Unix/Linux Tutorial for Beginners
Session I

Mihaela Martis

NBIS & Faculty of Medicine and Health Sciences
Division Cell Biology, IKE

Software/Hardware check

- can everyone hear me?
- can everyone see the slides?
- mute your microphone

A brief introduction

- this course is a joint effort of SciLifeLab, NBIS, and University of Ljubljana:

A brief introduction

- this course is a joint effort of SciLifeLab, NBIS, and University of Ljubljana:
  - Elixir → European life-science infrastructure, which coordinates, integrates and sustains bioinformatics resources

A brief introduction

- this course is a joint effort of SciLifeLab, NBIS, and University of Ljubljana:
  - Elixir → European life-science infrastructure, which coordinates, integrates and sustains bioinformatics resources
  - University of Ljubljana → provides tools infrastructure, data resources, and compute and training provision for specific biological domains

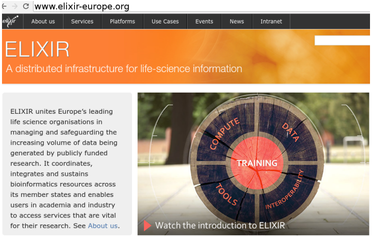
A brief introduction

- this course is a joint effort of SciLifeLab, NBIS, and University of Ljubljana:
    - Elixir → European life-science infrastructure, which coordinates, integrates and sustains bioinformatics resources
    - University of Ljubljana → provides tools infrastructure, data resources, and compute and training provision for specific biological domains
    - NBIS → provides bioinformatics support services, training events, and computational ressources

A brief introduction

- this course is a joint effort of SciLifeLab, NBIS, and University of Ljubljana:
    - Elixir → European life-science infrastructure, which coordinates, integrates and sustains bioinformatics resources
    - University of Ljubljana → provides tools infrastructure, data resources, and compute and training provision for specific biological domains
    - NBIS → provides bioinformatics support services, training events, and computational ressources
    - SciLifeLab → national center for molecular biosciences with focus on health and environmental research

Website



www.elixir-europe.org

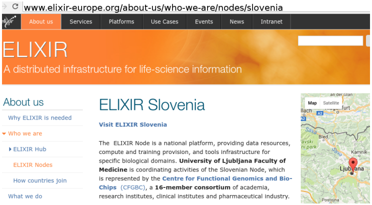
About us | Services | Platforms | Use Cases | Events | News | Intranet

ELIXIR

A distributed infrastructure for life-science information

ELIXIR unites Europe's leading life science organisations in managing and safeguarding the increasing volume of data being generated by publicly funded research. It coordinates, integrates and sustains bioinformatics resources across its member states and enables users in academia and industry to access services that are vital for their research. See About us.

Watch the introduction to ELIXIR



www.elixir-europe.org/about-us/who-we-are/nodes/slovenia

About us | Services | Platforms | Use Cases | Events | News | Intranet

ELIXIR

A distributed infrastructure for life-science information

About us

- Why ELIXIR is needed
- Who we are
  - ELIXIR Hub
  - ELIXIR Nodes
  - How countries join
- What we do

ELIXIR Slovenia

Visit ELIXIR Slovenia

The ELIXIR Node is a national platform, providing data resources, compute and training provision, and tools infrastructure for specific biological domains. **University of Ljubljana Faculty of Medicine** is coordinating activities of the Slovenian Node, which is represented by the **Centre for Functional Genomics and Bio-Chips** (CFGBC), a **16-member consortium** of academia, research institutes, clinical institutes and pharmaceutical industry.

nbis.se

Support | Infrastructure | Training | News | Events | About

NBIS

NATIONAL BIOINFORMATICS INFRASTRUCTURE SWEDEN

NBIS is a distributed national bioinformatics infrastructure, supporting life sciences in Sweden

Support | Infrastructure | Training

www.scilifelab.se

About us | Contact | Events | News | Press | Staff | Vacancies | På svenska | Search

SciLifeLab | TECHNOLOGIES & SERVICES | RESEARCH | EDUCATION | COLLABORATION | DATA

Infrastructure. Our technologies and services, available to researchers in all of Sweden, are managed by 441 persons with excellent knowledge and competence in their field.

NEWS » View all

2017-03-07 - NEWS
New treatment attacks liver disease and type 2 diabetes

2017-03-06 - NEWS
Nominate your favorite speaker

UPCOMING EVENTS » View all

2017-02-06 - COURSE
Introduction to high throughput screening, 2 ECTS

2017-03-13 - COURSE
RNA-seq, 1 hp

SciLifeLab is a national center for molecular biosciences with focus on health and environmental research.

The team

Linköping

Malin Larsson

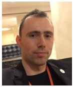
Mihaela Martis

Umeå

Allison Churcher

Jeanette Tångrot

Ljubljana



Aleš Papič

Patrik Pušnik

Brane Leskošek

Jure Dimec

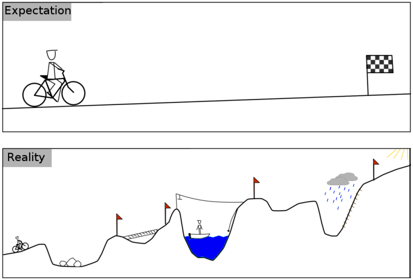
The participants

- 41 participants from 12 universities in 8 cities
- 30 participants in 3 classrooms:
  - TA in Ljubljana: Aleš & Patrik
  - TA in Umeå: Jeanette
  - TA in Linköping: Mihaela
- 11 remotely connected participants:
  - TA: Allison (Monday), Malin (Tuesday, Wednesday)

The goals

- Become familiar with the Unix/Linux operating system.
- Get comfortable with the command-line environment
- Learn powerful commands to process/explore your data.
- Be able to find documentation about individual commands.

The expectations



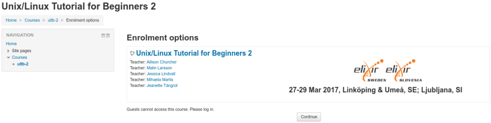
- will not cover bioinformatics tools
- frustration
- the exercises have some overhead, it's not expected that you do all
- keep playing to get proficient

What do you need to do before we start?

1. start your computer

What do you need to do before we start?

1. start your computer
2. open the e-learning platform:
   https://elixir.mf.uni-lj.si/course/view.php?id=17



Unix/Linux Tutorial for Beginners 2

Enrolment options

Unix/Linux Tutorial for Beginners 2

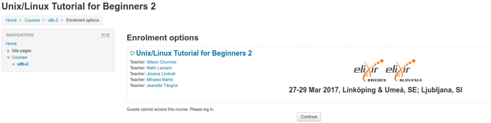
Teacher: Allison Churcher
Teacher: Malin Larsson
Teacher: Jessica Lindvall
Teacher: Mihaela Martis
Teacher: Jeanette Tångrot

27-29 Mar 2017, Linköping & Umeå, SE; Ljubljana, SI

Guests cannot access this course. Please log in.

What do you need to do before we start?

1. start your computer
2. open the e-learning platform: https://elixir.mf.uni-lj.si/course/view.php?id=17



Unix/Linux Tutorial for Beginners 2

Enrolment options

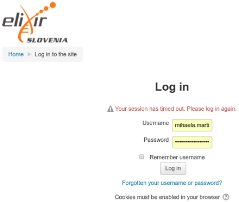
Unix/Linux Tutorial for Beginners 2

Teacher: Allison Churcher
Teacher: Malin Larsson
Teacher: Jessica Lindvall
Teacher: Mihaela Martis
Teacher: Jeanette Tångrot

27-29 Mar 2017, Linköping & Umeå, SE; Ljubljana, SI

Guests cannot access this course. Please log in.

3. log in to the platform


Home ► Log in to the site
Log in
Forgotten your username or password?

Log in

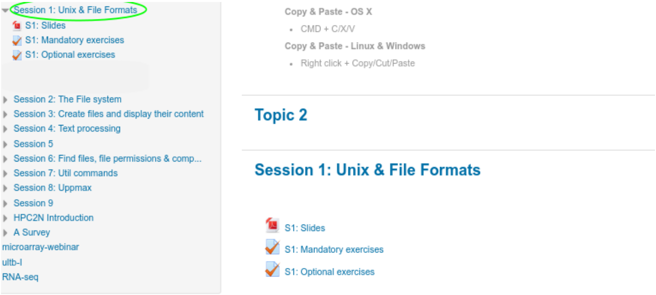
Your session has timed out. Please log in again.

Username: mihaela.marti
Password: ••••••••••••••••

□ Remember username

Cookies must be enabled in your browser

The e-learning platform



- Session 1: Unix & File Formats
  - S1: Slides
  - S1: Mandatory exercises
  - S1: Optional exercises
- Session 2: The File system
- Session 3: Create files and display their content
- Session 4: Text processing
- Session 5
- Session 6: Find files, file permissions & comp...
- Session 7: Util commands
- Session 8: Uppmax
- Session 9
- HPC2N Introduction
- A Survey

microarray-webinar

ultb-I

RNA-seq

Copy & Paste - OS X

- CMD + C/X/V

Copy & Paste - Linux & Windows

- Right click + Copy/Cut/Paste

Topic 2

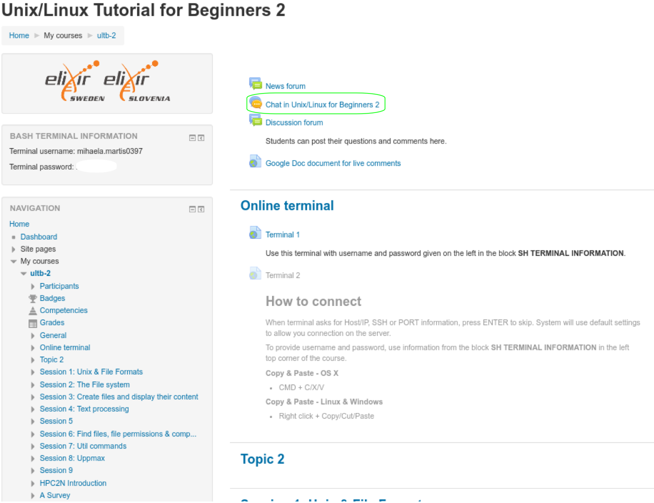
Session 1: Unix & File Formats

S1: Slides

S1: Mandatory exercises

S1: Optional exercises

The e-learning platform



Unix/Linux Tutorial for Beginners 2

Home ▶ My courses ▶ ultb-2

BASH TERMINAL INFORMATION

Terminal username: mihaela.martis0397

Terminal password: [redacted]

NAVIGATION

- Home
  - Dashboard
  - Site pages
  - My courses
    - ultb-2
      - Participants
      - Badges
      - Competencies
      - Grades
      - General
      - Online terminal
      - Topic 2
      - Session 1: Unix & File Formats
      - Session 2: The File system
      - Session 3: Create files and display their content
      - Session 4: Text processing
      - Session 5
      - Session 6: Find files, file permissions & comp...
      - Session 7: Util commands
      - Session 8: Uppmax
      - Session 9
      - HPC2N Introduction
      - A Survey

News forum

Chat in Unix/Linux for Beginners 2

Discussion forum

Students can post their questions and comments here.

Google Doc document for live comments

Online terminal

Terminal 1

Use this terminal with username and password given on the left in the block **SH TERMINAL INFORMATION**.

Terminal 2

How to connect

When terminal asks for Host/IP, SSH or PORT information, press ENTER to skip. System will use default settings to allow you connection on the server.

To provide username and password, use information from the block **SH TERMINAL INFORMATION** in the left top corner of the course.

Copy & Paste - OS X

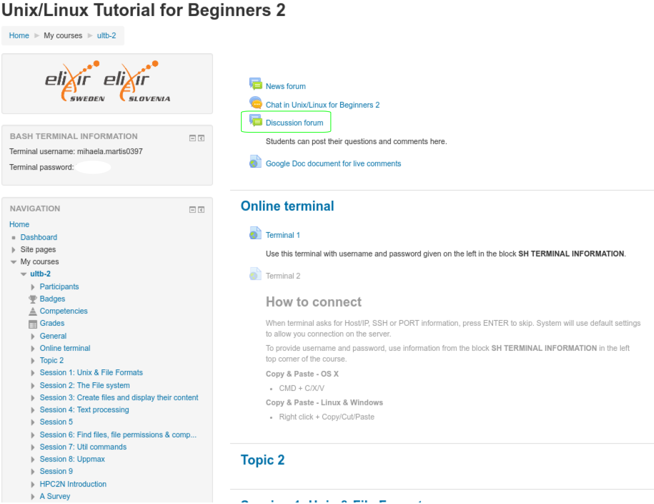
- CMD + C/X/V

Copy & Paste - Linux & Windows

- Right click + Copy/Cut/Paste

Topic 2

The e-learning platform



Unix/Linux Tutorial for Beginners 2

Home ▶ My courses ▶ ultb-2

BASH TERMINAL INFORMATION

Terminal username: mihaela.martis0397

Terminal password: [redacted]

NAVIGATION

- Home
- Dashboard
- Site pages
- My courses
  - ultb-2
    - Participants
    - Badges
    - Competencies
    - Grades
    - General
    - Online terminal
    - Topic 2
    - Session 1: Unix & File Formats
    - Session 2: The File system
    - Session 3: Create files and display their content
    - Session 4: Text processing
    - Session 5
    - Session 6: Find files, file permissions & comp...
    - Session 7: Util commands
    - Session 8: Uppmax
    - Session 9
    - HPC2N Introduction
    - A Survey

News forum

Chat in Unix/Linux for Beginners 2

Discussion forum

Students can post their questions and comments here.

Google Doc document for live comments

Online terminal

Terminal 1

Use this terminal with username and password given on the left in the block **SH TERMINAL INFORMATION**.

Terminal 2

How to connect

When terminal asks for Host/IP, SSH or PORT information, press ENTER to skip. System will use default settings to allow you connection on the server.

To provide username and password, use information from the block **SH TERMINAL INFORMATION** in the left top corner of the course.

Copy & Paste - OS X

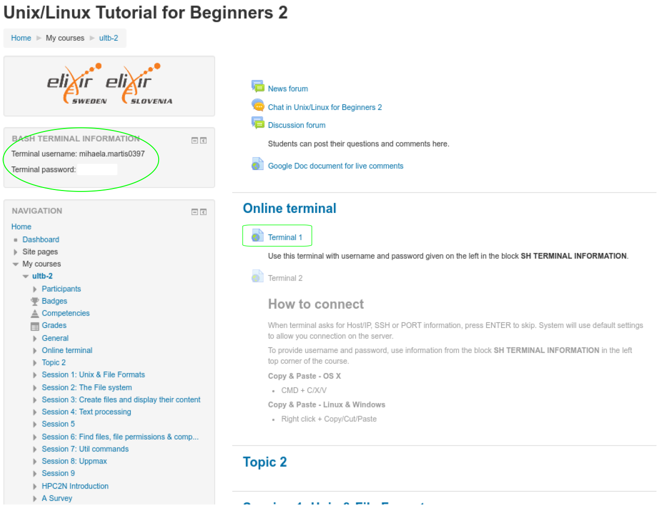
- CMD + C/X/V

Copy & Paste - Linux & Windows

- Right click + Copy/Cut/Paste

Topic 2

The e-learning platform



Unix/Linux Tutorial for Beginners 2

Home ▶ My courses ▶ ultb-2

BASH TERMINAL INFORMATION

Terminal username: mihaela.martis0397

Terminal password:

NAVIGATION

- Home
  - Dashboard
  - Site pages
  - My courses
    - ultb-2
      - Participants
      - Badges
      - Competencies
      - Grades
      - General
      - Online terminal
      - Topic 2
      - Session 1: Unix & File Formats
      - Session 2: The File system
      - Session 3: Create files and display their content
      - Session 4: Text processing
      - Session 5
      - Session 6: Find files, file permissions & comp...
      - Session 7: Util commands
      - Session 8: Uppmax
      - Session 9
      - HPC2N Introduction
      - A Survey

News forum

Chat in Unix/Linux for Beginners 2

Discussion forum

Students can post their questions and comments here.

Google Doc document for live comments

Online terminal

Terminal 1

Use this terminal with username and password given on the left in the block **SH TERMINAL INFORMATION**.

Terminal 2

How to connect

When terminal asks for Host/IP, SSH or PORT information, press ENTER to skip. System will use default settings to allow you connection on the server.

To provide username and password, use information from the block **SH TERMINAL INFORMATION** in the left top corner of the course.

Copy & Paste - OS X

- CMD + C/X/V

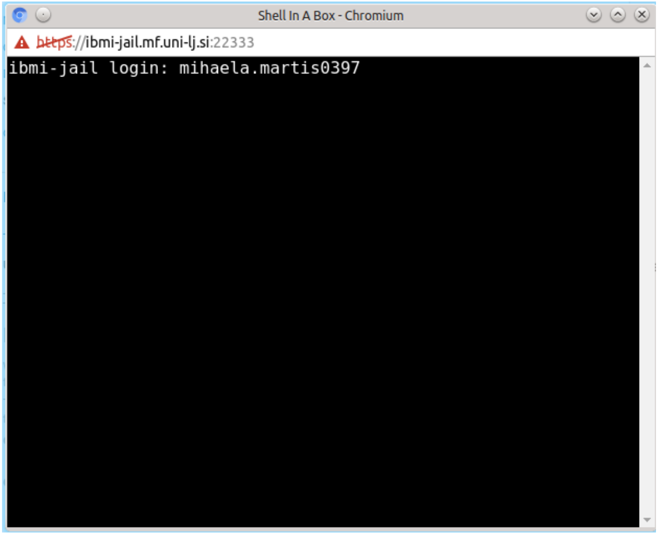
Copy & Paste - Linux & Windows

- Right click + Copy/Cut/Paste

Topic 2

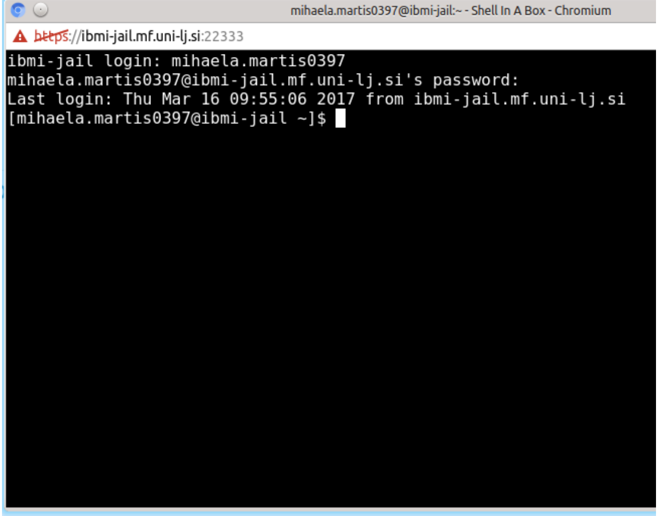
Session 1: Unix & File Formats

The terminal


Shell In A Box - Chromium
~~https~~://ibmi-jail.mf.uni-lj.si:22333

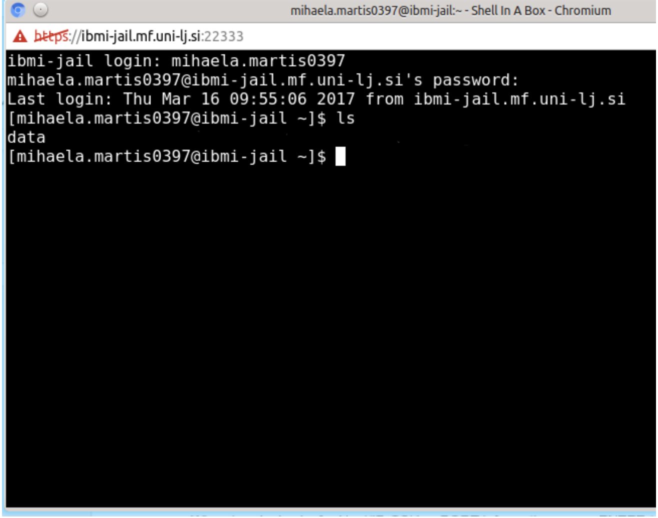
```
ibmi-jail login: mihaela.martis0397
```

The terminal


mihaela.martis0397@ibmi-jail:~ - Shell In A Box - Chromium
~~https~~://ibmi-jail.mf.uni-lj.si:22333

```
ibmi-jail login: mihaela.martis0397
mihaela.martis0397@ibmi-jail.mf.uni-lj.si's password:
Last login: Thu Mar 16 09:55:06 2017 from ibmi-jail.mf.uni-lj.si
[mihaela.martis0397@ibmi-jail ~]$
```

The terminal

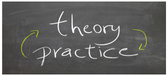

mihaela.martis0397@ibmi-jail:~ - Shell In A Box - Chromium
~~https~~://ibmi-jail.mf.uni-lj.si:22333

```
ibmi-jail login: mihaela.martis0397
mihaela.martis0397@ibmi-jail.mf.uni-lj.si's password:
Last login: Thu Mar 16 09:55:06 2017 from ibmi-jail.mf.uni-lj.si
[mihaela.martis0397@ibmi-jail ~]$ ls
data
[mihaela.martis0397@ibmi-jail ~]$ █
```

What should be now open on your computer?

- video-conference (pexip – only for the remote participants)
- e-learning platform
- terminal
- chat

Today's schedule



$9^{00} - 10^{30}$	*Getting started & file formats*
$10^{30} - 10^{40}$	**Coffee break**
$10^{40} - 12^{10}$	*Filesystem*
$12^{10} - 13^{10}$	**Lunch**
$13^{10} - 15^{00}$	*Text processing I*
$15^{00} - 15^{20}$	**Coffee break**
$15^{20} - 17^{00}$	*Text processing II*

Sources

The course material (slides & exercises) are in part derived from, or inspired by, third-party material and adapted for this course. I like to give credit to the following persons and institutions:

- ©Software Carpentry [1]
- Martin Dahlö (NBIS)
- Vince Buffalo, Bioinformatics Data Skills
- Lauren Mills, Common File Formats, *Current Protocols in Bioinformatics* A.1B.1-A.1B.18, March 2014

[1] `http://software-carpentry.org`, license: CC BY 4.0

Tasks of a computer

- run programs
- store data
- communicate with each other
- interact with us

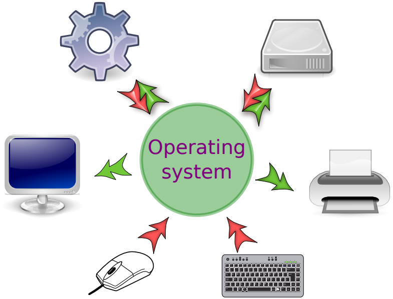
Tasks of a computer

- run programs
- store data
- communicate with each other
- interact with us

©procrastinatingnovelist.blogspot.se

Operating system (OS)

- is a collection of software that manages the hardware, the communication between programs, as well as the network communication with other systems
  → the computer program that starts when you turn on the computer



Components of an operating system

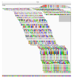
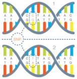
- kernel → has complete control over the computer's resources (i.e. CPU, memory, input/output devices)
- utilities → small programs and system libraries helping to manage the system and hardware
- user interfaces → enable users to interact with computers
  - keybord
  - graphical user interface (GUI)
  - command-line interface (CLI)

What is Unix?

- is a family of operating systems
- born in the beginning of the 70s at AT&T Bell Labs
- distributed under license to universities, US goverment & companies
- different types of UNIX:
    - Solaris (proprietary)
    - MacOS X (proprietary)
    - Linux (free)

Linux

- is a free (Unix-like) operating system based on the Linux kernel
- 1991, Linus Torvalds (University of Helsinki)
- source codes are free for all to copy, to study, to change and to share with others
- used for many computing platforms, like PC, supercomputers, smartphones (Android is based on the Linux kernel)

Unix/Linux benefits

- *stable* → famous for running months, or even years without crashing, freezing, or having to be rebooted
- *secure* → supports effective means to secure a system and prevent unauthorized access
- *fast* → the OS is very efficient at managing resources such as memory, CPU power, and disk space
- *networking* → the network is central for remote access and computation
- *multi-user* → different users can connect at the same server and work at the same time
- *multi-tasking* → several programs can run on the same server at the same time

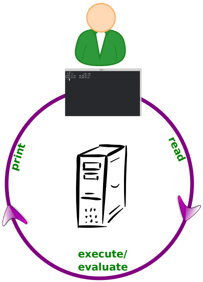
Command-line interface

- in the 70s GUIs were not available → computers were controlled with text commands through a CLI
- allows the direct interaction with the computer by typing in commands (instructions) into a screen

  → a conversation between user and computer
- text-based, accurate and efficient
- needs less hardware resources
- *xterm, console, terminal* → terminal emulators, programs that put an all-text mode window up and let the user interact with the shell

The shell

- command-line interpretor (CLI) → a program, which mediate between the user and the OS
- read-evaluate-print loop:



- lots of different command line shells exist, with different features and facilities

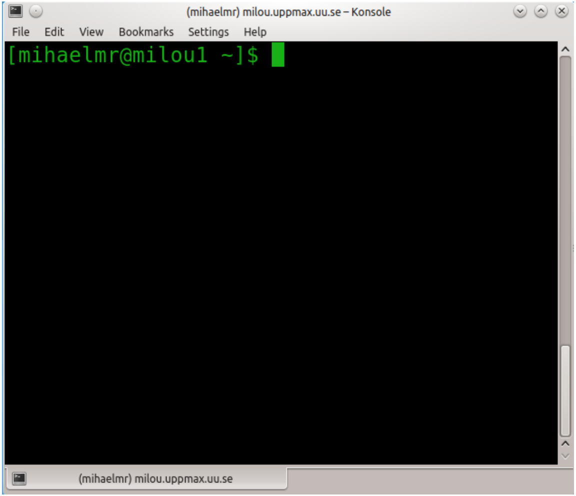
Common Shells

- Windows
  - DOS/Command Prompt
  - PowerShell
- Bourne shell (**sh**) – the original UNIX shell
- Bourne-Again shell (**bash**) – tends to be the LINUX and MAC OS X standard shell
- C shell (**csh**) – syntax related to the C programming language

Common Shells

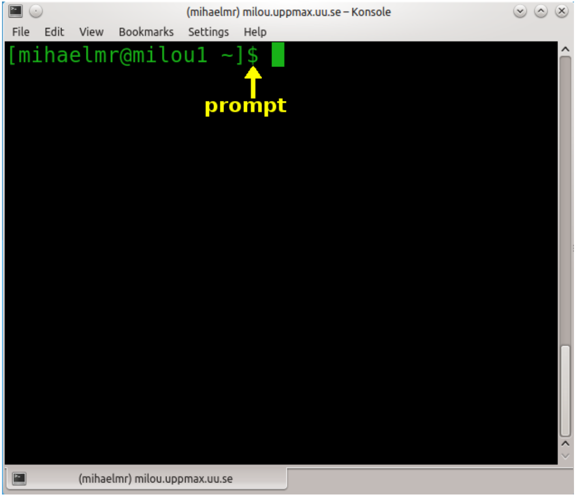
- Windows
  - DOS/Command Prompt
  - PowerShell
- Bourne shell (**sh**) – the original UNIX shell
- Bourne-Again shell (**bash**) – tends to be the LINUX and MAC OS X standard shell
- C shell (**csh**) – syntax related to the C programming language
- Case sensitive

The shell window


(mihaelmr) milou.uppmax.uu.se – Konsole
File
Edit
View
Bookmarks
Settings
Help
(mihaelmr) milou.uppmax.uu.se

```
[mihaelmr@milou1 ~]$
```

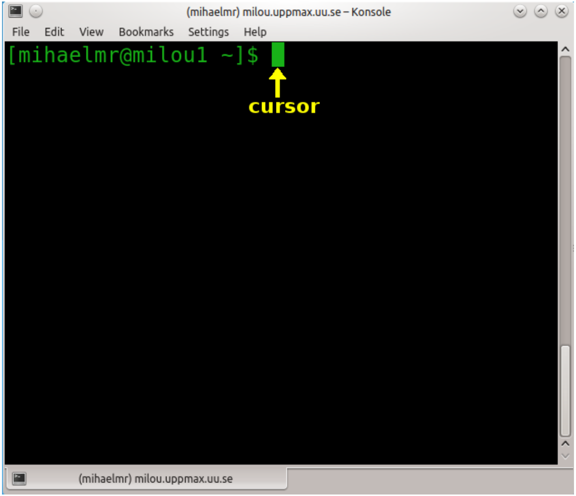
The shell window



```
[mihaelmr@milou1 ~]$ 
```

(mihaelmr) milou.uppmax.uu.se – Konsole
File Edit View Bookmarks Settings Help
prompt
(mihaelmr) milou.uppmax.uu.se

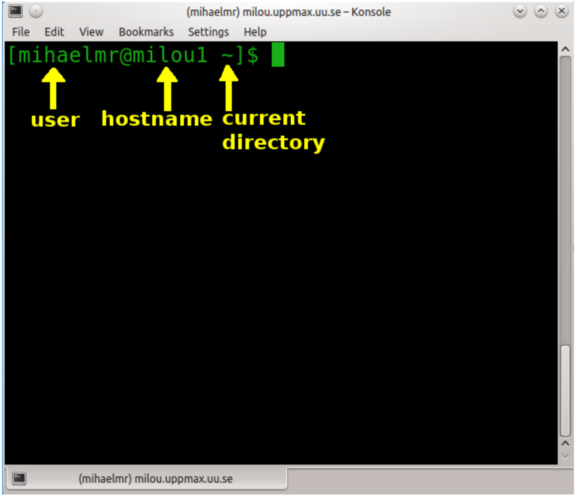
The shell window



(mihaelmr) milou.uppmax.uu.se – Konsole
File Edit View Bookmarks Settings Help
cursor
(mihaelmr) milou.uppmax.uu.se

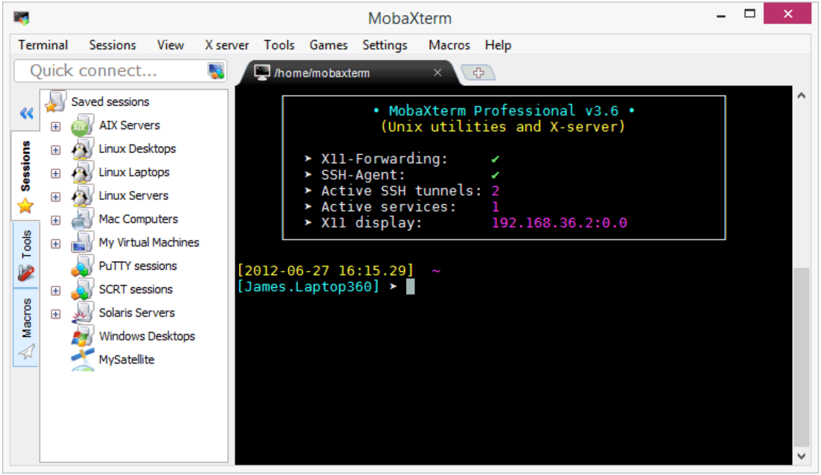
```
[mihaelmr@milou1 ~]$ █
```

The shell window


(mihaelmr) milou.uppmax.uu.se – Konsole
File Edit View Bookmarks Settings Help
user
hostname
current directory
(mihaelmr) milou.uppmax.uu.se

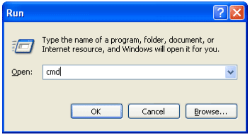
```
[mihaelmr@milou1 ~]$
```

MobaXterm shell window



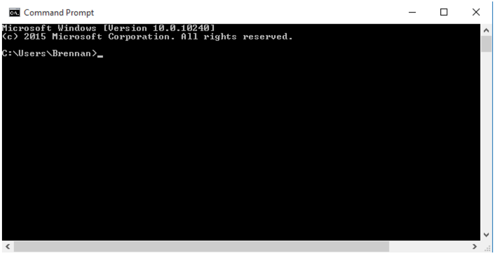
```
        • MobaXterm Professional v3.6 •
         (Unix utilities and X-server)

   ➤ X11-Forwarding:      ✓
   ➤ SSH-Agent:           ✓
   ➤ Active SSH tunnels: 2
   ➤ Active services:     1
   ➤ X11 display:         192.168.36.2:0.0

[2012-06-27 16:15.29]  ~
[James.Laptop360] ➤
```

Windows command prompt



Type the name of a program, folder, document, or Internet resource, and Windows will open it for you.

Open: cmd



```
Microsoft Windows [Version 10.0.10240]
(c) 2015 Microsoft Corporation. All rights reserved.

C:\Users\Brennan>_
```

→ 'cmd' is the command line interface for Windows, offering a DOS-like environment.

Why use the command line/shell?

- programming language features → conditional expressions, loops, variables
- shell commands can be combined into **scripts** → automate processes
- easiest way to interact with remote machines and supercomputers (e.x. Uppmax)
- provides a history of executed commands
- interactive use
  - manage files and directories
  - find, run, and control commands/programs/processes

What is a command?

- is an instruction typed in at the command line and processed by the shell
- commands are either
  - external → executable programs written in a programming language (e.x. Fastqc)
  - built-in → are part of the shell (e.x. cd)
- are roughly the same in any Linux distribution
- syntax: *command [options] [arguments]*

Command line structure

- **command name** specifies the basic operation required

  e.g. dir,cd,ls
- **command arguments** specifies what to work on (e.x. names of files)

  e.g. lpr myfile.txt (→ print myfile.txt)
- **command options** specifies how to work is to be done
  - e.g. ls -l -a (→ list file details, including hidden files)

Command syntax conventions

Syntax	Description	Example
brackets [ ]	the information enclosed is optional	[-h host_location]
braces {}	a set of mutually exclusive options	{-l IP | -n hostname }
vertical bar |	separates mutually exclusive options	{-l IP | -n hostname}
<>	placeholder, replace with appropriate value	<username>

Good to know

- commands are **case sensitive** (image.jpg $\neq$ Image.JPG)
- **autocompletion** → write the first letters and type the ⇆
- ↑ , ↓ – navigate in the commands historical
- ← , → – move the cursor back or forth along the current command line
- Ctrl a, Ctrl e – move the cursor to beginning/end of command line
- Ctrl ← / → – move from one word to another

First simple commands

- whoami – print the user's login name
- finger *user_name* – system info about a user
- hostname – print the name of the local host
- date – print the operating system date and time
- pwd – display the name of the current directory
- who – determine the users logged on the machine

Practical example

- *'goslim.txt'* → list of *Arabidopsis* identifier associated with GO-terms

```
AT1G01010 GO:0005634 GO:0005634 GO:0007275
AT1G01020 GO:0032541 GO:0004525 GO:0006665 GO:0097036
AT1G01030 GO:0003700 GO:0005634
AT1G01040 GO:0004525 GO:0035279
...
```

- tasks:
  - How many genes are in the file?
  - How many different GO-terms are in the file?
  - How many of these are shared by different genes?
  - Find the three most frequent GO-terms!

Unix/Linux solution I

- How many genes are in the file?

```
$ wc -l goslim.txt
30485 goslim.txt

$ grep -c "^" goslim.txt
30485
```

Unix/Linux solution I

- How many genes are in the file?

```
$ wc -l goslim.txt
30485 goslim.txt

$ grep -c "^" goslim.txt
30485
```

- How many different GO-terms are in the file?

```
$ cat goslim.txt|tr " " "\n"|grep "^GO:"|sort|uniq|wc -l
6315
```

Unix/Linux solution II

- How many of these are shared/not shared by different genes?

```
$ cat goslim.txt|sed 's/ /\n/g'|grep "^GO:"|sort|uniq -d|wc -l
5268

$ cat goslim.txt|sed 's/ /\n/g'|grep "^GO:"|sort|uniq -u|wc -l
1047
```

Unix/Linux solution II

- How many of these are shared/not shared by different genes?

```
$ cat goslim.txt|sed 's/ /\n/g'|grep "ˆGO:"|sort|uniq —d|wc —l
5268

$ cat goslim.txt|sed 's/ /\n/g'|grep "ˆGO:"|sort|uniq —u|wc —l
1047
```

- Find the three most frequent GO-terms!

```
$ cat goslim.txt|tr " " "\n"|grep "ˆGO:"|sort|uniq —c|sort —nr|head —3
15527 GO:0005634
 9983 GO:0003674
 9318 GO:0008150
```

... in detail

```
$ cat goslim.txt|tr " " "\n"|grep ^GO:|sort|uniq -c|sort -nr|head -3
```

cat → reads data from the file, and outputs the content

| → pass the output from one program to the input of another one

tr → transform one pattern into another one

grep → search for patterns and display the result lines

sort → sort the data

uniq → remove duplicates

head → display the first 3 lines

Find help

- problems with the UNIX commands:
  - use the command man
  - use a search engine (e.g. Google)
  - ask someone with more experience
- bioinformatics problems:
  - seqanswers.com
  - biostars.org

The *man* command

- usage: man <command>

```
$ man whoami
NAME
    whoami - print effective userid
SYNOPSIS
    whoami [OPTION] ...
DESCRIPTION
    Print the user name associated with the current
        effective user ID
    --help display this help and exit
    --version output version information and exit
AUTHOR
    Written by Richard Mlynarik
```

The *help* command

- displays brief summaries of shell builtin commands
- usage: help <options> <command>
- -d – output short description for each topic

```
$ help -d cd
cd - Change the shell working directory
```

- -s – output only a short usage synopsis

```
$ help -s cd
cd: cd [-L|[-P [-e]]] [dir]
```

Summary

- **UNIX** is a family of operating systems (OS)
- **UNIX/Linux** OS can manage multiple users, multiple tasks, and networting
- The **shell** (Command Line Interpretor) is a program that reads commands typed into a console/terminal and executes them
- A **command** is an instruction typed in at the command line and processed by the shell

Bioinformatics file formats

- flat files → simple ASCII text files that contain data in a certain format
  - data files → complete sets of sequence and annotation data
  - alignment files → alignments created by pair-wise or multiple genome alignment programs
  - annotation files → sequence coordinates and diverse annotations on those coordinates

Data formats	Alignment formats	Annotation formats
FASTA	**SAM/BAM**	GFF3/GTF
FASTQ	MAF	**VCF**
GenBank	Stockholm	BED
EMBL		WIG

FASTA format

- one of the simplest and most flexible file formats
- used to store sequences and their accession/description (e.g. CDS, proteins, reference genomes, transcript sequences)
- each sequence entry
  - description line, begins with a '>' sign
  - sequence line

```
>sp|P42645|14335_ARATH 14-3-3-like protein GF14 OS=At
MSSDSSREENVYLAKLAEQAERYEEMVEFMEKVAKTVETEELTVEERNLLSVAYKNVIGA
RRASWRIISSIEQKEDSRGNSDHVSIIKDYRGKIETELSKICDGILNLLEAHLIPAASLA
ESKVFYLKMKGDYHRYLAEFKTGAERKEAAESTLVAYKSAQDIALADLAPTHPIRLGLAL
NFSVFYYEILNSSDRACSLAKQAFDEAISELDTLGEESYKDSTLIMQLLRDNLTLWTSDL
NDEAGDDIKEAPKEVQKVDEQAQPPPSQ
```

Fasta format particularities

- no standard specification for the identifier format in the description line
- sequence written in IUPAC single-letter code
- an '*' in a protein sequence indicates a translation stop
- lowercase letters indicate regions of low complexity or repeats

Database	Identifier syntax	Example
NCBI	gi|accession:locus	>gi|240256493:c5245820-5243745
NCBI RefSeq	gi|accession|ref|accession	>gi|320461685|ref|NM_001202423.1|
Swiss-Prot	sp|accession|name	>sp|P42645|14335_ARATH
TrEMBL	tr|accession|name	>tr|A8MUZ1|A8MUZ1_HUMAN
ENA	ENA|accession|name	>ENA|AAH24005|AAH24005.1
ENSEMBL	accession	>TRMT1L-003

Common naming convention

- don't separate '>' and identifier by a space (~~'> ID'~~)
- split the description line into 2 parts: the identifier and the comment separated by a white space

```
>gene_00234544 length=231;type=dna
GAGAACTGATTCTGTTACGCGCGGAGAACTGATTCTGTTACGCGCGTTCT
```

- use unique ID

File extension	Meaning
.fasta/.fa/.fas	generic fasta
.fna	fasta nucleic acid
.ffn	fasta nucleotide of gene regions
.faa	fasta amino acid
.frn	fasta non-coding RNA

FASTQ format

- used to store high-throughput sequencing data
- extends the FASTA format by including a numeric quality score to each base
- consists of four parts:
  - description line, beginning with '@'
  - raw sequence in IUPAC
  - a '+' placeholder line
  - quality score

```
@DJB775P1:248:DOMDGACXX:7:1202:12362:49613
GATTTGGGGTTCAAAGCAGTATCGATCAAATAGTAAATCCATTTGTTCAACTCACAGTTT
+
!''*((((***+))%%%++)(%%%%).1***-+*''))**55CCF>>>>>>CCCCCCC65
```

Quality scores

- reflect the probability that a base call was incorrect
- calculated as Phred quality scores: $PhredQ = -10log(p)$
- higher Q scores indicate a smaller probability of error

Quality score	Error probability	Accuracy
10	0.1	90%
20	0.01	99%
30	0.001	99.9%

SAM/BAM

- stores the result of mapping raw reads onto reference genomes
- SAM – **S**equence **A**lignment/**M**ap
- BAM – **B**inary **A**lignment/**M**ap
- contain 2 sections:
  - header lines – starts with '@' followed by a two-letter record type code
  - alignment data – stored in 12 tab-delimited columns

```
@HD VN:1.5 SO:coordinate
@SQ SN:ref LN:15072434
r001 83  ref 2012257  40 50M = 2011868  -439 CAAAAAATTTTGAAAAAAAAAATT [...]
r002 163 ref 2011868  60 50M = 2012257   439 GTGGAGACAGCGCCAAAACACCAC [...]
r003 83  ref 13331006 60 50M = 13330604 -452 CTAGCGCGCGCACCGCCCGTGTTG [...]
r004 163 ref 13330604 60 50M = 13331006  452 TGGAAATAGTTCAGTTTAAAGCAT [...]
```

The SAM header

- contain vital metadata about the reference sequences, read and sample information, processing steps and comments
- line starts with '@' followed by a 2-letter code and tab-delimited KEY:VALUE pairs
- @SQ – ref. sequence information: **@SQ SN:rye LN:4000**
- @RG – read group/sample information: **@RG ID:VB00023_L001 SM:celegans-01 PL:Illumina**
- @PG – information about the programs used to create the SAM/BAM files: **@PG ID:bwa**

The SAM alignment section

Column	Field	Description
1	QNAME	query name
2	FLAG	bitwise flag
3	RNAME	reference name
4	POS	position on the ref. sequence
5	MAPQ	mapping quality
6	CIGAR	format for describing the alignment
7	RNEXT	reference name of the next read
8	PNEXT	position of the next read
9	TLEN	template length
10	SEQ	aligned sequence
11	QUAL	quality score of alignment
12	OPTIONAL	additional information

→ http://samtools.sourceforge.net/SAMv1.pdf

CIGAR STRINGS

- encode which bases of an alignment are matches/mismatches (M), insertions (I), deletions (D), soft (S) or hard (H) clipped
- soft clipping → only part of the query sequence is aligned to the reference
- hard clipping → similar to soft clipping, but regions without match are not present in the sequence stored in the SAM file

```
76M
  76 base pairs matched without INDEL's to the reference
43S6M1I26M
  43S -- 43 bases soft clipped
  6M  --  6 matched bases (mismatch/match)
  1I  --  1 base pair insertion to the reference
  26M -- 26 matched bases
```

The VCF format

- **V**ariant **C**all **F**ormat → standardized text file format for representing SNP, indel, and structural variation calls
- **two main parts**: the header, and the variant call records
- **header** → describes dataset, reference, and defines the annotations used to qualify and quantify the variants
- **data lines** → each line represent a single variant

VCF file header information

```
##fileformat=VCFv4.1
##FILTER=<ID=LowQual,Description="Low quality">
##FORMAT=<ID=AD,Number=.,Type=Integer,Description="Allelic depths for the ref">
##FORMAT=<ID=DP,Number=1,Type=Integer,Description="Total Depth">
##FORMAT=<ID=FA,Number=.,Type=Float,Description="Allele Frequency">
##FORMAT=<ID=GT,Number=1,Type=String,Description="Genotype">
```

- 1st line → VCF specification version
- filter lines → tell what filter have been applied
- FORMAT and INFO lines → define the annotations contained in the FORMAT and INFO columns of the VCF file

Variant call records

```
#CHROM  POS   ID   REF ALT  QUAL  FILTER  INFO  FORMAT  P1  P2
chr1  10043 rs2 T G . REJECT . GT:AD:DP:FA  0:17,2:8:0.105  0/1:4,0:4:0.00
chr1  10079 rs3 T G 47 PASS SOMATIC GT:AD:DP:FA  0:18,5:10:0.217  0/1:4,0:4:0.00
chr1  10157 .   T C 29 PASS SOMATIC GT:AD:DP:FA  0:8,0:6:0.00  0/1:5,2:7:0.286
```

- each site record is structured into columns
- the first 7 columns are mandatory and represent the properties observed at the level of the variant site
- 8th column (INFO) → site-level annotations
- 9th column (FORMAT) → sample-level annotation
- 10th column → sample-name columns
- ’.’ → serve as a placeholder

Sample-level annotation

```
#CHROM  POS   ID   REF ALT  QUAL  FILTER  INFO  FORMAT  P1  P2
chr1  10157 .   T C 29 PASS . GT:AD:DP:FA  0:8,0:6:0.00  0/1:5,2:7:0.286
```

- GT → genotype (0= REF allele, 1= 1st ALT allele, 2= 2nd ALT allele)

  0/0 – sample is homozygous reference, 0/1 sample is heterozygous
- AD → allele depth, number of reads that support each of the reported allele
- DP → filtered depth, number of filtered reads that support each of the reported allele

http://gatkforums.broadinstitute.org/wdl/discussion/1268/what-is-a-vcf-and-how-should-i-interpret-it

Summary

- **FASTA** - text-based format for representing either DNA- or AA-sequences together with their names
- **FASTQ** - text-based format for storing both a biological sequence and its corresponding quality scores
- **SAM/BAM** - is a generic alignment format for storing read alignments against reference
- **VCF** - text file used for storing gene sequence variations