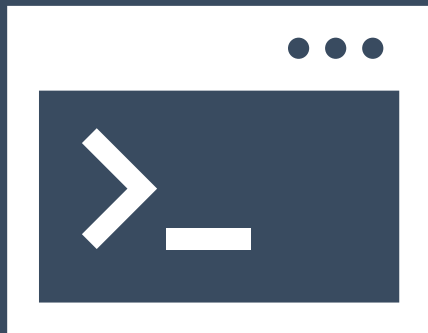


Shell for Bioinformatics

<https://tinyurl.com/hbc-shell-online>



Harvard Chan Bioinformatics Core
in collaboration with
HMS Research Computing



Introductions!





Shannan Ho Sui
Director



Meeta Mistry
Associate Director



Lorena Pantano
*Director of Bioinformatics
Platform*



John Quackenbush
Faculty Advisor



Upen Bhattarai



Heather Wick



Will Gammerdinger



Noor Sohail



Alex Bartlett



Elizabeth
Partan



Emma Berdan



James Billingsley



Zhu Zhuo



Maria Simoneau



Shannan Ho Sui
Director



Meeta Mistry
Associate Director



Lorena Pantano
*Director of Bioinformatics
Platform*



John Quackenbush
Faculty Advisor



Upen Bhattarai



Heather Wick



Will Gammerdinger



Noor Sohail



Alex Bartlett



Elizabeth
Partan



Emma Berdan



James Billingsley



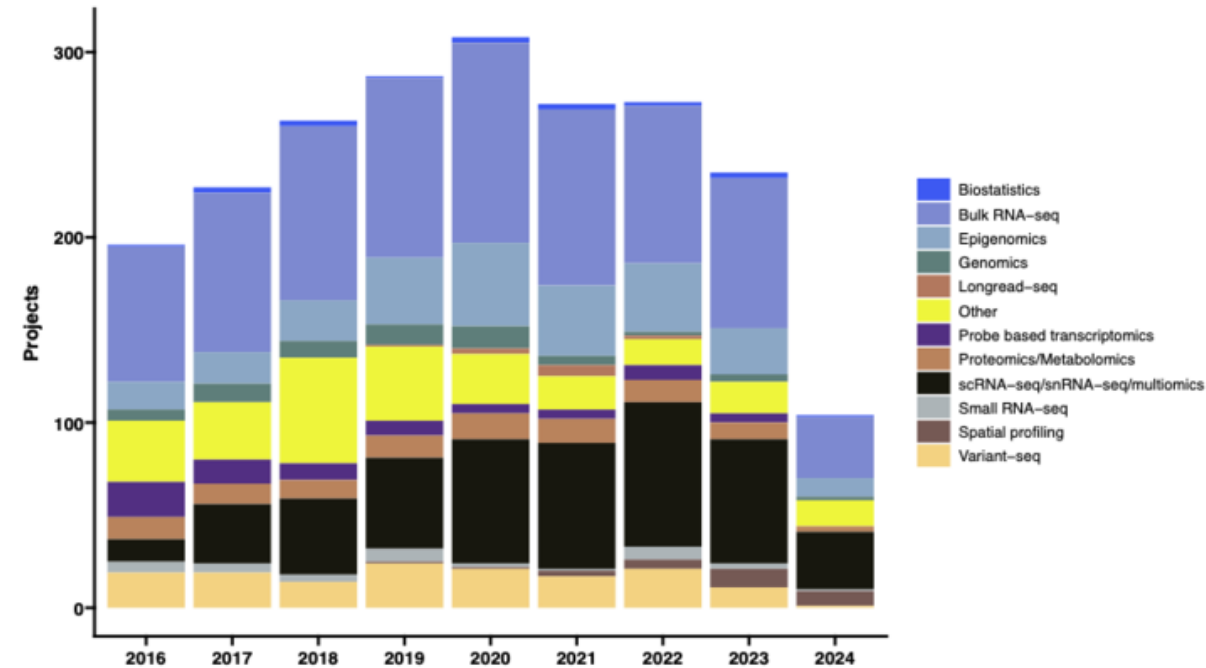
Zhu Zhuo



Maria Simoneau

Consulting

- ❖ Transcriptomics: Bulk, single cell, small RNA
- ❖ Epigenomics: ChIP-seq, CUT&RUN, ATAC-seq, DNA methylation
- ❖ Variant discovery: WGS, resequencing, exome-seq and CNV
- ❖ Multiomics integration
- ❖ Spatial biology
- ❖ Experimental design and grant support



Consulting

- ❖ Transcriptomics: Bulk, single cell, small RNA
- ❖ Epigenomics: ChIP-seq, CUT&RUN, ATAC-seq, DNA methylation
- ❖ Variant discovery: WGS, resequencing, exome-seq and CNV
- ❖ Multiomics integration
- ❖ Spatial biology
- ❖ Experimental design and grant support



HARVARD
T.H. CHAN
SCHOOL OF PUBLIC HEALTH

NIEHS



THE HARVARD CLINICAL
AND TRANSLATIONAL
SCIENCE CENTER



HARVARD
MEDICAL SCHOOL

Training

A key component of the HBC's mission is its training initiative. Our dedicated training team holds workshop to help researchers at Harvard better understand analytical methods for NGS data.

[HBC's training team](#) is made up of four PhD-level scientists who devote substantial time to material development, training and community building/outreach. All members of the training team also participate in consultations on research projects to ensure they remain up-to-date on current best practices in NGS analysis.

Our hands-on workshops focus on **basic data skills** and **analysis of high-throughput sequencing data**, with an emphasis on **experimental design**, current **best practices** and **reproducibility**. Our workshops are designed for **wet-lab biologists** aiming to independently design sequencing-based experiments and analysing the resulting data.

We offer three types of workshops:

1. [Short, 3-hour monthly workshops](#) (*Current topics in bioinformatics*)
2. [Basic Data Skills](#)**
3. [Advanced Topics: Analysis of high-throughput sequencing \(NGS\) data](#)**

***The basic data skills workshops serve as the foundation for the advanced workshops.*

<https://bioinformatics.sph.harvard.edu/training>

Training

A key component of the HBC's mission is its training initiative. Our dedicated training team holds workshops for researchers at Harvard better understand analytical methods for NGS data.

[HBC's training team](#) is made up of four PhD-level scientists who devote substantial time to material development, training and community building/outreach. All members of the training team also participate in consultative research projects to ensure they remain up-to-date on current best practices in NGS analysis.

Our hands-on workshops focus on **basic data skills** and **analysis of high-throughput sequencing** with an emphasis on **experimental design**, current **best practices** and **reproducibility**. Our workshops are designed for **wet-lab biologists** aiming to independently design sequencing-based experiments and analysing sequencing data.

We offer three types of workshops:

1. [Short, 3-hour monthly workshops](#) (*Current topics in bioinformatics*)
2. [Basic Data Skills](#)**
3. [Advanced Topics: Analysis of high-throughput sequencing \(NGS\) data](#)**

***The basic data skills workshops serve as the foundation for the advanced workshops.*

<https://bioinformatics.sph.harvard.edu/training>



HARVARD
T.H. CHAN
SCHOOL OF PUBLIC HEALTH

DF/HCC
DANA-FARBER / HARVARD CANCER CENTER

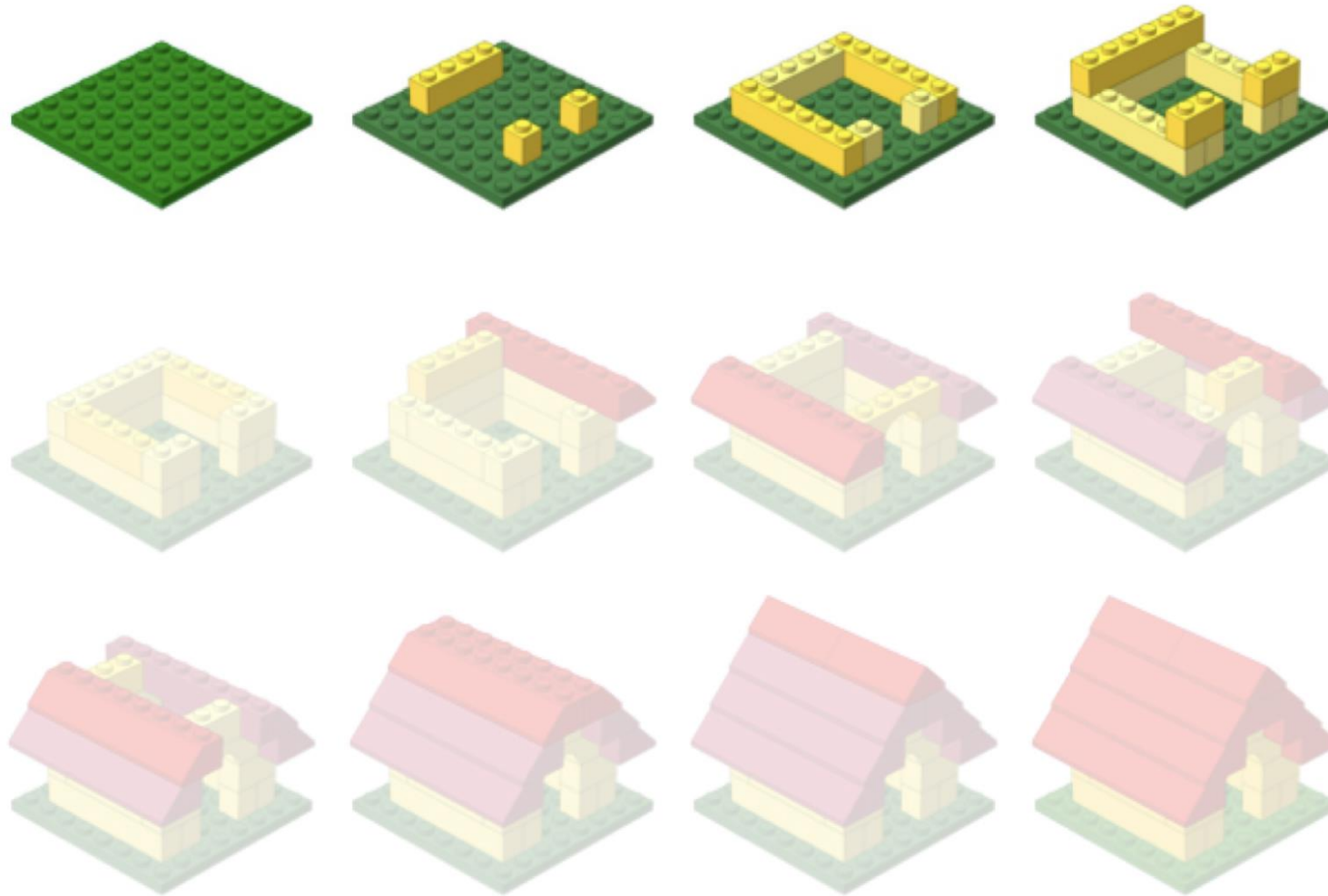


THE HARVARD CLINICAL
AND TRANSLATIONAL
SCIENCE CENTER



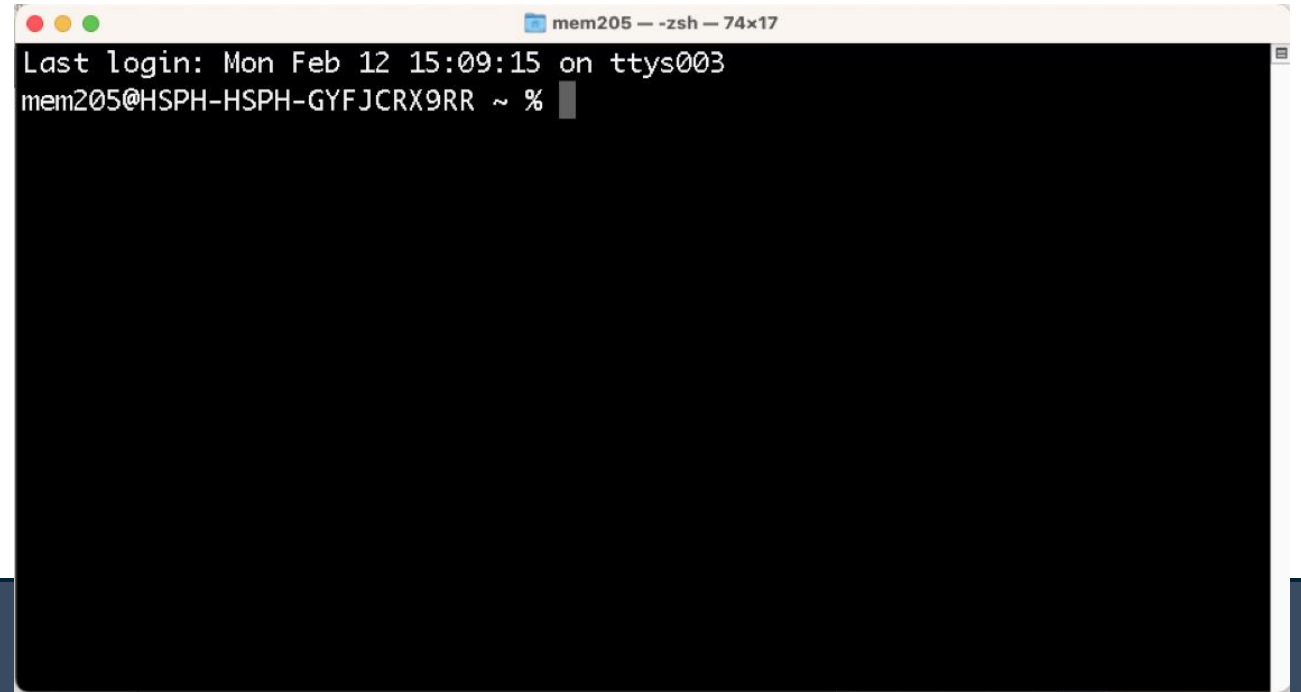
HARVARD
MEDICAL SCHOOL

Workshop scope



Learning Bioinformatics

What is shell?



- ❖ Shell is a program that allows users to control Unix/Linux OS with text commands

Terminology

- ❖ **Unix/Linux** - The operating systems of High Performance Computers (HPC)

Terminology

- ❖ **Unix/Linux** - The operating systems of High Performance Computers (HPC)
- ❖ **Shell** - A program that allows users to control Unix/Linux OS with text commands

Terminology

- ❖ **Unix/Linux** - The operating systems of High Performance Computers (HPC)
- ❖ **Shell** - A program that allows users to control Unix/Linux OS with text commands
- ❖ **Bash** - The most prevalent kind of shell

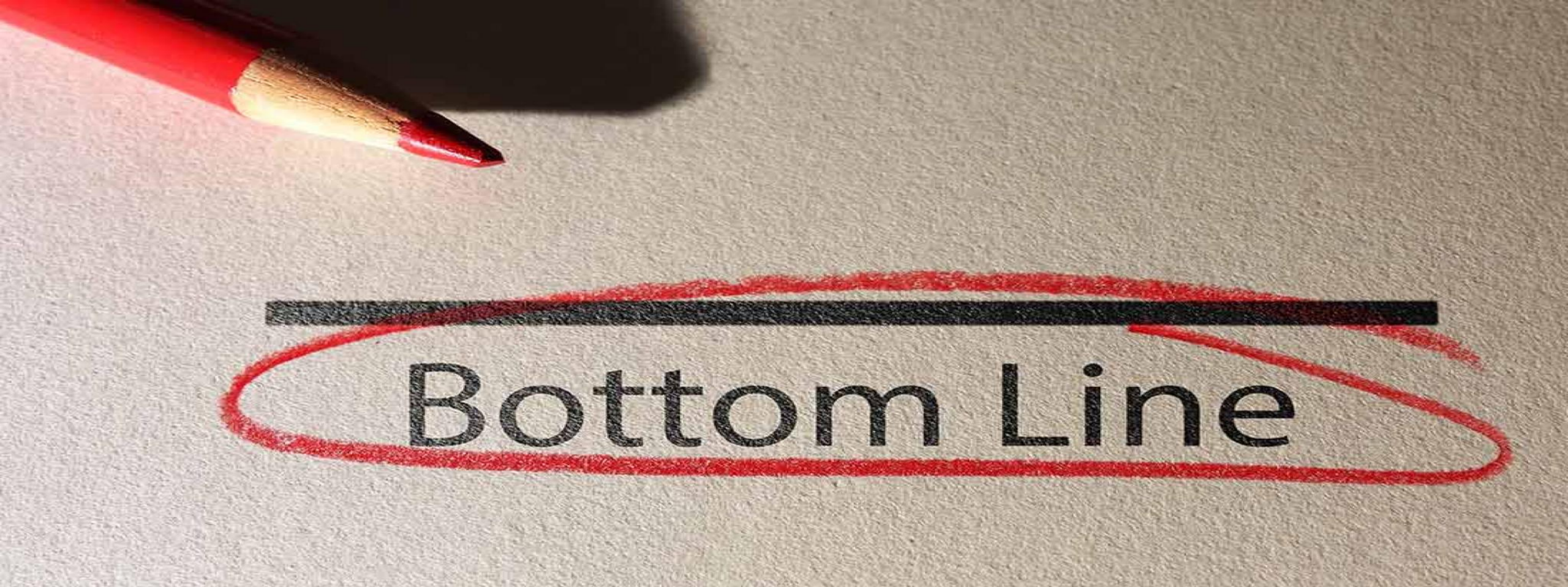


Image source: Balboa Capital Blog

If you plan to process raw high throughput sequencing data yourself, you will need to learn shell.

1. You need more resources than what is available on your laptop

- ❖ Sequence data files are LARGE
- ❖ Processing these data require increased CPU and memory
- ❖ High performance compute clusters have the necessary resources!



2. Many bioinformatics tools are only available as command-line tools

10XGenomics/
cellranger

10x Genomics Single Cell Analysis

10x
GENOMICS™

staraligner



SAMtools

3. Many genomics filetypes are binary



- ❖ Binary files are not human readable
- ❖ Binary files need an interpreter

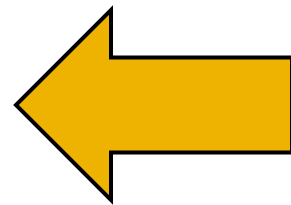
4. There are many useful commands that can help work with enormous data files

❖ Commands for easily viewing files: less, cat, head, tail

```
0 ##gff-version 3.2.1
1 ##sequence-region ctg123 1 1497228
2 ctg123 . gene 1000 9000 . + . ID=gene00001;Name=EDEN
3 ctg123 . TF_binding_site 1000 1012 . + . ID=tfbs00001;Parent=gene00001
4 ctg123 . mRNA 1050 9000 . + . ID=mRNA00001;Parent=gene00001;Name=EDEN.1
5 ctg123 . mRNA 1050 9000 . + . ID=mRNA00002;Parent=gene00001;Name=EDEN.2
6 ctg123 . mRNA 1300 9000 . + . ID=mRNA00003;Parent=gene00001;Name=EDEN.3
7 ctg123 . exon 1300 1500 . + . ID=exon00001;Parent=mRNA00003
8 ctg123 . exon 1050 1500 . + . ID=exon00002;Parent=mRNA00001,mRNA00002
9 ctg123 . exon 3000 3902 . + . ID=exon00003;Parent=mRNA00001,mRNA00003
10 ctg123 . exon 5000 5500 . + . ID=exon00004;Parent=mRNA00001,mRNA00002,mRNA00003
11 ctg123 . exon 7000 9000 . + . ID=exon00005;Parent=mRNA00001,mRNA00002,mRNA00003
12 ctg123 . CDS 1201 1500 . + 0 ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
13 ctg123 . CDS 3000 3902 . + 0 ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
14 ctg123 . CDS 5000 5500 . + 0 ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
15 ctg123 . CDS 7000 7600 . + 0 ID=cds00001;Parent=mRNA00001;Name=edenprotein.1
16 ctg123 . CDS 1201 1500 . + 0 ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
17 ctg123 . CDS 5000 5500 . + 0 ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
18 ctg123 . CDS 7000 7600 . + 0 ID=cds00002;Parent=mRNA00002;Name=edenprotein.2
19 ctg123 . CDS 3301 3902 . + 0 ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
20 ctg123 . CDS 5000 5500 . + 1 ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
21 ctg123 . CDS 7000 7600 . + 1 ID=cds00003;Parent=mRNA00003;Name=edenprotein.3
22 ctg123 . CDS 3391 3902 . + 0 ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
23 ctg123 . CDS 5000 5500 . + 1 ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
24 ctg123 . CDS 7000 7600 . + 1 ID=cds00004;Parent=mRNA00003;Name=edenprotein.4
```

5. Automation is the name of the game

- ❖ Launch many jobs with one command
- ❖ Code is used and reused to iterate tasks over multiple files
- ❖ Parallelization to complete tasks using multiple cores and increase speed!



This could be you
watching your analysis
run!

Learning Objectives



- ❖ Navigate around the command line interface (bash/shell)
- ❖ Create and manipulate text files
- ❖ Submit jobs to a high-performance computing cluster

Logistics



Course schedule

Workshop Schedule

Day 1

Time	Topic	Instructor
9:30 - 10:10	Workshop introduction	Noor
10:10 - 11:40	Introduction to Shell	Heather
11:40 - 12:00	Overview of self-learning materials and homework submission	Noor

Before the next class:

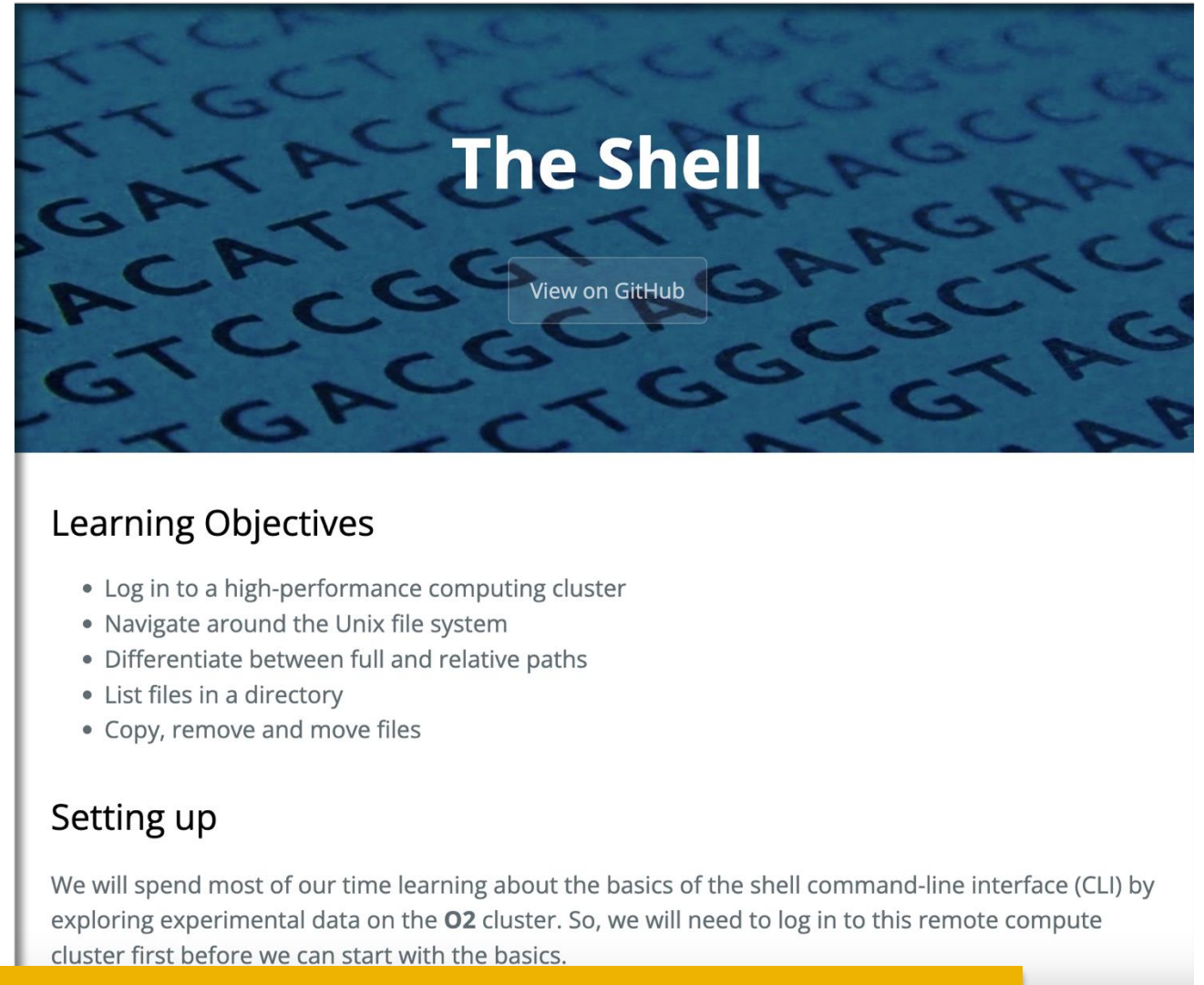
I. Please **study the contents** and **work through all the code** within the following lessons:

1. [Wildcards and shortcuts in Shell](#)
[Click here for a preview of this lesson](#)
2. [Examining and creating files](#)
[Click here for a preview of this lesson](#)
3. [Searching and redirection](#)
[Click here for a preview of this lesson](#)

<https://tinyurl.com/hbc-shell-online>

Course materials

- ❖ We continuously update our materials to reflect changes in the field/software

A preview of the 'The Shell' course materials. The top section features a blue background with a repeating pattern of DNA base pairs (A, T, C, G) in a lighter blue color. The title 'The Shell' is written in a large, white, sans-serif font. Below the title is a small, rounded rectangular button with the text 'View on GitHub'. The bottom section has a white background and contains the text 'Learning Objectives' followed by a bulleted list of five items: 'Log in to a high-performance computing cluster', 'Navigate around the Unix file system', 'Differentiate between full and relative paths', 'List files in a directory', and 'Copy, remove and move files'. Below this is the text 'Setting up' followed by a paragraph: 'We will spend most of our time learning about the basics of the shell command-line interface (CLI) by exploring experimental data on the O2 cluster. So, we will need to log in to this remote compute cluster first before we can start with the basics.'

<https://tinyurl.com/hbc-shell-online>

Single Screen & 3 Windows

The image shows a video conference interface with three participants: Mary Piper (Co-host, me), Jihe Liu (Host), and Troubleshooter (Radhika) (Co-host). The main window displays the RStudio environment, which is split into several panes:

- Source Editor:** Contains R code for assignment operators, functions, and rounding. The code is as follows:

```
1 # Assignment operator
2 x <- 3
3
4 # Functions
5 getwd()
6
7 sqrt(81)
8
9 round(3.14159)
10 ?round
11
```
- Console:** Shows the output of the R code:

```
> x <- 3
> # Functions
> getwd()
[1] "/Users/mariyaper/Desktop/R-testing"
> sqrt(81)
[1] 9
> round(3.14159)
[1] 3
> ?round
>
```
- Environment:** Displays the current environment, showing the variable `x` with the value `3`.
- Documentation:** Shows the R documentation for the `round` function, titled "Rounding of Numbers".

The video conference interface also includes a top bar with the text "You are viewing Jihe Liu's screen" and a bottom bar with controls for Unmute, Stop Video, Invite, Share Screen, and Reactions.

Single Screen & 3 Windows

Zoom

Our Recommendation

```
> x <- 3
> # Functions
> getwd()
[1] "/Users/mariyaper/Desktop/R-testing"
> sqrt(81)
[1] 9
> round(3.14159)
[1] 3
> ?round
>
```

Rounding of Numbers

Description

ceiling takes a single numeric argument x and returns a numeric vector containing the smallest integers not less than the corresponding elements of x.

floor takes a single numeric argument x and returns a numeric vector containing the largest integers not greater than the corresponding elements of x.

trunc takes a single numeric argument x and returns a numeric vector containing the integers formed by truncating the values in x toward 0.

round rounds the values in its first argument to the specified number of decimal places (default 0). See 'Details' about "round to even" when rounding off a 5.

signif rounds the values in its first argument to the specified number of significant digits.

Usage

```
ceiling(x)
floor(x)
trunc(x, ...)
```

Single Screen & 3 Windows

The image illustrates a single-screen setup for a video conference, RStudio, and a web browser. The video conference window at the top left shows three participants: Mary Piper, Troubleshooter, and Jihe Liu. The RStudio window at the bottom left shows a script editor with R code, a console, and a viewer pane. The web browser window at the top right shows a page titled 'Web Browser' with a large green text overlay.

Participants (3)

- Mary Piper (Co-host, me)
- Jihe Liu (Host)
- Troubleshooter (Radhika) (Co-host)

Web Browser

RStudio

```
1 # Assignment operator
2 x <- 3
3
4 # Functions
5 getwd()
6
7 sqrt(81)
8
9 round(3.14159)
10 ?round
11
```

Console

```
> x <- 3
> # Functions
> getwd()
[1] "/Users/mariyapiper/Desktop/R-testing"
> sqrt(81)
[1] 9
> round(3.14159)
[1] 3
> ?round
>
```

Viewer

Values

x
3

Rounding of Numbers

Description

integers not less than the corresponding elements of x.

floor: takes a single numeric argument x and returns a numeric vector containing the largest integers not greater than the corresponding elements of x.

trunc: takes a single numeric argument x and returns a numeric vector containing the integers formed by truncating the values in x toward 0.

round: rounds the values in its first argument to the specified number of decimal places (default 0). See 'Details' about "round to even" when rounding off a 5.

signif: rounds the values in its first argument to the specified number of significant digits.

Usage

```
ceiling(x)
floor(x)
trunc(x, ...)
```

*Our
Recommendation*

Single Screen & 3 Windows

Our Recommendation

R Studio

```
1 # Assignment operator
2 x <- 3
3
4 # Functions
5 getwd()
6
7 sqrt(81)
8
9 round(3.14159)
10 ?round
11
```

Console

```
> x <- 3
> # Functions
> getwd()
[1] "/Users/mariyaper/Desktop/R-testing"
> sqrt(81)
[1] 9
> round(3.14159)
[1] 3
> ?round
>
```

Values

Variable	Value
x	3

R Documentation

Rounding of Numbers

Description

ceiling takes a single numeric argument x and returns a numeric vector containing the smallest integers not less than the corresponding elements of x.

floor takes a single numeric argument x and returns a numeric vector containing the largest integers not greater than the corresponding elements of x.

trunc takes a single numeric argument x and returns a numeric vector containing the integers formed by truncating the values in x toward 0.

round rounds the values in its first argument to the specified number of decimal places (default 0). See 'Details' about "round to even" when rounding off a 5.

signif rounds the values in its first argument to the specified number of significant digits.

Usage

```
ceiling(x)
floor(x)
trunc(x, ...)
```


Single Screen & 3 Windows

The image is a composite of three overlapping windows, each with a colored border: blue for Zoom, green for a web browser, and pink for R Studio.

Zoom Window (Blue Border): Shows a video call with three participants: Mary Piper, Troubleshooter (...), and Jihe Liu. Below the video feed is the R console output from a session. The code and output are as follows:

```
483  
484  
485 getwd()  
486  
487 # square root function  
488 sqrt(81)  
489  
490 # round function  
491 round(3.14159)  
492 ?round  
493  
494  
495
```

Web Browser Window (Green Border): Shows a page titled "Web Browser" with a URL bar displaying "https://hsph-bioc...".

R Studio Window (Pink Border): Shows the R Studio interface with a script editor, environment pane, and console. The script editor contains the following code:

```
1 # Assignment operator  
2 x <- 3  
3  
4 # Functions  
5 getwd()  
6  
7 sqrt(81)  
8  
9 round(3.14159)  
10 ?round  
11
```

The console output shows the execution of the code:

```
> x <- 3  
> # Functions  
> getwd()  
[1] "/Users/mariyaper/Desktop/R-testing"  
> sqrt(81)  
[1] 9  
> round(3.14159)  
[1] 3  
> ?round  
>
```

The environment pane shows the variable `x` with the value `3`. The help pane shows the documentation for the `round` function.

Our Recommendation

R Studio

Course participation

- ❖ Mandatory review of self-learning lessons and assignments
- ❖ Attendance required for all classes
- ❖ Your questions and active participation drive learning
- ❖ **We look forward to all of your questions!**



Course participation

- ❖ At-home lessons and exercises after each session
- ❖ Cover material not previously discussed
- ❖ Provides us feedback to help pace the course appropriately
- ❖ 3-5 hours to complete
- ❖ Homework load is heavier in the beginning of this workshop series and tapers off

Using AI for Assignments

❖ Do

- ❖ Try to resolve error messages with it
- ❖ Test code written by AI on a dataset where you have expected results
- ❖ Take the time to review the generated code line-by-line

❖ Don't

- ❖ Implement it in replacement to learning
- ❖ Write code that you don't understand
- ❖ Assume the output from an AI process is correct

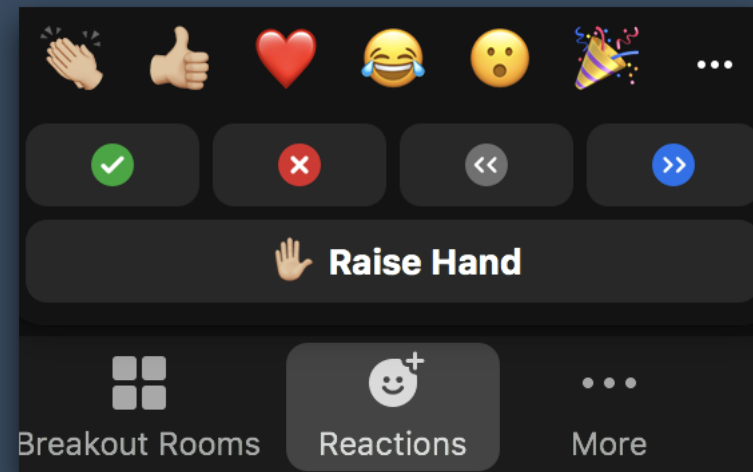
Odds & Ends

❖ Quit/minimize all applications that are not required for class

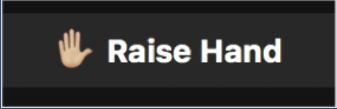
❖ Are you all set?

❖  = "agree", "I'm all set"

❖  = "disagree", "I need help"



Odds & Ends

- ❖ Questions for the presenter?
 - ❖ Post the question in the Chat window OR
 - ❖  when the presenter asks for questions
 - ❖ Let the Troubleshooter know

Odds & Ends

❖ Questions for the presenter?

- ❖ Post the question in the Chat window OR

- ❖  when the presenter asks for questions

- ❖ Let the Troubleshooter know

❖ Technical difficulties with software?

- ❖ Start a private chat with the Troubleshooter with a description of the problem

Thanks!

- ❖ Kathleen Chappell and Andy Bergman from HMS-RC
- ❖ Data Carpentry

These materials have been developed by members of the teaching team at the Harvard Chan Bioinformatics Core (HBC). These are open access materials distributed under the terms of the Creative Commons Attribution license (CC BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Contact Us

- ❖ *HBC training team:* hbctraining@hsph.harvard.edu
- ❖ *HBC consulting:* bioinformatics@hsph.harvard.edu
- ❖ *O2 (HMS-RC):* rchelp@hms.harvard.edu