



Bridging the gaps in bioinformatics/What's next?

Outlook – how to keep learning

Congratulations!



Learning bioinformatics is like..

Phase 2:
Hitting the plateau



Learning bioinformatics is like..

Phase 2:
Hitting the plateau



- Time needed for phase 1 is highly variable
 - We learn by building on previous knowledge
- Taking a long time in phase 1 doesn't mean you will also be slow in phase 2

More activities in this programme

- 3-day face-to-face workshop in SQL
- Exchange visits
- Virtual seminars

Think about what your needs are, and let us know.

- Programme is still under development, we may be able to fit your desired topic in

Take more courses



coursera

Explore ▾

What do you want to learn?



(1225 results for python...)

Hello, What Do You Want To Learn?

Learn Python, Jav

Search

<https://www.geeksforgeeks.org>

Learn to Code

With the world's largest web developer site.

Search our tutorials, e.g. HTML



<https://www.w3schools.com>

Getting the most out of this course: practice!



Getting comfortable on the command-line takes time

- If you don't have access to linux yet, try to get it
- Use the terminal for navigation whenever possible

Try running the software from the course on your own data

Open-source software is wonderful – most of the time..

- Anybody can make and publish software
 - There are some excellent tools out there
 - But also some very crappy ones..
 - Like bad publications...

👍 Be critical. Test it!

👍 Talk to your colleagues (reputation matters)

Open-source software is wonderful – most of the time..

- Computers have different operating systems, with different versions. That's a significant challenge for software developers.
 - You might run into installation issues.
 - If you have followed the instructions, and it doesn't work, make a bug report
 - If there are no instructions – be **very** critical

If you get stuck (and you will)

There are **loads** of online forums!

And its okey to ask. But, before you do:

- Check if there is already an answer on the forum
- Make a summary title, and introduce your problem
- Try to make your problem reproducible (but don't post sensitive data!)



Good coding practice if for back home too

- Make github repositories of your own
 - Its for free. You can keep it private if you are shy
- Keep your files organized. Review your file-tree.
- Meticulous note-taking is not just for the lab
 - Get in the habit of making a README file in all your working directories
 - Save commands and observations
 - If you do not have a record of what you did, no-one will be able to help you
- Check your output at every step
- Write lots of comment lines in your code! And don't forget usage instructions.