

KSBi-BIML 2024

Bioinformatics & Machine Learning(BIML)
Workshop for Life and Medical Scientists

생명정보학 & 머신러닝 워크샵 (온라인)



Beginning Version Control for Bioinformatics

장혜식 _ 서울대학교



KSBI
KOREAN SOCIETY FOR
BIOINFORMATICS

| 한국생명정보학회



본 강의 자료는 한국생명정보학회가 주관하는 BIML 2024 워크샵 온라인 수업을 목적으로 제작된 것으로 해당 목적 이외의 다른 용도로 사용할 수 없음을 분명하게 알립니다.

이를 다른 사람과 공유하거나 복제, 배포, 전송할 수 없으며 만약 이러한 사항을 위반할 경우 발생하는 **모든 법적 책임은 전적으로 불법 행위자 본인에게 있음을 경고**합니다.

KSBI-BIML 2024

Bioinformatics & Machine Learning(BIML) Workshop for Life and Medical Scientists

안녕하십니까?

한국생명정보학회가 개최하는 동계 교육 워크숍인 BIML-2024에 여러분을 초대합니다. 생명정보학 분야의 연구자들에게 최신 동향의 데이터 분석기술을 이론과 실습을 겸비해 전달하고자 도입한 전문 교육 프로그램인 BIML 워크숍은 2015년에 시작하여 올해로 벌써 10년 차를 맞이하게 되었습니다. BIML 워크숍은 국내 생명정보학 분야의 최초이자 최고 수준의 교육프로그램으로 크게 인공지능과 생명정보분석 두 개의 분야로 구성되어 있습니다. 올해 인공지능 분야에서는 최근 생명정보 분석에서도 응용이 확대되고 있는 다양한 인공지능 기반 자료모델링 기법들에 대한 현장 강의를 진행될 예정이며, 관련하여 심층학습을 이용한 단백질구조예측, 유전체분석, 신약개발에 대한 이론과 실습 강의를 함께 제공될 예정입니다. 또한 단일세포오믹스, 공간오믹스, 메타오믹스, 그리고 롱리드염기서열 자료 분석에 대한 현장 강의는 많은 연구자의 연구 수월성 확보에 큰 도움을 줄 것으로 기대하고 있습니다.

올해 BIML의 가장 큰 변화는 최근 연구 수요가 급증하고 있는 의료정보자료 분석에 대한 현장 강의를 추가하였다는 것입니다. 특히 의료정보자료 분석을 많이 수행하시는 의과학자 및 의료정보 연구자들께서 본 강좌를 통해 많은 도움을 받으실 수 있기를 기대하고 있습니다. 또한 다양한 생명정보학 분야에 대한 온라인 강좌 프로그램도 점차 증가하고 있는 생명정보 분석기술의 다양화에 발맞추기 위해 작년과 비교해 5강좌 이상을 신규로 추가했습니다. 올해는 무료 강좌 5개를 포함하여 35개 이상의 온라인 강좌가 개설되어 제공되며, 연구 주제에 따른 연관된 강좌 추천 및 강연료 할인 프로그램도 제공되며, 온라인을 통한 Q&A 세션도 마련될 예정입니다. BIML-2024는 국내 주요 연구 중심 대학의 전임 교원이자 각 분야 최고 전문가들의 강의로 구성되었기에 해당 분야의 기초부터 최신 연구 동향까지 포함하는 수준 높은 내용의 강의를 될 것이라 확신합니다.

BIML-2024을 준비하기까지 너무나 많은 수고를 해주신 운영위원회의 정성원, 우현구, 백대현, 김태민, 김준일, 김상우, 장혜식, 박종은 교수님과 KOBIC 이병욱 박사님께 커다란 감사를 드립니다. 마지막으로 부족한 시간에도 불구하고 강의 부탁을 흔쾌히 허락하시고 훌륭한 현장 강의와 온라인 강의를 준비하시는데 노고를 아끼지 않으신 모든 강사분들께 깊은 감사를 드립니다.

2024년 2월

한국생명정보학회장 이 인 석

Beginning Version Control for Bioinformatics

앗! 지금 보고 있는 RNA-Seq 분석에서 지난 주까지 분명히 기대했던 GO term이 나왔는데 갑자기 나오지 않는다. 마감이 코앞이라 정신없이 작업했더니 뭘 고쳤는지 기억도 잘 나지 않는다. 당장 내일이 프로젝트 팀 회의인데 뭐라고 말해야 하지... 좌절이다.

코드는 텍스트 파일 여러 개의 뭉치이지만 현실에선 하나의 축이 더해져야 한다. 바로 시간축이다. 1주일만 지나도 내 코드가 남의 코드 같은 느낌이 든다. 작업하면서 바뀌는 부분들을 제대로 기록하고 관리하면 오래된 코드에서도 왜 어떻게 그런 식으로 바뀌었는지 언제든 확인하고 더 효율적으로 일할 수 있다. 특히 여러 사람이 공동으로 작업할 때는 이슈를 관리하고 여러 명의 변경 사항을 통합하고 같이 만들어 낸 버그를 추적하는 데 버전 컨트롤(Version Control)이 꼭 필요하다.

이 강의에서는 버전 컨트롤러로 가장 널리 사용되고 있는 git의 기본 사용법을 익히고 버전 컨트롤의 주요 개념들에 대해서 소개한다. 그리고, git 저장소를 중심으로 이슈 트래킹, 코드 교환, 배포, 연속 통합(Continuous Integration) 등 현대적인 개발 환경의 핵심 플랫폼을 제공하는 GitHub 서비스의 기본적인 사용법과 개념을 같이 익혀서 좀 더 재현 가능하고 안전한 생물정보학 데이터 분석을 시작하는 동기를 마련하는 것을 목표로 한다.

강의는 다음의 내용을 포함한다:

- Version Control의 기본 개념
- Git 사용법
- GitHub의 주요 개념
- GitHub 사용 실습

* 교육생준비물:

수강 도중에 동영상을 멈추고 실습하는 경우 macOS, Linux 또는 Windows가 설치돼 있고 인터넷에 접속 가능한 컴퓨터 필요.

* 강의 난이도: 초급

* 강의: 장혜식 (서울대학교 생명과학부 조교수)

Curriculum Vitae

Speaker Name: Hyesik Chang, Ph.D.



► Personal Info

Name Hyesik Chang
Title Assistant Professor
Affiliation Seoul National University

► Contact Information

Address 1 Gwanak-ro Gwanak-gu, Seoul, 08826
Email hyeshik@snu.ac.kr
Website <https://qbio.io>

Research Interest

High-throughput sequencing, post-transcriptional regulation, RNA-protein interaction

Educational Experience

1998–2007 B.S.E. in Information and Industrial Engineering, Yonsei University, Korea
2007–2009 M.S.E. in Bio and Brain Engineering, KAIST, Korea
2009–2014 Ph.D. in Biological Sciences, Seoul National University, Korea

Professional Experience

2001–2005 Software Developer, Solution Development Team, LinuxKorea, Inc.
2014–2019 Research Assistant Professor, IBS Center for RNA Research, Seoul National University
2018– Research Fellow, Center for RNA Research, Institute for Basic Science
2019– Assistant Professor, School of Biological Sciences, Seoul National University

Selected Publications (5 maximum)

1. D. Kim, J.-Y. Lee, J.-S. Yang, J. W. Kim, V. N. Kim, and H. Chang. (2020) "The Architecture of SARS-CoV-2 Transcriptome." *Cell*, 181(4):914–921.
2. H. Chang¹, J. Yeo¹, J.-G. Kim, H. Kim, M. Lee, J. Lim, H. H. Kim, J. Ohk, H.-Y. Jeon, H. Lee, H. Jung, K.-W. Kim, and V. N. Kim. (2018) "Terminal uridylyltransferases execute programmed clearance of maternal transcriptome in vertebrate embryos." *Molecular Cell*, 70:72–82.e7.
3. J. Lim¹, M. Ha¹, H. Chang¹, S. C. Kwon, D. K. Simanshu, D. J. Patel, and V. N. Kim. (2014) "Uridylation by TUT4 and TUT7 marks mRNA for degradation." *Cell*, 159(6):1365–1376.
4. H. Chang¹, J. Lim¹, M. Ha, and V. N. Kim. (2014) "TAIL-seq: genome-wide determination of poly(A) tail length and 3' end modifications." *Molecular Cell*, 53(6):1044–1052.
5. J. Cho¹, H. Chang¹, S. C. Kwon, B. Kim, Y. Kim, J. Choe, M. Ha, Y. K. Kim, and V. N. Kim. (2012) "LIN28A is a suppressor of ER-associated translation in embryonic stem cells." *Cell*, 151(4):765–777.

¹Co-first authors

KSBi-BIML 2022

Beginning Version Control for Bioinformatics

장혜식

서울대학교 생명과학부

버전 컨트롤 — 왜 하나요?

1. 잘 돌아가는 프로그램을 만들었다.
2. 아주 조금 뭘 하나 고쳤다.
3. 프로그램이 갑자기 오류가 어마어마하게 난다.
4. 원래대로 돌려줬다.
5. 그래도 계속 오류가 난다.
6. 빨리 결과가 필요하다고 옆에서 채근한다.

버전 컨트롤 — 왜 하나요?

1. 어제 까진 프로그램이 아주 잘 돌아왔다.
2. 어젯밤 갑자기 영감을 받아서 어마어마한 기능을 잔뜩 만들어서 넣었다.
 - 아직 완성되지는 않아서 정상동작 하지는 않는다.
3. 갑자기 오늘 새 데이터가 나왔는데 급하게 처리해서 결과를 보내야 한다.
4. 임시로 새 기능을 막아두고 돌리려고 하는데, 정확히 어디를 고쳤는지 기억이 나지 않는다.
5. 빨리 결과를 보내달라고 공동작업자가 채근한다.

3

해결책

- | | | |
|---------------------------------------|---------------------------------|--|
| matam_hw_3 - Final 2 | matam_hw_3 - Final 3 - no leaks | matam_hw_3 - Final 3.3- enhanced prin... |
| matam_hw_3 - Final 3 | matam_hw_3 - Final | matam_hw_3 - v4 -Fixed typos |
| matam_hw_3 - v4 -Ready for submission | matam_hw_3 - v4 | matam_hw_3 |
| matam_hw_3 - submit | matam_hw_3 - submit - Copy | |

4

버전 컨트롤 — 왜 하나요?

1. 여러 명이 같이 작업하는 프로그램이 아주 잘 돌고 있다.
2. 나는 기능 A를 추가했다. - 아주 잘 돈다.
3. 재는 기능 B를 추가했다. - 아주 잘 돈다.
4. 둘을 합쳤다. - 힘들게 합쳤지만 하나도 안 돈다.
5. 어디를 고쳐서 문제가 생겼는지 찾는다.
6. 하지만 고친 곳이 어디인지도 잘 모르겠다.
7. 빨리 결과가 필요하다고 다른 사람이 채근한다.

5

RESEARCH | REPORT

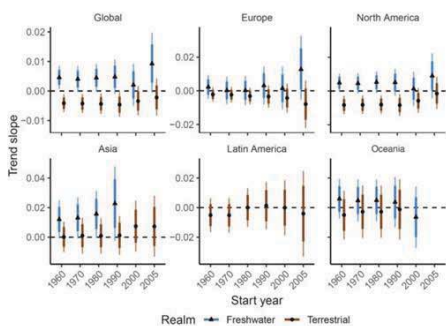


Fig. 3. Trend estimates (± 80 and 95% credible intervals) for progressively shorter time periods since 1960. Each time slice included data until the last sampling date but excluded any sites spanning <9 years within the time slice. Only estimates with at least four datasets or 20 sites are shown. The continents are ordered by data availability. Annotation is as in Fig. 2.

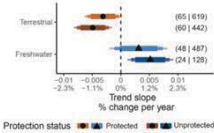


Fig. 4. Trend estimates (± 80 , 90, and 95% credible intervals) for terrestrial and freshwater insects inside and outside of protected areas. Bracketed numbers indicate the number of studies and number of sites underlying each estimate. Annotation is as in Fig. 2.

of recent high-profile case studies [e.g., 3 to 6% loss per year (10, 11)], which were included in our analysis]. Nevertheless, our more synthetic estimate translates to an average loss of 5.81% per decade in terrestrial ecosystems. Such a decline is concerning given the critical role that insects play in food webs and ecosystem services and may contribute to other changes such as the declines observed for some insectivorous bird populations (32–36). At the same time, we found an average increase in freshwater insect abundances that might, at least partially, reflect improvements in water quality. This, in combination with our finding that trends were weaker in protected areas, suggests that appropriate habitat protection and restoration may be effective strategies for mitigating changes in insect assemblages.

REFERENCES AND NOTES

1. A. D. Chapman, *Numbers of Living Species in Australia and the World* (Newcomb, ed. 2, 2009).
2. N. E. Stork, *Annu. Rev. Entomol.* **63**, 31–45 (2018).
3. C. W. Sabrosky, *Syst. Zool.* **2**, 21–36 (1953).
4. E. D. Wilson, *Conserv. Biol.* **1**, 344–346 (1987).
5. J. A. Thomas et al., *Science* **303**, 1879–1888 (2004).
6. M. L. Forister, J. P. Jahnke, K. L. Cooper, J. S. Wilson, A. M. Shapiro, *Ecology* **92**, 2222–2235 (2011).
7. A. Valtonen et al., *J. Anim. Ecol.* **86**, 730–738 (2017).
8. R. Diaz et al., *Science* **345**, 403–405 (2014).
9. S. Schuch, K. Wesche, M. Schäfer, *Biol. Conserv.* **149**, 75–83 (2012).
10. C. A. Hallmann et al., *PLoS ONE* **12**, e0185809 (2017).
11. B. C. Lester, A. Garcia, *Proc. Natl. Acad. Sci. U.S.A.* **115**, E10397–E10406 (2018).
12. G. Vogel, *Science* **356**, 576–579 (2017).
13. B. Jarvis, *The insect apocalypse is here: What does it mean for the rest of life on Earth?* *The New York Times Magazine*, 27 November 2018, pp. 41–48.
14. See the supplementary materials.
15. B. Lehner, *P. Natl. Acad. Sci. U.S.A.* **101**, 122 (2004).
16. B. W. Hillen, T. Pawelczyk, *Science* **301**, 985–988 (2018).

17. A. J. van Strien et al., *Biol. Conserv.* **200**, 44–50 (2016).
18. I. Rochlin, A. Faras, D. V. Ninivaggio, C. M. Barker, A. M. Klobutsky, *Nat. Commun.* **7**, 13604 (2016).
19. C. L. Outhwaite, R. D. Gregory, R. E. Chandler, B. Collen, N. J. B. Isaac, *Nat. Ecol. Evol.* **4**, 284–292 (2020).
20. D. A. Kiese, J. S. Shapiro, *J. Econ. Entomol.* **134**, 359–366 (2019).
21. H. F. V. Bratton et al., *Environ. Sci. Technol.* **53**, 1834–1843 (2019).
22. F. Bourassa, B. Groszeth, *Sci. Total Environ.* **409**, 4899–4916 (2011).
23. P. Bigos, M. Tobieszewski, J. Narnesnik, *Mar. Pollut. Bull.* **78**, 26–42 (2014).
24. V. Lencorn, *Sci. Total Environ.* **622–623**, 563–575 (2018).
25. Y. Cai, Y. Lu, Z. Gong, *J. Freshwat. Ecol.* **30**, 157–168 (2015).
26. K. Shukri et al., *Ecology* **85**, 939–954 (2004).
27. R. D. Roberts, A. V. Zhulodov, D. F. Pavlov, *Aquat. Sci.* **75**, 27–38 (2013).
28. L. A. Henry, V. Douchovnikoff, *Annu. Rev. Environ. Resour.* **33**, 437–460 (2008).
29. J. C. Habel, M. J. Samways, T. Schmidt, *Biodivers. Conserv.* **28**, 1343–1360 (2019).
30. K. Peris, *Ecology of Urban*.
31. UNP/WHO, *UNEP*.
32. C. A. Hallmann, R. P. B. H. de Knop, E. Jongejans, K. V. Rosenberg et al., *Sci. Data* **7**, 1–11 (2020).
33. D. E. Bower, H. Hildroy, K. Böhring-Giese, *Conserv. Biol.* **35**, 1–11 (2021).
34. R. van Klink, D. E. Bower, *Conserv. Biol.* **35**, 1–11 (2021).
35. R. van Klink, D. E. Bower, *Conserv. Biol.* **35**, 1–11 (2021).
36. R. van Klink et al., *A global insect assemblage*, *KNB* (2020). <https://doi.org/10.26434/chemrxiv-2020-08-01>

the manuscript, and all authors substantially edited the text.

Competing interests: The authors declare no competing interests.
Data and materials availability: The data frames used for the analyses are available as data S1 and S2 in the supplementary materials, excluding datasets with access licenses that precluded distribution of a derived product. Links to these datasets, and all other publicly available datasets, are provided in table S1. All code for this analysis is available on GitHub (<https://github.com/roelvanklink/Final-insect-abundance-changes>) and is archived on Zenodo (35). The underlying database, including extended metadata, is available on KNB (36).

SUPPLEMENTARY MATERIALS

ACKNOWLEDGMENTS
 We thank R. Vermeulen (M. Driessen, J. Owen, F. G. Butterfly Monitoring Israel, Institution for making data the Greenland Ecosystems provided by the Department Denmark, The ECN data was Environment Research Cou, S. Sloves, P. Kell, A. T. Cui help with data extraction, and commenting on earlier Funding: R.v.K., J.M.C., D.E. German Centre for Integrative Halle-Jena-Leipzig and its German Research Foundation by the Russian Foundation Some of the data analyzed to the IGER Network (IS24 0080529, 0217714, DEB-04 DEB-1226696, DEB-0832620, DEB-0939398, DEB-183001 DEB-0620552, DEB-1234262, OCE-0925213, OCE-79209 OCE-1237140, and OCE-1832378). **Author contributions:** R.v.K. and J.M.C. conceived the study. R.v.K. and K.B.G. performed the literature search. A.D., D.E., and A.S.S. collected the data. R.v.K. and D.E.B. analyzed the data. R.v.K. and J.M.C. wrote the first version of the manuscript, and all authors substantially edited the text.

COMPETING INTERESTS AND MATERIALS AVAILABILITY
 The data frames used for the analyses are available as data S1 and S2 in the supplementary materials, excluding datasets with access licenses that precluded distribution of a derived product. Links to these datasets, and all other publicly available datasets, are provided in table S1. All code for this analysis is available on GitHub (<https://github.com/roelvanklink/Final-insect-abundance-changes>) and is archived on Zenodo (35). The underlying database, including extended metadata, is available on KNB (36).

DATA AVAILABILITY STATEMENT
 The data frames used for the analyses are available as data S1 and S2 in the supplementary materials, excluding datasets with access licenses that precluded distribution of a derived product. Links to these datasets, and all other publicly available datasets, are provided in table S1. All code for this analysis is available on GitHub (<https://github.com/roelvanklink/Final-insect-abundance-changes>) and is archived on Zenodo (35). The underlying database, including extended metadata, is available on KNB (36).

Supplementary Materials
<https://doi.org/10.26434/chemrxiv-2020-08-01>
 Source: <https://www.nature.com/content/368/6480/417/suppl/DC1>
 Materials and Methods
 Figs. S1 to S7
 Tables S1 to S3
 External Data S1 and S2
 References (37–206)
 10 May 2019; accepted 3 March 2020
 10.1101/2020.03.03.331931

Sequence analysis

VisFeature: a stand-alone program for visualizing and analyzing statistical features of biological sequences

Jun Wang¹, Pu-Feng Du^{1,*}, Xin-Yu Xue¹, Guang-Ping Li¹, Yuan-Ke Zhou¹, Wei Zhao¹, Hao Lin² and Wei Chen^{3,4,*}

¹College of Intelligence and Computing, Tianjin University, Education, School of Life Science and Technology, of China, Chengdu 610054, China, ²Innovative Institute of Medicine, Chengdu 611137, China and ³Center for University of Science and Technology, Tangshan 056503

*To whom correspondence should be addressed.
Associate Editor: John Hancock

Received on July 5, 2019; revised on August 12, 2019; editorial decision on August 20, 2019

statistical features of all types of biological sequence, including DNA, RNA and proteins. VisFeature also integrates sequence data retrieval, multiple sequence alignments and statistical feature generation functions.

Availability and implementation: VisFeature is a desktop application that is implemented using JavaScript/Electron and R. The source codes of VisFeature are freely accessible from the GitHub repository (<https://github.com/wangjun1996/VisFeature>). The binary release, which includes an example dataset, can be freely downloaded from the same GitHub repository (<https://github.com/wangjun1996/VisFeature/releases>).

Contact: ndu@tju.edu.cn or chenweimu@gmail.com

Abstract

Summary: Many efforts have been made in developing bioinformatics algorithms to predict functional attributes of genes and proteins from their primary sequences. One challenge in this process is to intuitively analyze and understand the statistical features that have been selected by heuristic or iterative methods. In this paper, we developed VisFeature, which aims to be a helpful software tool that allows the users to intuitively visualize and analyze statistical features of all types of biological sequence, including DNA, RNA and proteins. VisFeature also integrates sequence data retrieval, multiple sequence alignments and statistical feature generation functions.

Availability and implementation: VisFeature is a desktop application that is implemented using JavaScript/Electron and R. The source codes of VisFeature are freely accessible from the GitHub repository (<https://github.com/wangjun1996/VisFeature>). The binary release, which includes an example dataset, can be freely downloaded from the same GitHub repository (<https://github.com/wangjun1996/VisFeature/releases>).

Contact: ndu@tju.edu.cn or chenweimu@gmail.com

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Over the last two decades, the number of known biological sequences has been growing exponentially. It is urgent to understand their functional attributes. Many efficient computational methods for generating statistical features from sequences have been developed. Several web servers and stand-alone programs have been released for practical applications, such as PseAAC (Shen and Chou, 2008), PseAAC-General (Du *et al.*, 2014), PseKNC-General (Chen *et al.*, 2015), Pse-in-One (Lin *et al.*, 2015) and UltraPse (Du *et al.*, 2017). These software tools provide efficient and convenient solutions to generate statistical features for biological sequences. However, a helpful software tool for visualizing the statistical features is still lacking. Although existing programs can display nucleotide sequences using dinucleotide property curves in a genome browser style (Friedel *et al.*, 2009), the abilities to visualize protein sequences and to intuitively compare statistical features are still missing.

To this end, we developed VisFeature, which is an open-source stand-alone program that can visualize and analyze various types of statistical features of all types of biological sequences, including

DNA, RNA and proteins. To the best of our knowledge, this is the first toolkit that is designed especially for this purpose. VisFeature integrates sequence feature visualization and analysis, statistical feature visualization and comparison, online database querying, multiple sequence alignment and color sequence visualization together. All these functions are useful in the explorative stage of developing predictive algorithms for functional attributes.

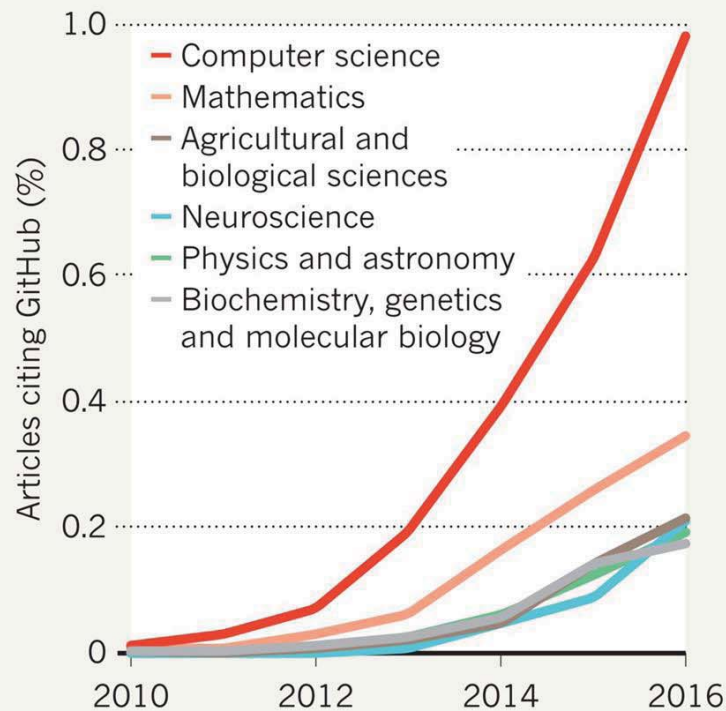
2 Implementations

VisFeature is mainly implemented by using JavaScript, with the Electron framework. The remaining part of VisFeature is implemented by R scripts.

The input of VisFeature is a FASTA format file. This file can be chosen directly from the local computer. With an internet connection, VisFeature is capable of querying the UniProt database or NCBI databases using sequence identifiers or query expressions. The sequences in the query results can be saved as a FASTA file. This is the second way to obtain a FASTA file in VisFeature.

GROWING INFLUENCE OF GITHUB

An increasing proportion of research articles cite GitHub in their references.



git 실습 - 준비

sandbox: ~/vcontrol/safereiz

```
sandbox:~$ cd ~/vcontrol/safereiz/  
sandbox:~/vcontrol/safereiz$ ls  
find-safe-re.py human-miRNA-hairpins.fa mylib.py restriction-enzymes.txt  
sandbox:~/vcontrol/safereiz$
```

새 Repository 초기화 (git init)

sandbox: ~/vcontrol/safereiz

```
sandbox:~$ cd ~/vcontrol/safereiz/  
sandbox:~/vcontrol/safereiz$ ls  
find-safe-re.py human-miRNA-hairpins.fa mylib.py restriction-enzymes.txt  
sandbox:~/vcontrol/safereiz$ git init  
Initialized empty Git repository in /home/guests/lime/vcontrol/safereiz/.git/  
sandbox:~/vcontrol/safereiz$
```

git에 파일 추가 (git add)

sandbox: ~/vcontrol/saferenz

```
sandbox:~/vcontrol/saferenz$ ls
find-safe-re.py human-miRNA-hairpins.fa mylib.py restriction-enzymes.txt
sandbox:~/vcontrol/saferenz$ git add *.fa *.py *.txt
sandbox:~/vcontrol/saferenz$
sandbox:~/vcontrol/saferenz$
```

git에 처음 커밋하기 (git commit)

sandbox: ~/vcontrol/saferenz

```
sandbox:~/vcontrol/saferenz$ ls
find-safe-re.py human-miRNA-hairpins.fa mylib.py restriction-enzymes.txt
sandbox:~/vcontrol/saferenz$ git add *.fa *.py *.txt
sandbox:~/vcontrol/saferenz$
sandbox:~/vcontrol/saferenz$ git commit

*** Please tell me who you are.

Run

  git config --global user.email "you@example.com"
  git config --global user.name "Your Name"

to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got 'lime@sandbox.(none)')
sandbox:~/vcontrol/saferenz$
```

사용자 이름 설정

```
sandbox: ~/vcontrol/saferenz$ ls
find-safe-re.py human-miRNA-hairpins.fa mylib.py restriction-enzymes.txt
sandbox:~/vcontrol/saferenz$ git add *.fa *.py *.txt
sandbox:~/vcontrol/saferenz$
sandbox:~/vcontrol/saferenz$ git commit

*** Please tell me who you are.

Run

git config --global user.email "you@example.com"
git config --global user.name "Your Name"

to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got 'lime@sandbox.(none)')
sandbox:~/vcontrol/saferenz$ git config --global user.email "janedoe@snu.ac.kr"
sandbox:~/vcontrol/saferenz$ git config --global user.name "Jane Doe"
sandbox:~/vcontrol/saferenz$
```

(진짜로) 첫 번째 커밋

```
sandbox: ~/vcontrol/saferenz$ ls
find-safe-re.py human-miRNA-hairpins.fa mylib.py restriction-enzymes.txt
sandbox:~/vcontrol/saferenz$ git add *.fa *.py *.txt
sandbox:~/vcontrol/saferenz$
sandbox:~/vcontrol/saferenz$ git commit

*** Please tell me who you are.

Run

git config --global user.email "you@example.com"
git config --global user.name "Your Name"

to set your account's default identity.
Omit --global to set the identity only in this repository.

fatal: unable to auto-detect email address (got 'lime@sandbox.(none)')
sandbox:~/vcontrol/saferenz$ git config --global user.email "janedoe@snu.ac.kr"
sandbox:~/vcontrol/saferenz$ git config --global user.name "Jane Doe"
sandbox:~/vcontrol/saferenz$ git commit -a
```

커밋 메시지 작성하기

sandbox: ~/vcontrol/safere

GNU nano 2.9.3 /home/guests/lime/vcontrol/safere/.git/COMMIT_EDITMSG Modified

```
Import initial repository.
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch master
#
# Initial commit
#
# Changes to be committed:
#   new file:   find-safe-re.py
#   new file:   human-miRNA-hairpins.fa
#   new file:   mylib.py
#   new file:   restriction-enzymes.txt
#
```

nano (pico) 단축키:
Ctrl-O 저장
Ctrl-X 끝

File Name to Write: /home/guests/lime/vcontrol/safere/.git/COMMIT_EDITMSG

^G Get Help M-D DOS Format M-A Append M-B Backup File
^C Cancel M-M Mac Format M-P Prepend ^T To Files

커밋 메시지-어디에 쓰이는지?

The screenshot shows a GitHub commit history page for a repository. The page is organized by date, with sections for 'Commits on Jun 24, 2017', 'Commits on Mar 30, 2017', 'Commits on Mar 13, 2017', 'Commits on Feb 24, 2017', and 'Commits on Apr 13, 2020'. A central text box with a black border contains the following text:

60글자 내외
뭐 했는지 대충은 알아볼 수 있게
관련 이슈가 있으면 링크를 단다.
자세한 내용은 아래에 쓴다.

The background shows various commit messages such as 'Compatibility fix for a recent version of pandas.', 'Fix a segfault that happens', 'Add support for MiSeq che', 'Add tailseeker dir to paths', 'Remove U3 and 7SL RNAs', 'Merge branch 'master' of github.com:hyeshik/tailseeker', 'Fix invalid behavior on refdb setting (resolves #6).', 'fix for Neon after test.', 'Added documentation entry for added functionality, simde and no_simd.', 'changes for SIMDe.', 'd building non-SIMD version.', 'Merge remote-tracking branch 'refs/remotes/origin/master'', 'r981: asmgene to check duplicate genes', and 'travis: added arm64 test.'

프로그램 수정: 버그를 고쳐보자

```
sandbox: ~/vcontrol/safere | x find-safe-re.py x mylib.py x
1 import mylib
2
3 HAIRPIN_FILE = 'human-miRNA-hairpins.fa'
4 RESENZ_FILE = 'restriction-enzymes.txt'
5
6 hairpin_seqs = mylib.load_fasta(HAIRPIN_FILE)
7 re_patterns = mylib.load_restriction_enzyme_list(RESENZ_FILE)
8
9 for name, pattern in re_patterns.items():
10     print(f"Testing enzyme: {name}")
11
12     matching_hairpins = [
13         seqname
14         for seqname, seq in hairpin_seqs.items()
15         if pattern.search(seq)]
16
17     print(" No of matching hairpins:", len(matching_hairpins))
18     if len(matching_hairpins):
19         print(" First few examples:", ', '.join(matching_hairpins[:5]))
20
21     print()
22
23
```

문제: 생각보다 훨씬 적은 수가 매치됨

```
sandbox: ~/vcontrol/safere | x find-safe-re.py x mylib.py x
sandbox:~/vcontrol/safere | x python find-safe-re.py | head
Testing enzyme: XhoI
  No of matching hairpins: 0

Testing enzyme: EcoRI
  No of matching hairpins: 0

Testing enzyme: BamHI
  No of matching hairpins: 0

Testing enzyme: BanI
sandbox:~/vcontrol/safere | x
```

DNA와 RNA서열 모두에서 찾을 수 있게 수정

```
sandbox: ~/vcontrol/safereiz × find-safe-re.py × mylib.py ×
1 import re
2
3 IUPAC_CODES = {
4     'A': 'A',
5     'C': 'C',
6     'G': 'G',
7     'T': '[UT]',
8     'U': '[UT]',
9     'R': '[AG]',
10    'Y': '[CTU]',
11    'S': '[GC]',
12    'W': '[ATU]',
13    'K': '[GTU]',
14    'M': '[AC]',
15    'B': '[CGTU]',
16    'D': '[AGTU]',
17    'H': '[ACTU]',
18    'V': '[ACG]',
19    'N': '[ACGTU]',
20 }
21
22 def load_fasta(filename):
23     sequences = {}
24
25     current = None
26     for line in open(filename):
27         if line.startswith('>'):
28             current = line[1:].split(None, 1)[0]
29             sequences[current] = ''
```

드디어 되는듯!? ClaI이 정답이구나!

```
sandbox: ~/vcontrol/safereiz × find-safe-re.py × mylib.py ×
sandbox:~/vcontrol/safereiz$ python find-safe-re.py | head -22
Testing enzyme: XhoI
No of matching hairpins: 2
First few examples: hsa-mir-219a-1, hsa-mir-151a

Testing enzyme: EcoRI
No of matching hairpins: 5
First few examples: hsa-mir-183, hsa-mir-146a, hsa-mir-186, hsa-mir-377, hsa-mir-485

Testing enzyme: BamHI
No of matching hairpins: 5
First few examples: hsa-mir-182, hsa-mir-127, hsa-mir-340, hsa-mir-331, hsa-mir-196b

Testing enzyme: BanI
No of matching hairpins: 19
First few examples: hsa-mir-24-1, hsa-mir-33a, hsa-mir-7-2, hsa-mir-212, hsa-mir-222

Testing enzyme: AvaI
No of matching hairpins: 18
First few examples: hsa-let-7e, hsa-mir-26b, hsa-mir-93, hsa-mir-208a, hsa-mir-187

Testing enzyme: ClaI
No of matching hairpins: 0
sandbox:~/vcontrol/safereiz$
```

변경된 파일 확인 (git status)

```
sandbox: ~/vcontrol/saferenl × find-safe-re.py × mylib.py ×
Testing enzyme: BanI
  No of matching hairpins: 19
  First few examples: hsa-mir-24-1, hsa-mir-33a, hsa-mir-7-2, hsa-mir-212, hsa-mir-222

Testing enzyme: AvaI
  No of matching hairpins: 18
  First few examples: hsa-let-7e, hsa-mir-26b, hsa-mir-93, hsa-mir-208a, hsa-mir-187

Testing enzyme: ClaI
  No of matching hairpins: 0
sandbox:~/vcontrol/saferenz$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

       modified:   mylib.py

Untracked files:
  (use "git add <file>..." to include in what will be committed)

       .ipynb_checkpoints/
       __pycache__

no changes added to commit (use "git add" and/or "git commit -a")
sandbox:~/vcontrol/saferenz$
```

변경된 내용 확인 (git diff)

```
sandbox: ~/vcontrol/saferenl × find-safe-re.py × mylib.py ×
  (use "git checkout -- <file>..." to discard changes in working directory)

       modified:   mylib.py

Untracked files:
  (use "git add <file>..." to include in what will be committed)

       .ipynb_checkpoints/
       __pycache__

no changes added to commit (use "git add" and/or "git commit -a")
sandbox:~/vcontrol/saferenz$ git diff
diff --git a/mylib.py b/mylib.py
index 10fc90d..6ab63f6 100644
--- a/mylib.py
+++ b/mylib.py
@@ -4,8 +4,8 @@ IUPAC_CODES = {
     'A': 'A',
     'C': 'C',
     'G': 'G',
-    'T': 'T',
-    'U': 'U',
+    'T': '[UT]',
+    'U': '[UT]',
     'R': '[AG]',
     'Y': '[CTU]',
     'S': '[GC]',
sandbox:~/vcontrol/saferenz$
```


변경된 내용 확정 (git commit)

```
sandbox: ~/vcontrol/saferenl × find-safe-re.py × mylib.py ×
(use "git checkout -- <file>..." to discard changes in working directory)

    modified:   mylib.py

Untracked files:
  (use "git add <file>..." to include in what will be committed)

    .ipynb_checkpoints/
    __pycache__/

no changes added to commit (use "git add" and/or "git commit -a")
sandbox:~/vcontrol/saferenz$ git diff
diff --git a/mylib.py b/mylib.py
index 10fc90d..6ab63f6 100644
--- a/mylib.py
+++ b/mylib.py
@@ -4,8 +4,8 @@ IUPAC_CODES = {
     'A': 'A',
     'C': 'C',
     'G': 'G',
-    'T': 'T',
-    'U': 'U',
+    'T': '[UT]',
+    'U': '[UT]',
     'R': '[AG]',
     'Y': '[CTU]',
     'S': '[GC]',
sandbox:~/vcontrol/saferenz$ git commit -a
```

변경된 내용 확정 (커밋)

```
sandbox: ~/vcontrol/saferenl × find-safe-re.py × mylib.py ×
GNU nano 2.9.3 /home/guests/lime/vcontrol/saferenz/.git/COMMIT_EDITMSG Modified

Fix problem on detection in RNA seqs
# Please enter the commit message for your changes. Lines starting
# with '#' will be ignored, and an empty message aborts the commit.
#
# On branch master
# Changes to be committed:
#   modified:   mylib.py
#
# Untracked files:
#   .ipynb_checkpoints/
#   __pycache__/
#

File Name to Write: /home/guests/lime/vcontrol/saferenz/.git/COMMIT_EDITMSG
^G Get Help      M-D DOS Format   M-A Append      M-B Backup File
^C Cancel        M-M Mac Format   M-P Prepend     ^T To Files
```

변경 로그 확인 (git log)

```
sandbox: ~/vcontrol/saferen | find-safe-re.py | mylib.py
+++ b/mylib.py
@@ -4,8 +4,8 @@ IUPAC_CODES = {
    'A': 'A',
    'C': 'C',
    'G': 'G',
-   'T': 'T',
-   'U': 'U',
+   'T': '[UT]',
+   'U': '[UT]',
    'R': '[AG]',
    'Y': '[CTU]',
    'S': '[GC]',
}
sandbox:~/vcontrol/saferenz$ git commit -a
[master 5b54247] Fix problem on detection in RNA seqs
1 file changed, 2 insertions(+), 2 deletions(-)
sandbox:~/vcontrol/saferenz$ git log
commit 5b542471bbb71a5615729433bcaddad3c36155b7 (HEAD -> master)
Author: Jane Doe <janedoe@snu.ac.kr>
Date: Mon Apr 27 00:27:48 2020 +0900

    Fix problem on detection in RNA seqs

commit 6d3a09d02b95534595976d95def748231d1ff770
Author: Jane Doe <janedoe@snu.ac.kr>
Date: Sun Apr 26 16:12:15 2020 +0900

    Import initial repository.
sandbox:~/vcontrol/saferenz$
```

갑자기 마음이 변했다. 되돌려 보기

```
sandbox: ~/vcontrol/saferen | find-safe-re.py | mylib.py
[master 5b54247] Fix problem on detection in RNA seqs
1 file changed, 2 insertions(+), 2 deletions(-)
sandbox:~/vcontrol/saferenz$ git log
commit 5b542471bbb71a5615729433bcaddad3c36155b7 (HEAD -> master)
Author: Jane Doe <janedoe@snu.ac.kr>
Date: Mon Apr 27 00:27:48 2020 +0900

    Fix problem on detection in RNA seqs

commit 6d3a09d02b95534595976d95def748231d1ff770
Author: Jane Doe <janedoe@snu.ac.kr>
Date: Sun Apr 26 16:12:15 2020 +0900

    Import initial repository.
sandbox:~/vcontrol/saferenz$ git checkout 6d3a
Note: checking out '6d3a'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by performing another checkout.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -b with the checkout command again. Example:

    git checkout -b <new-branch-name>

HEAD is now at 6d3a09d Import initial repository.
sandbox:~/vcontrol/saferenz$
```

정말 옛날 버전으로 돌아갔나?

```
sandbox: ~/vcontrol/safereniz ✕  
  
sandbox:~/vcontrol/saferenz$ python find-safe-re.py | head  
Testing enzyme: XhoI  
  No of matching hairpins: 0  
  
Testing enzyme: EcoRI  
  No of matching hairpins: 0  
  
Testing enzyme: BamHI  
  No of matching hairpins: 0  
  
Testing enzyme: BanI  
sandbox:~/vcontrol/saferenz$ head mylib.py  
import re  
  
IUPAC_CODES = {  
    'A': 'A',  
    'C': 'C',  
    'G': 'G',  
    'T': 'T',  
    'U': 'U',  
    'R': '[AG]',  
    'Y': '[CTU]',  
sandbox:~/vcontrol/saferenz$
```

실수로 파일을 지움

```
sandbox: ~/vcontrol/safereniz ✕  
  
sandbox:~/vcontrol/saferenz$ ls  
find-safe-re.py      mylib.py      restriction-enzymes.txt  
human-miRNA-hairpins.fa  __pycache__  
sandbox:~/vcontrol/saferenz$ rm -f mylib.py  
sandbox:~/vcontrol/saferenz$
```

git checkout으로 위기 극복!

sandbox: ~/vcontrol/saferen... X

```
 sandbox:~/vcontrol/saferenz$ ls
find-safe-re.py      mylib.py      restriction-enzymes.txt
human-miRNA-hairpins.fa  __pycache__
 sandbox:~/vcontrol/saferenz$ rm -f mylib.py
 sandbox:~/vcontrol/saferenz$ ls
find-safe-re.py  human-miRNA-hairpins.fa  __pycache__  restriction-enzymes.txt
 sandbox:~/vcontrol/saferenz$ git checkout mylib.py
 sandbox:~/vcontrol/saferenz$ ls
find-safe-re.py      mylib.py      restriction-enzymes.txt
human-miRNA-hairpins.fa  __pycache__
 sandbox:~/vcontrol/saferenz$
```

```
usage: git [--version] [--help] [-C <path>] [-c <name>=<value>]
      [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
These are common Git commands used in various situations:

start a working area (see also: git help tutorial)
  clone      Clone a repository into a new directory
  init       Create an empty Git repository or reinitialize an existing one

work on the current change (see also: git help everyday)
  add        Add file contents to the index
  mv         Move or rename a file, a directory, or a symlink
  reset      Reset current HEAD to the specified state
  rm         Remove files from the working tree and from the index

examine the history and state (see also: git help revisions)
  bisect     Use binary search to find the commit that introduced a bug
  grep       Print lines matching a pattern
  log        Show commit logs
  show       Show various types of objects
  status     Show the working tree status

grow, mark and tweak your common history
  branch     List, create, or delete branches
  checkout   Switch branches or restore working tree files
  commit     Record changes to the repository
  diff       Show changes between commits, commit and working tree, etc
  merge      Join two or more development histories together
  rebase     Reapply commits on top of another base tip
  tag        Create, list, delete or verify a tag object signed with GPG

collaborate (see also: git help workflows)
  fetch      Download objects and refs from another repository
  pull       Fetch from and integrate with another repository or a local branch
  push       Update remote refs along with associated objects

'git help -a' and 'git help -g' list available subcommands and some
concept guides. See 'git help <command>' or 'git help <concept>'
to read about a specific subcommand or concept.
```

Git의 이모저모

31

버전 컨트롤?

- 주로 소프트웨어 프로젝트에서 **소스코드 관리**를 위해 만들어진 프로그램 또는 시스템. (형상관리라고도 불림)
 - 예: RCS, CVS, Subversion, Git, Mercurial, Perforce
- 여러 명이 **공동작업**할 때 특히 필수적임
 - 같은 코드를 여러 명이 효율적으로 **공유**하기 위해서
 - 현재 버전 뿐만 아니라 **예전 버전**을 모두 보관하기 위해서
 - 사람들이 어떤 걸 **바꾸고** 있는지 살펴보기 위해서
 - 여러 사람이 동시에 작업하다가 생긴 **충돌을 해결**하기 위해서
 - 소스코드 뿐만 아니라 **문서**나 **사진** 등도 넣을 수 있음

32

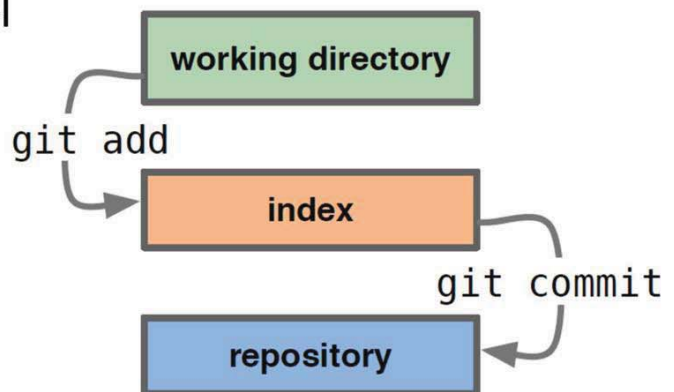
버전 컨트롤?

- 혼자 작업할 때는 전혀 필요 없나?
 - 혼자 일할 때도 **변화를 기록**하기에 매우 쉬워서 도움이 됨.
 - 특히 **타임머신**처럼 예전으로 돌아갈 때, 왜 그렇게 고쳤는지 기억나지 않을 때, 언제 **무엇 때문에 무엇과 같이** 그걸 고쳤는지 기억나지 않을 때 사용할 수 있음.
 - 동시에 **여러 가지 버전**을 유지해야 할 때도 사용할 수 있음.
 - 다른 사람의 소스코드를 **가져와서 쓸 때** 고치면서 쓰는 경우, 의존성을 잘 유지하려고 하는 경우 쓰면 편리함.

33

일반적인 워크플로우

- `git status`
변화점을 보고, 빠진 게 없는지 살핀다.
- 작업
- `git add`
커밋할 것들을 추가한다.
- `git commit -m "뭐라?"`
커밋해서 저장소로 보낸다.



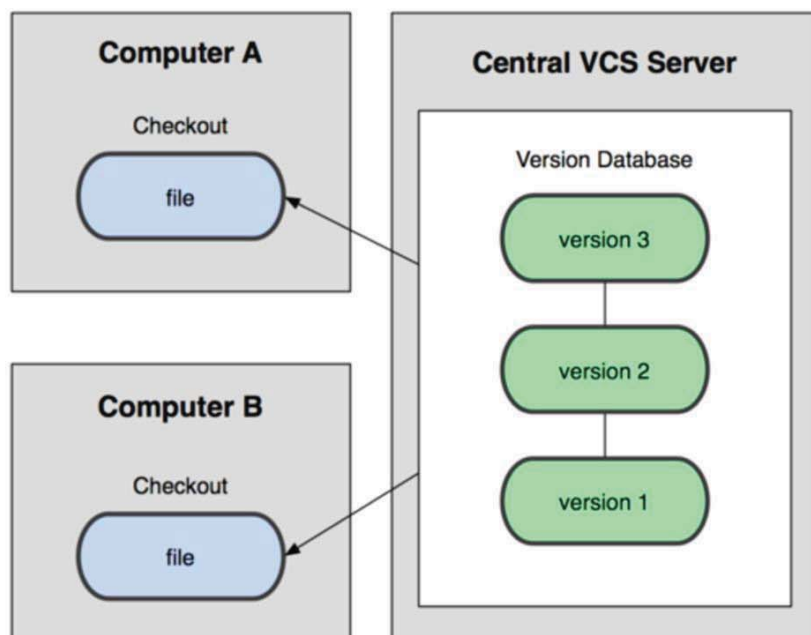
34

저장소 (Repository)

- 저장소 (Repository 또는 줄여서 Repo)
 - 모든 파일의 모든 역사를 보관해 두는 곳.
 - 직접 여기 안에 들어가서 고칠 일은 없음.
 - 변경은 모두 작업 디렉토리 (working directory)에서 함.
 - 커밋(commit)하면 저장소에 들어가게 됨.
- 고전적 버전컨트롤 도구들은 한 군데에만 저장소가 존재했음:
RCS, CVS, Subversion 등
- 최근에 쓰이는 분산 버전컨트롤 도구들은 모든 작업 디렉토리에 저장소를 같이 저장해두고 사용함:
Git, Mercurial 등
- 무엇을 저장소에 넣을지는 모두 수동으로 add 해야 함.

35

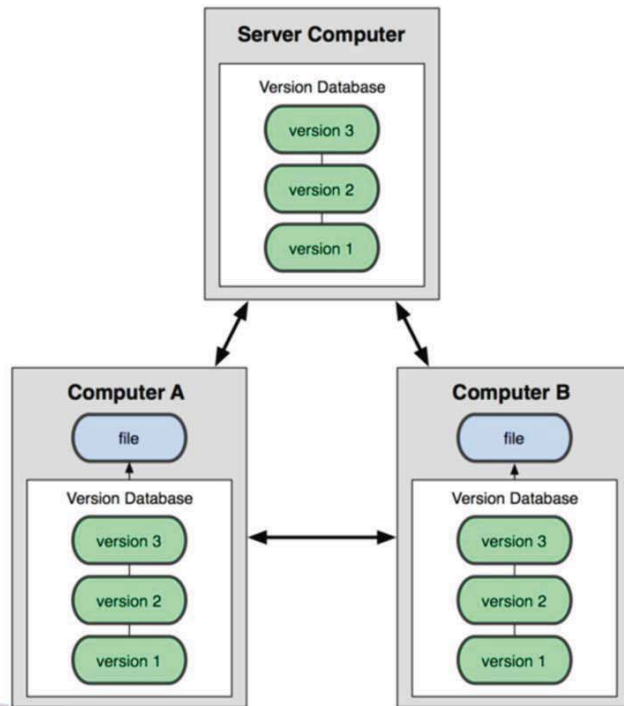
중앙집중식 저장소 (Centralized repository model)



36

분산식 저장소

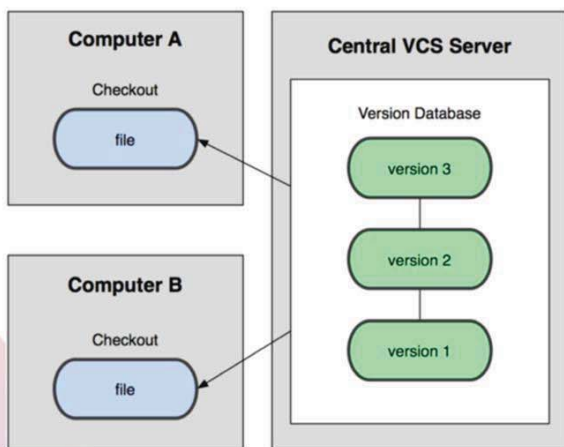
(Distributed repository model)



37

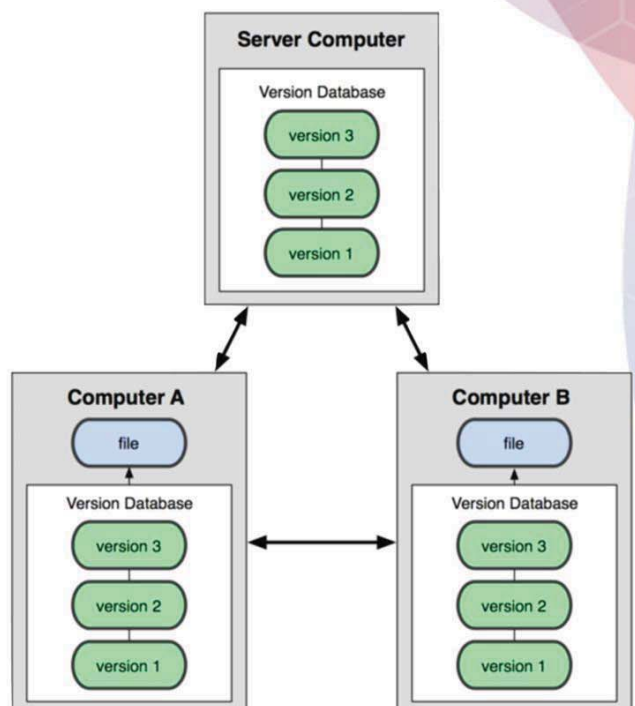
중앙집중식 저장소

(Centralized repository model)



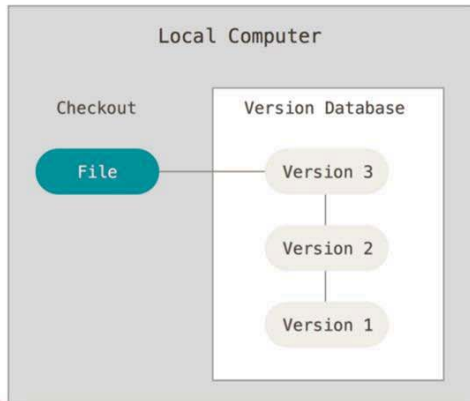
분산식 저장소

(Distributed repository model)



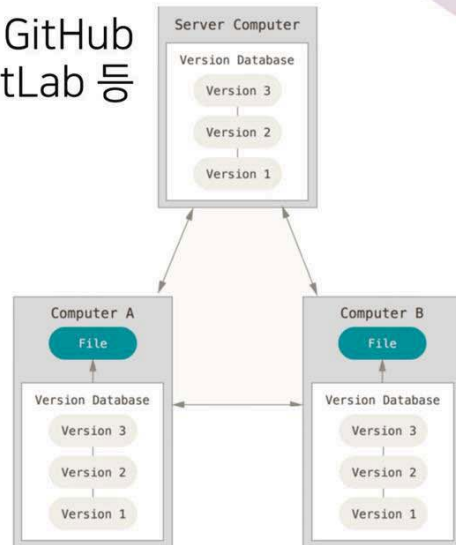
38

Git을 쓰는 다른 방법



내 컴퓨터에서 혼자 쓴다.

GitHub
GitLab 등



다른 중앙 서버에 올려두고
같이 주고 받고 하며 쓴다.

39

저장소엔 무엇을 넣고 무엇을 넣지 말까?

넣을 것

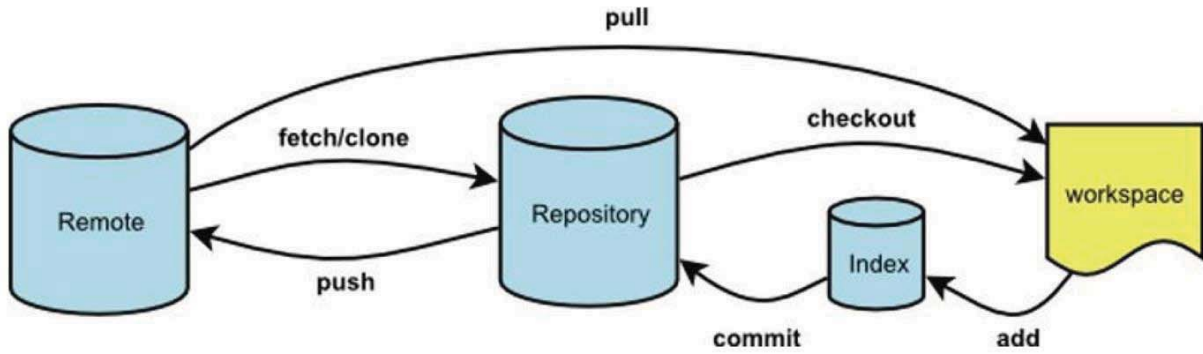
소스 코드 (.py .r)
파이프라인 (.sh Makefile)
환경설정 (.ini .yaml)
템플릿
변화를 기록해야 하는 데이터
작은 예제 데이터

넣지 말 것

실행 파일 (.exe .o)
데이터 파일 (.bam .fastq)
압축 파일 (.gz .tar .zip)
큰 파일
항상 막 바뀌는 파일
언제든지 새로 만들 수 있는 파일

40

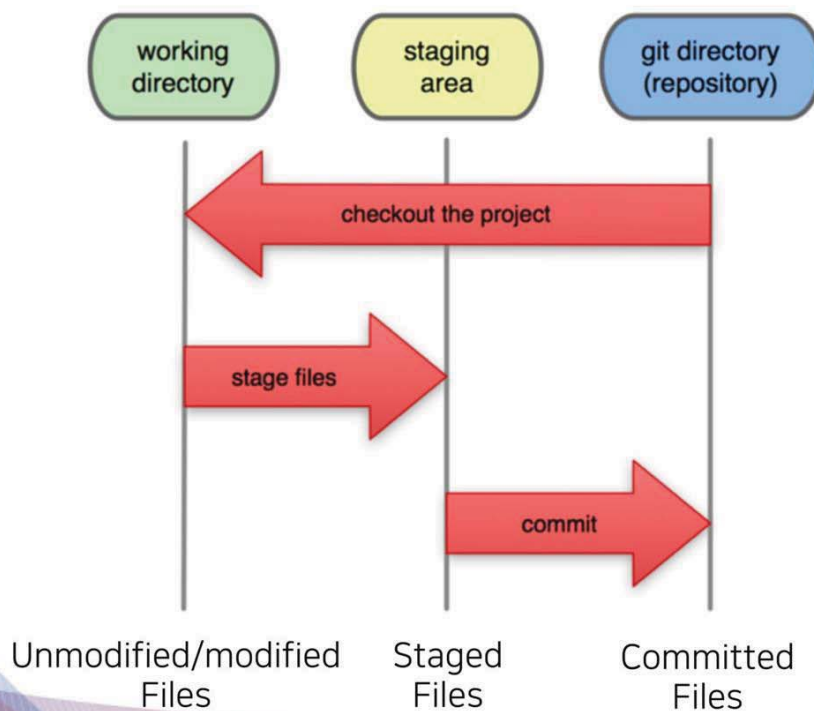
git 명령어가 하는 일



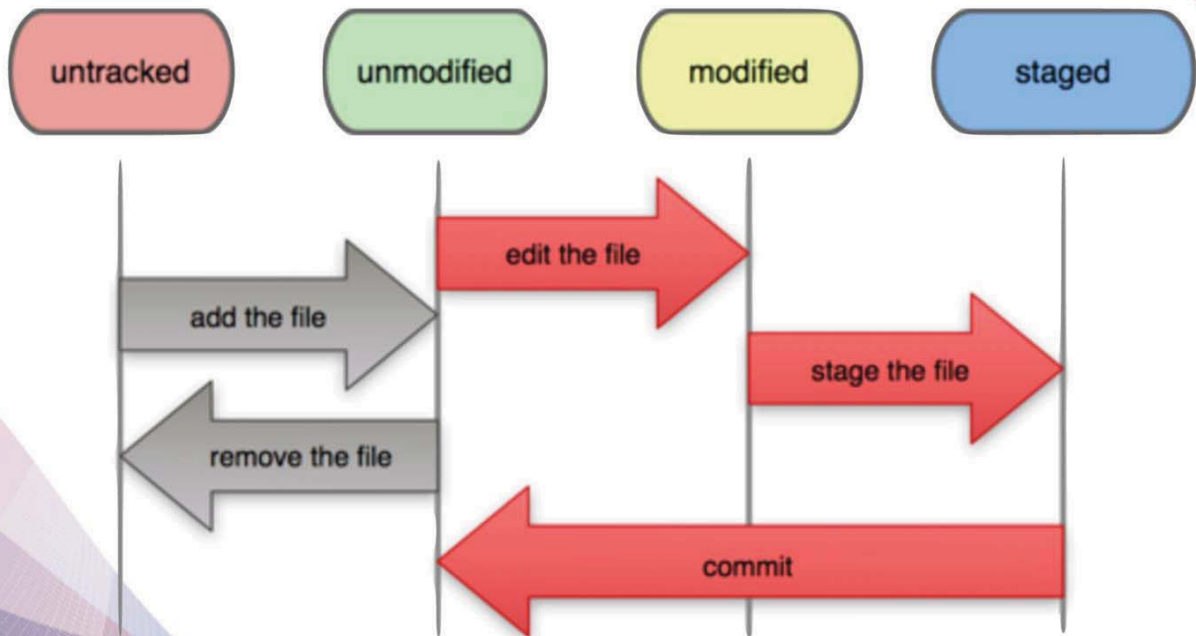
너무 복잡하면..

1. 몇 시간 더 연습해 본다.
2. GUI 클라이언트를 써 본다.

로컬 git 저장소에는 세 영역이 있다



작업 하면서 Git에 들어가게 되는 과정



43

커밋 메시지 쓰기

	COMMENT	DATE
○	CREATED MAIN LOOP & TIMING CONTROL	14 HOURS AGO
○	ENABLED CONFIG FILE PARSING	9 HOURS AGO
○	MISC BUGFIXES	5 HOURS AGO
○	CODE ADDITIONS/EDITS	4 HOURS AGO
○	MORE CODE	4 HOURS AGO
○	HERE HAVE CODE	4 HOURS AGO
○	AAAAAAAAA	3 HOURS AGO
○	ADKFJSLKDFJSDKLFJ	3 HOURS AGO
○	MY HANDS ARE TYPING WORDS	2 HOURS AGO
○	HAAAAAAAAAANDS	2 HOURS AGO

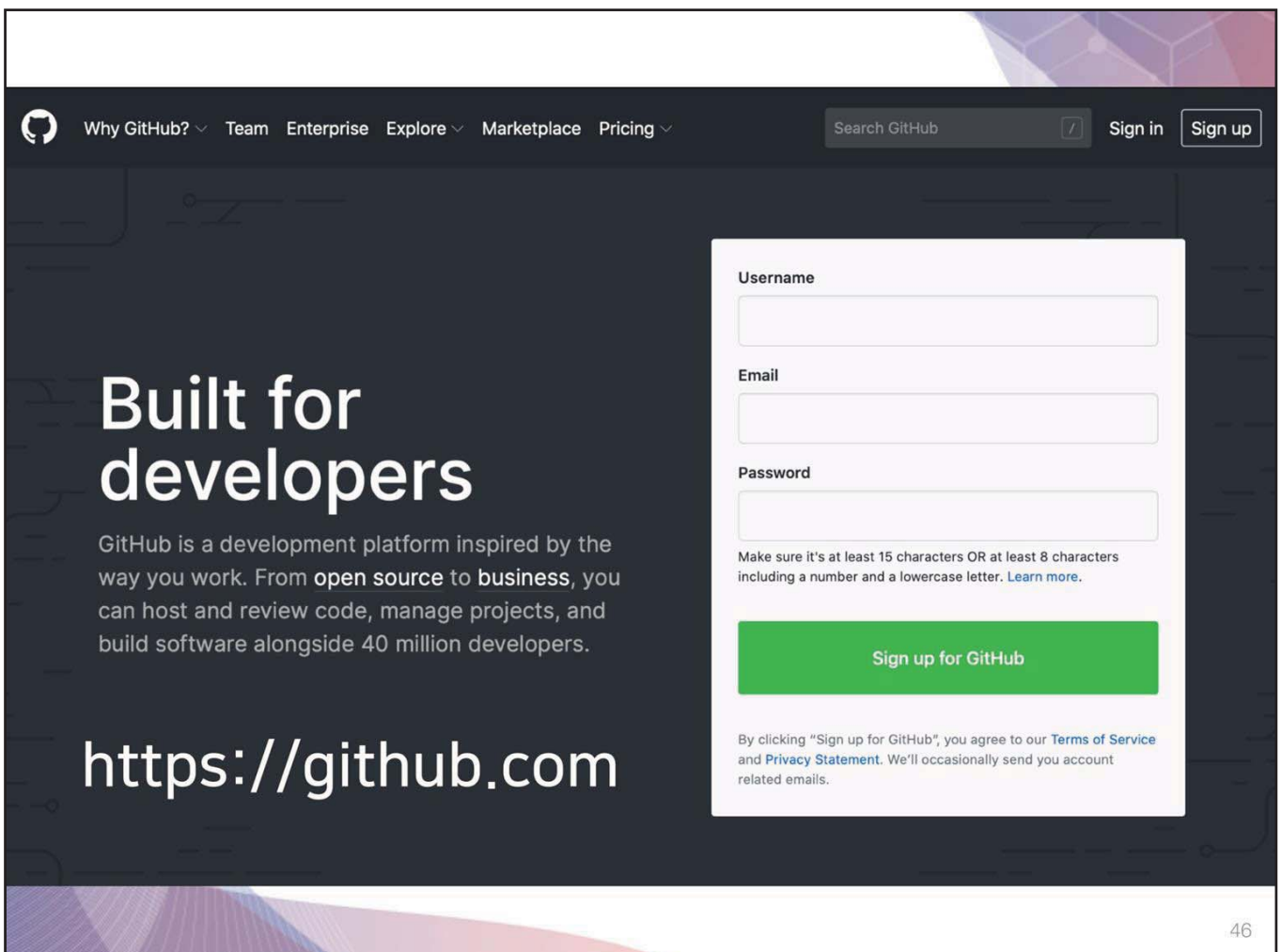
AS A PROJECT DRAGS ON, MY GIT COMMIT MESSAGES GET LESS AND LESS INFORMATIVE.

<http://xkcd.com/1296/>

커밋 메시지 쓰기

- 커밋 메시지는 60자 내외 짧은 글로
(문법이나 제대로 갖춘 문장구조는 보통 무시한다)
- 짧게 쓴다고 아무 말이나 쓰면 안 된다.
 - 나쁜 예: "Fixed a bunch of things"
 - 좋은 예: "Corrected the calculation of median scores"
- Git을 쓸 때는 최대한 커밋을 자주 하는 것이 유리하다. 커밋 메시지가 너무 길다면 더 자주 할 것.
- 팀 작업에서는 커밋 메시지에 credit을 다는 문화가 있는 곳도 있다.

45



The screenshot shows the GitHub sign-up page. At the top, there is a navigation bar with links for 'Why GitHub?', 'Team', 'Enterprise', 'Explore', 'Marketplace', and 'Pricing'. A search bar and 'Sign in' / 'Sign up' buttons are also present. The main content area features the GitHub logo and the text 'Built for developers'. Below this, a paragraph describes GitHub as a development platform. A large URL 'https://github.com' is displayed. On the right side, there is a sign-up form with fields for 'Username', 'Email', and 'Password'. A green 'Sign up for GitHub' button is at the bottom of the form. Below the button, there is a disclaimer about terms of service and privacy.

Why GitHub? Team Enterprise Explore Marketplace Pricing Search GitHub Sign in Sign up

Built for developers

GitHub is a development platform inspired by the way you work. From open source to business, you can host and review code, manage projects, and build software alongside 40 million developers.

https://github.com

Username

Email

Password

Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more.](#)

[Sign up for GitHub](#)

By clicking "Sign up for GitHub", you agree to our [Terms of Service](#) and [Privacy Statement](#). We'll occasionally send you account related emails.

46

프로젝트 새로 만들기

Search or jump to... Pull requests Issues Marketplace Explore

Explore Topics Trending Collections Events GitHub Sponsors

Here's what we found based on your interests...

Based on repositories you've starred

oushujun / EDTA 76 Stars

Code Issues Pull requests

Open Happy EDTA users with successful cases 4

oushujun commented on Aug 21, 2019

Hi all,

Just update the testing result. It seems that new release TIR can close this issue.

1. Please install a new env for the EDTA 20190802 release
2. Follow the step by [Read more](#) rovided.

good first issue

New repository Import repository New gist New organization New project

aristocratos / bashtop 1.8k Stars Linux resource monitor

microsoft / reverse-proxy 651 Stars A toolkit for developing high-performance HTTP reverse proxy applications.

fastai / fastbook 5.1k Stars Draft of the fastai book

GitHubDaily / GitHubDaily 6.6k Stars GitHubDaily 分享内容定期整理与分类。欢迎推荐、自荐项目，让更多人知道你的项目。

프로젝트 새로 만들기

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner: hyeshik / Repository name: bioitfund-learning-git

Great repository names are short and memorable. Need inspiration? [Read about reimagined-guacamole?](#)

Description (optional)

Public Anyone can see this repository. You choose who can commit.

Private You choose who can see and commit to this repository.

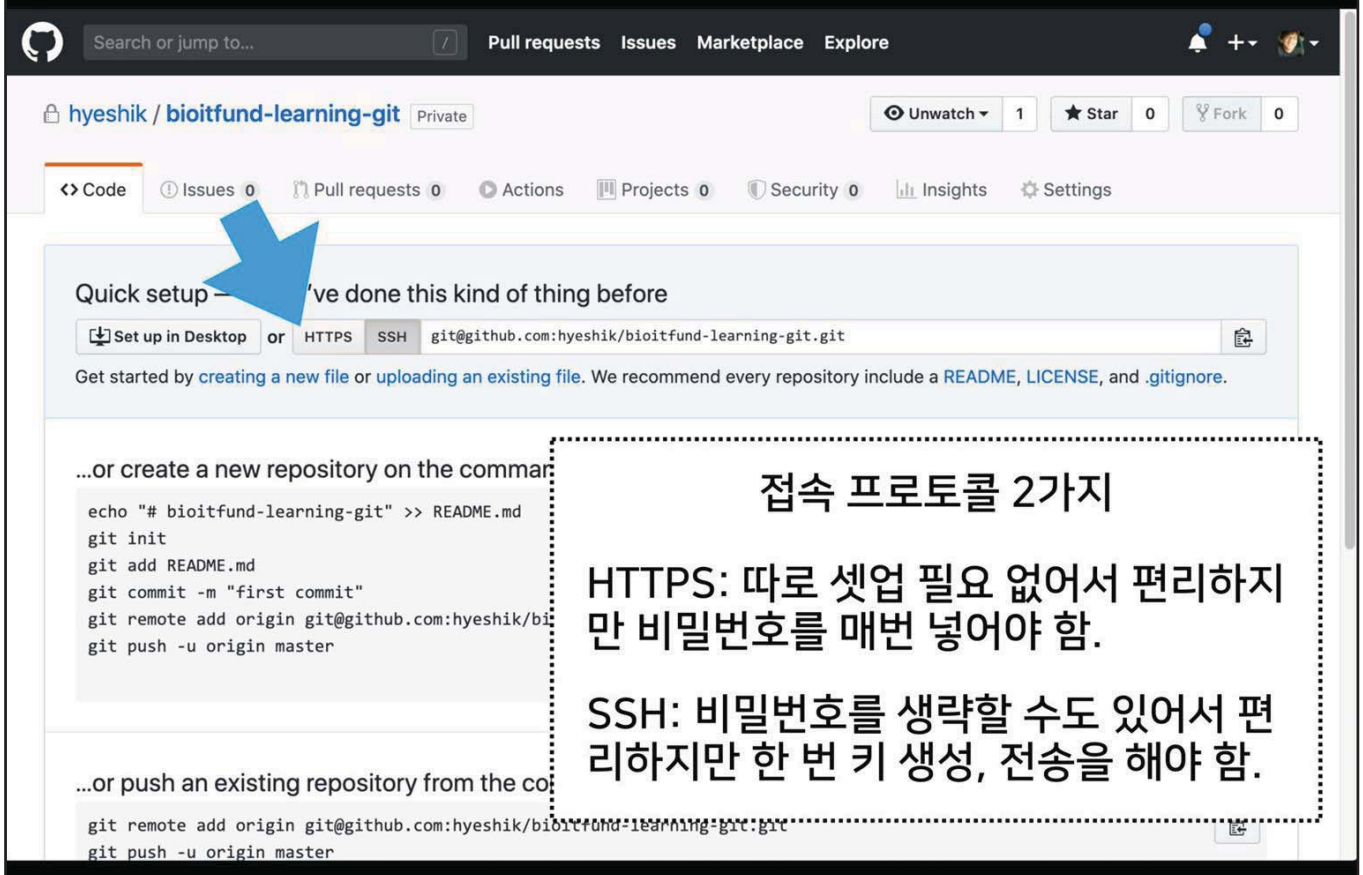
Skip this step if you're importing an existing repository.

Initialize this repository with a README This will let you immediately clone the repository to your computer.

Add .gitignore: None Add a license: None

Create repository

새 프로젝트 화면



hyeshik / bioitfund-learning-git Private

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Actions Projects 0 Security 0 Insights Settings

Quick setup — you've done this kind of thing before

Set up in Desktop or **HTTPS** SSH git@github.com:hyeshik/bioitfund-learning-git.git

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

...or create a new repository on the command line

```
echo "# bioitfund-learning-git" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin git@github.com:hyeshik/bioitfund-learning-git.git
git push -u origin master
```

...or push an existing repository from the command line

```
git remote add origin git@github.com:hyeshik/bioitfund-learning-git.git
git push -u origin master
```

접속 프로토콜 2가지

HTTPS: 따로 셋업 필요 없어서 편리하지만 비밀번호를 매번 넣어야 함.

SSH: 비밀번호를 생략할 수도 있어서 편리하지만 한 번 키 생성, 전송을 해야 함.

GitHub측 remote 주소 등록하기

sandbox: ~/vcontrol/safereiz

```
sandbox:~/vcontrol/safereiz$ git remote add origin https://github.com/hyeshik/bioitfund-learning-git.git
sandbox:~/vcontrol/safereiz$
```

GitHub으로 보내기

sandbox: ~/vcontrol/safere

```
sandbox:~/vcontrol/safere$ git push -u origin master
Username for 'https://github.com': hyeshik
Password for 'https://hyeshik@github.com':
Counting objects: 9, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (9/9), done.
Writing objects: 100% (9/9), 10.79 KiB | 1.80 MiB/s, done.
Total 9 (delta 2), reused 0 (delta 0)
remote: Resolving deltas: 100% (2/2), done.
To https://github.com/hyeshik/bioitfund-learning-git.git
 * [new branch]      master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
sandbox:~/vcontrol/safere$
```

Two factor authentication을 쓰는 경우...

The screenshot shows the GitHub Developer settings page for creating a new personal access token. The page title is "New personal access token". Below the title, there is a note explaining that personal access tokens function like ordinary OAuth access tokens and can be used instead of a password for Git over HTTPS, or to authenticate to the API over Basic Authentication. A text input field contains the name "sandbox". Under the "Select scopes" section, there is a list of scopes with checkboxes. The "repo" scope is selected, and its sub-scopes are also checked: "repo:status", "repo_deployment", "public_repo", and "repo:invite". Other scopes like "write:packages", "read:packages", and "delete:packages" are not selected.

github.com/settings/tokens/new

Settings / Developer settings

GitHub Apps
OAuth Apps
Personal access tokens

New personal access token

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

Note

sandbox

What's this token for?

Select scopes

Scopes define the access for personal tokens. [Read more about OAuth scopes.](#)

<input checked="" type="checkbox"/> repo	Full control of private repositories
<input checked="" type="checkbox"/> repo:status	Access commit status
<input checked="" type="checkbox"/> repo_deployment	Access deployment status
<input checked="" type="checkbox"/> public_repo	Access public repositories
<input checked="" type="checkbox"/> repo:invite	Access repository invitations
<input type="checkbox"/> write:packages	Upload packages to github package registry
<input type="checkbox"/> read:packages	Download packages from github package registry
<input type="checkbox"/> delete:packages	Delete packages from github package registry

Repository 잘 올라갔는지 확인하기

The screenshot shows the GitHub interface for a repository named 'bioitfund-learning-git' by user 'hyeshik'. The repository is private and has 1 watch, 0 stars, and 0 forks. The main content area shows a commit history table with the following entries:

Commit Message	Time Ago
Jane Doe Fix problem on detection in RNA seqs	Latest commit 5b54247 2 hours ago
find-safe-re.py	Import initial repository. 11 hours ago
human-miRNA-hairpins.fa	Import initial repository. 11 hours ago
mylib.py	Fix problem on detection in RNA seqs 2 hours ago
restriction-enzymes.txt	Import initial repository. 11 hours ago

At the bottom of the commit list, there is a prompt: "Add a README with an overview of your project." with a green button labeled "Add a README".

GitHub 프로젝트 공간 살펴보기

README.md 만들기

수정사항 당겨오기

sandbox: ~/vcontrol/safere

(use "git add <file>..." to include in what will be committed)

```
.ipython_checkpoints/  
__pycache__/
```

nothing added to commit but untracked files present (use "git add" to track)

```
sandbox:~/vcontrol/safere$ git pull
```

```
Username for 'https://github.com': hyeshik
```

```
Password for 'https://hyeshik@github.com':
```

```
Already up to date.
```

```
sandbox:~/vcontrol/safere$ git pull
```

```
Username for 'https://github.com': hyeshik
```

```
Password for 'https://hyeshik@github.com':
```

```
remote: Enumerating objects: 4, done.
```

```
remote: Counting objects: 100% (4/4), done.
```

```
remote: Compressing objects: 100% (3/3), done.
```

```
remote: Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
```

```
Unpacking objects: 100% (3/3), done.
```

```
From https://github.com/hyeshik/bioitfund-learning-git
```

```
5b54247..121344b master -> origin/master
```

```
Updating 5b54247..121344b
```

```
Fast-forward
```

```
 README.md | 9 ++++++++
```

```
1 file changed, 9 insertions(+)
```

```
create mode 100644 README.md
```

```
sandbox:~/vcontrol/safere$ ls
```

```
find-safe-re.py          mylib.py                README.md
```

```
human-miRNA-hairpins.fa __pycache__             restriction-enzymes.txt
```

README.md 수정

수정사항 다시 올려 보내기

sandbox: ~/vcontrol/safere

Date: Mon Apr 27 02:50:28 2020 +0900

Create README

commit 5b542471bbb71a5615729433bcaddad3c36155b7

Author: Jane Doe <janedoe@snu.ac.kr>

Date: Mon Apr 27 00:27:48 2020 +0900

Fix problem on detection in RNA seqs

commit 6d3a09d02b95534595976d95def748231dlff770

Author: Jane Doe <janedoe@snu.ac.kr>

Date: Sun Apr 26 16:12:15 2020 +0900

Import initial repository.

sandbox:~/vcontrol/safere\$ git push -u origin master

Username for 'https://github.com': hyeshik

Password for 'https://hyeshik@github.com':

Counting objects: 3, done.

Delta compression using up to 8 threads.

Compressing objects: 100% (3/3), done.

Writing objects: 100% (3/3), 326 bytes | 326.00 KiB/s, done.

Total 3 (delta 2), reused 0 (delta 0)

remote: Resolving deltas: 100% (2/2), completed with 2 local objects.

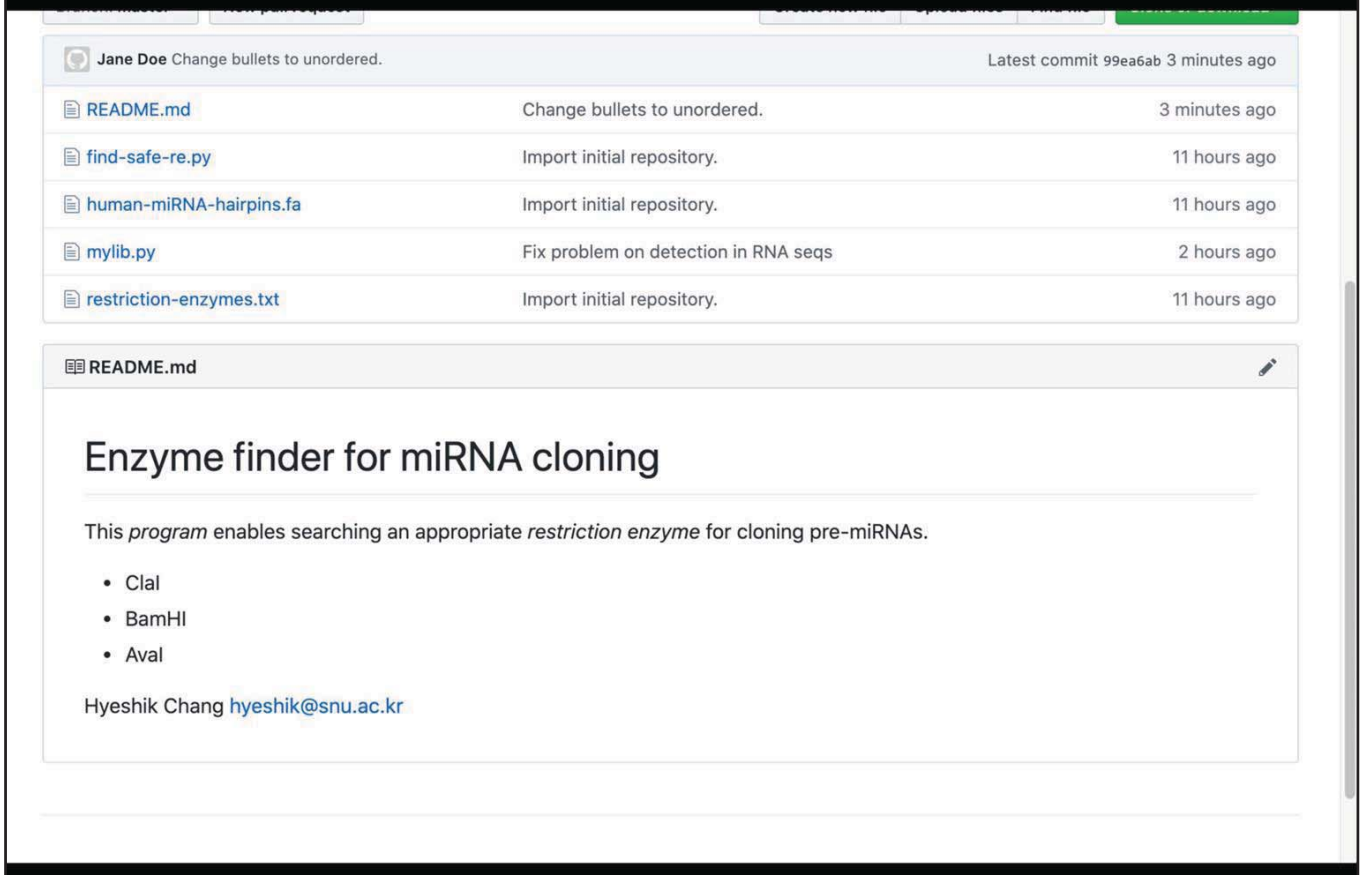
To https://github.com/hyeshik/bioitfund-learning-git

121344b..99ea6ab master -> master

Branch 'master' set up to track remote branch 'master' from 'origin'.

sandbox:~/vcontrol/safere\$

수정 완료 확인하기



The screenshot shows a GitHub repository interface. At the top, a commit by Jane Doe is highlighted, with the message "Change bullets to unordered." and a timestamp of "3 minutes ago". Below this, a list of commits is shown, including files like README.md, find-safe-re.py, human-miRNA-hairpins.fa, mylib.py, and restriction-enzymes.txt. The README.md file is selected, showing its content: "Enzyme finder for miRNA cloning". The text describes a program for searching restriction enzymes and lists Clal, BamHI, and Aval as examples. The author is identified as Hyesik Chang with the email hyesik@snu.ac.kr.

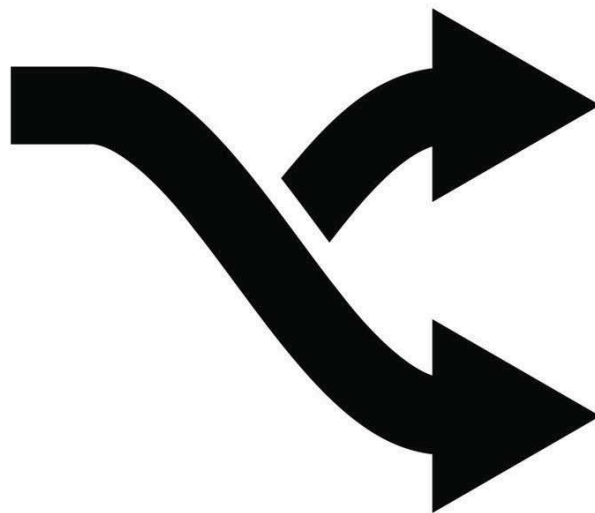
File	Commit Message	Time
README.md	Change bullets to unordered.	3 minutes ago
find-safe-re.py	Import initial repository.	11 hours ago
human-miRNA-hairpins.fa	Import initial repository.	11 hours ago
mylib.py	Fix problem on detection in RNA seqs	2 hours ago
restriction-enzymes.txt	Import initial repository.	11 hours ago

Enzyme finder for miRNA cloning

This *program* enables searching an appropriate *restriction enzyme* for cloning pre-miRNAs.

- Clal
- BamHI
- Aval

Hyesik Chang hyeshik@snu.ac.kr



Forking

Fork 하기

github.com/hyeshik/snubioinfo-test-git

hyeshik / snubioinfo-test-git

Unwatch 1 Star 0 Fork 0

Code Issues 0 Pull requests 0 Actions Projects 0 Wiki Security 0 Insights

No description, website, or topics provided.

Manage topics

1 commit 1 branch 0 packages 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find file Clone or download

hyeshik Import the initial files Latest commit c9d2c69 1 minute ago

File	Commit Message	Time
README.md	Import the initial files	1 minute ago
Snakefile	Import the initial files	1 minute ago
gen-candidates.py	Import the initial files	1 minute ago

README.md

Example project for SNU bioinformatics class

Fork한 프로젝트 당겨오기

github.com

ChangLabSNU / snubioinfo-test-git

Watch 0 Star 0 Fork 1

Code Pull requests 0 Actions Projects 0 Wiki Security 0 Insights Settings

No description, website, or topics provided.

Manage topics

1 commit 1 branch 0 packages 0 releases 1 contributor

Branch: master New pull request Create new file Upload files Find file Clone or download

This branch is even with hyeshik:master.

hyeshik Import the initial files

File	Commit Message	Time
README.md	Import the initial files	
Snakefile	Import the initial files	
gen-candidates.py	Import the initial files	

README.md

Example project for SNU bioinformatics class

Clone with SSH Use HTTPS

Use a password protected SSH key.

git@github.com:ChangLabSNU/snubioinfo-te

Open in Desktop Download ZIP

2 minutes ago

Fork한 프로젝트 당겨오기

```
sandbox: ~/vcontrol/snubio X
sandbox:~/vcontrol/saferenz$ git commit -a
[master a884e71] This is an example project for a class.
 1 file changed, 4 insertions(+), 2 deletions(-)
sandbox:~/vcontrol/saferenz$ ls
find-safe-re.py      mylib.py      README.md
human-miRNA-hairpins.fa  __pycache__  restriction-enzymes.txt
sandbox:~/vcontrol/saferenz$ git push
Username for 'https://github.com': hyeshik
Password for 'https://hyeshik@github.com':
Counting objects: 3, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 381 bytes | 381.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/hyeshik/bifclass-git.git
 99ea6ab..a884e71 master -> master
sandbox:~/vcontrol/saferenz$ cd ..
sandbox:~/vcontrol$ git clone https://github.com/내 사용자명 /snubioinfo-test-git.git
Cloning into 'snubioinfo-test-git'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 5 (delta 0), reused 5 (delta 0), pack-reused 0
Unpacking objects: 100% (5/5), done.
sandbox:~/vcontrol$ cd snubioinfo-test-git/
sandbox:~/vcontrol/snubioinfo-test-git$
```

받은 파일 고쳐보기

```
sandbox: ~/vcontrol/snubio X  README.md X
1 # Example project for SNU bioinformatics class
2
3 ## Spike-in sequence generator for small RNA sequencing
4
5 This program generates many random sequences following the
6 sequence profiles of human mature microRNA sequences.
7 It produces the final list of the sequences after filtering
8 out the potential matches to human, mouse, Caenorhabditis elegans, or
9 zebrafish microRNAs.
10
11 Hyeshik Chang
12
```

받은 파일 고쳐보기

```
sandbox: ~/vcontrol/snubio x README.md x
1 # Example project for SNU bioinformatics class
2
3 ## Spike-in sequence generator for small RNA sequencing
4
5 This program generates many random sequences following the
6 sequence profiles of human mature microRNA sequences.
7 It produces the final list of the sequences after filtering
8 out the potential matches to human, mouse, Caenorhabditis elegans, or
9 zebrafish microRNAs.
10
11 Hyeshik Chang
12
```

고친 내용 커밋하고 다시 푸시

```
sandbox: ~/vcontrol/snubio x
sandbox:~/vcontrol/snubioinfo-test-git$ git commit -a -m 'Change Xenopus to C. elegans'
[master 0751568] Change Xenopus to C. elegans
1 file changed, 1 insertion(+), 1 deletion(-)
sandbox:~/vcontrol/snubioinfo-test-git$ git push -u origin
Username for 'https://github.com': hyeshik
Password for 'https://hyeshik@github.com':
Counting objects: 3, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 382 bytes | 382.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/ChangLabSNU/snubioinfo-test-git.git
c9d2c69..0751568 master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
sandbox:~/vcontrol/snubioinfo-test-git$
```

고친 내용이 적용되었는지 확인

Branch: master | New pull request | Create new file | Upload files | Find file | Clone or download

README.md

This branch is 1 commit ahead of hyeshik:master. Pull request Compare

Jane Doe	Change Xenopus to C. elegans	commit 0751568 1 minute ago
README.md	Change Xenopus to C. elegans	1 minute ago
Snakefile	Import the initial files	8 minutes ago
gen-candidates.py	Import the initial files	8 minutes ago

Example project for SNU bioinformatics class

Spike-in sequence generator for small RNA sequencing

This program generates many random sequences following the sequence profiles of human mature microRNA sequences. It produces the final list of the sequences after filtering out the potential matches to human, mouse, *Caenorhabditis elegans*, or zebrafish microRNAs.

Hyeshik Chang

<https://github.com/ChangLabSNU/snubioinfo-test-git/blob/master/README.md>

원저자의 저장소로 pull request 보내기

Comparing changes

Choose two branches to see what's changed or to start a new pull request. If you need to, you can also [compare across forks](#).

base repository: hyeshik/snubioinfo-test-git | base: master | head repository: ChangLabSNU/snubioinfo-te... | compare: master

✓ Able to merge. These branches can be automatically merged.

[Create pull request](#) Discuss and review the changes in this comparison with others.

1 commit | 1 file changed | 0 commit comments | 1 contributor

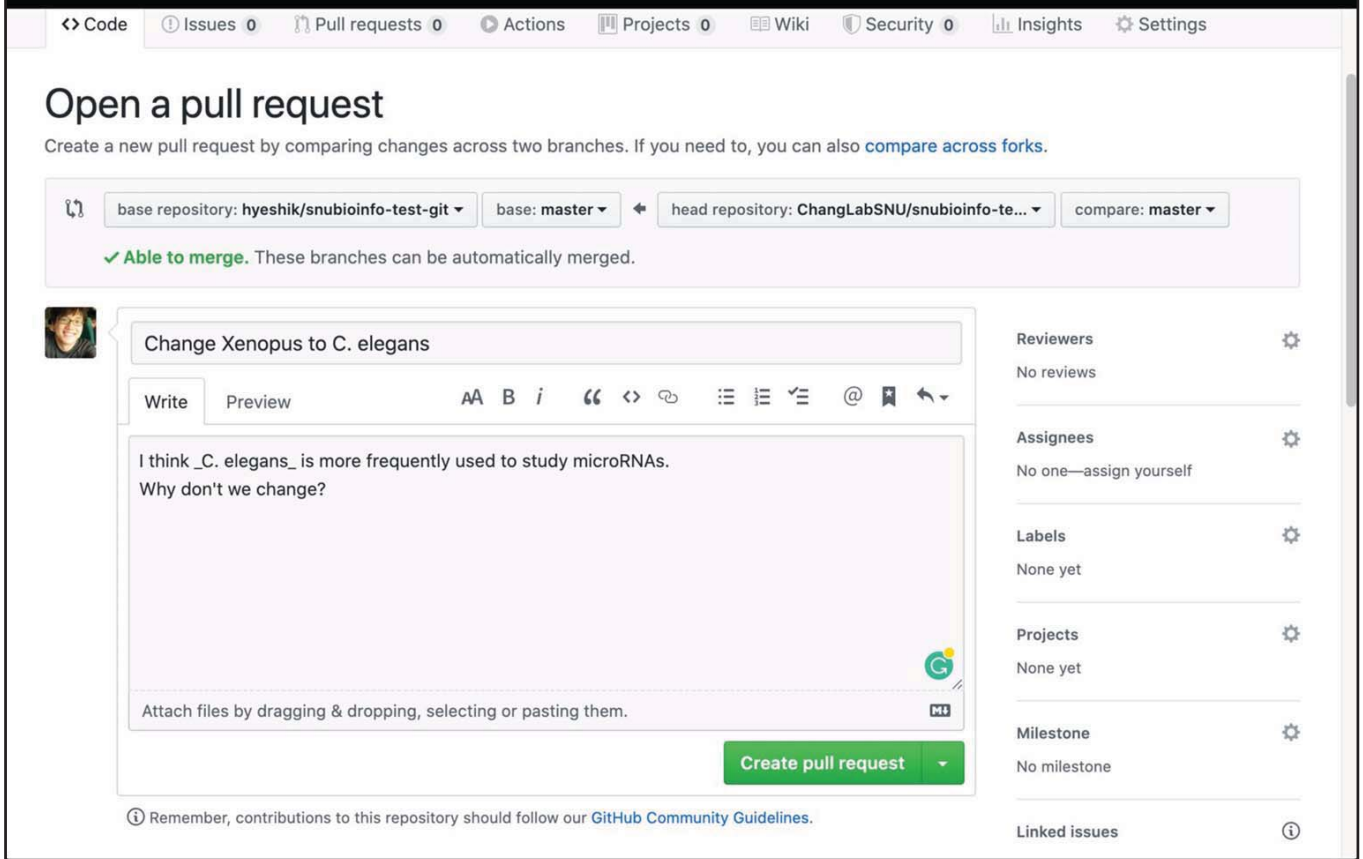
Commits on Apr 27, 2020

Jane Doe | Change Xenopus to C. elegans | 0751568

Showing 1 changed file with 1 addition and 1 deletion. Unified Split

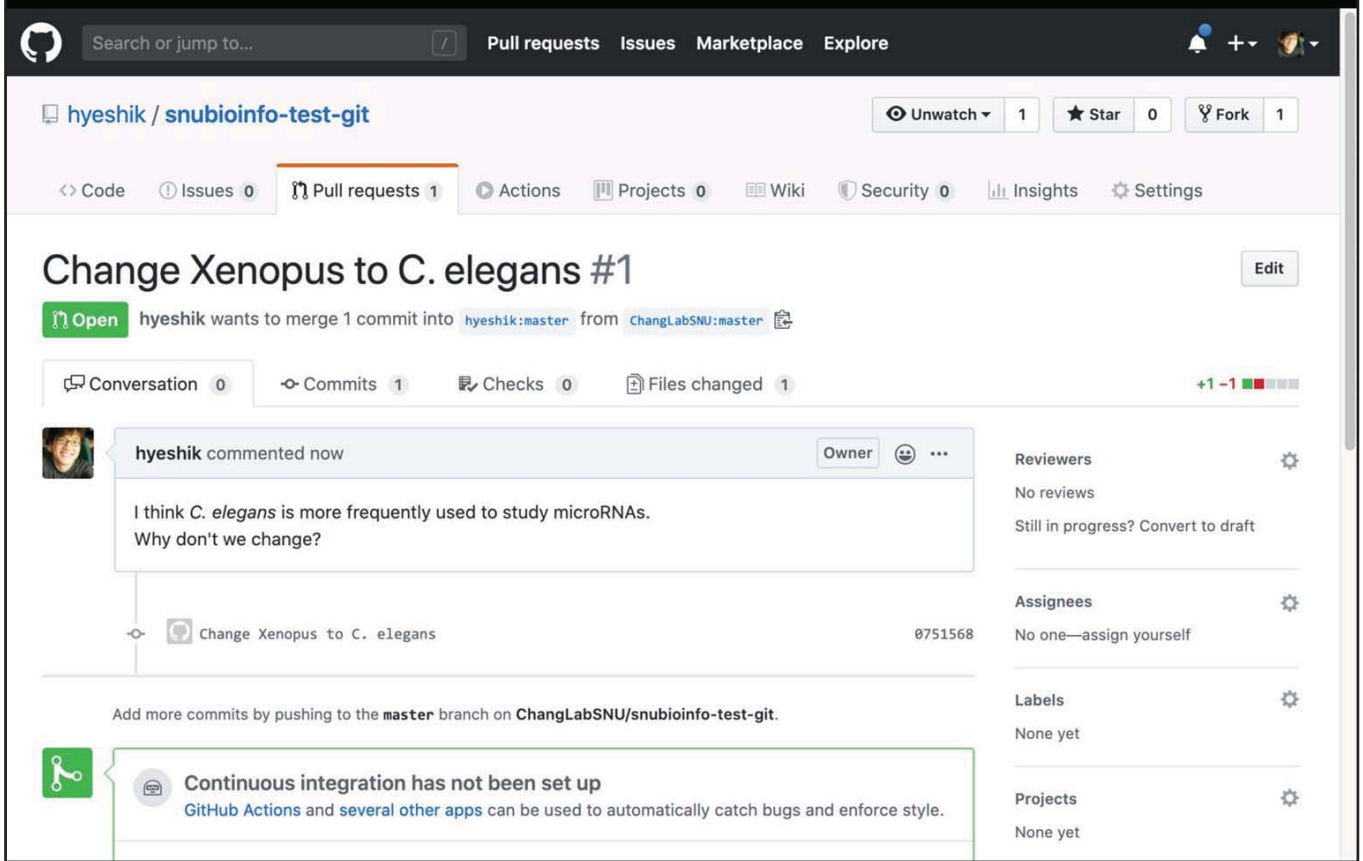
```
@@ -5,7 +5,7 @@
5 5 This program generates many random sequences following the
6 6 sequence profiles of human mature microRNA sequences.
7 7 It produces the final list of the sequences after filtering
8 - out the potential matches to human, mouse, Xenopus Laevis, or
8 + out the potential matches to human, mouse, Caenorhabditis eLegans, or
9 9 zebrafish microRNAs.
10 10
```

Pull request 메시지 작성하기



The screenshot shows the GitHub interface for creating a pull request. At the top, there are navigation tabs for Code, Issues (0), Pull requests (0), Actions, Projects (0), Wiki, Security (0), Insights, and Settings. The main heading is "Open a pull request" with a subtext: "Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#)." Below this, there are dropdown menus for "base repository: hyeshik/snubioinfo-test-git", "base: master", "head repository: ChangLabSNU/snubioinfo-te...", and "compare: master". A green checkmark indicates "Able to merge. These branches can be automatically merged." The main content area has a title "Change Xenopus to C. elegans" and a rich text editor with the text: "I think `_C. elegans_` is more frequently used to study microRNAs. Why don't we change?". A "Create pull request" button is visible. On the right, there are sections for Reviewers, Assignees, Labels, Projects, Milestone, and Linked issues, all currently empty.

Pull request 처리 중 토론

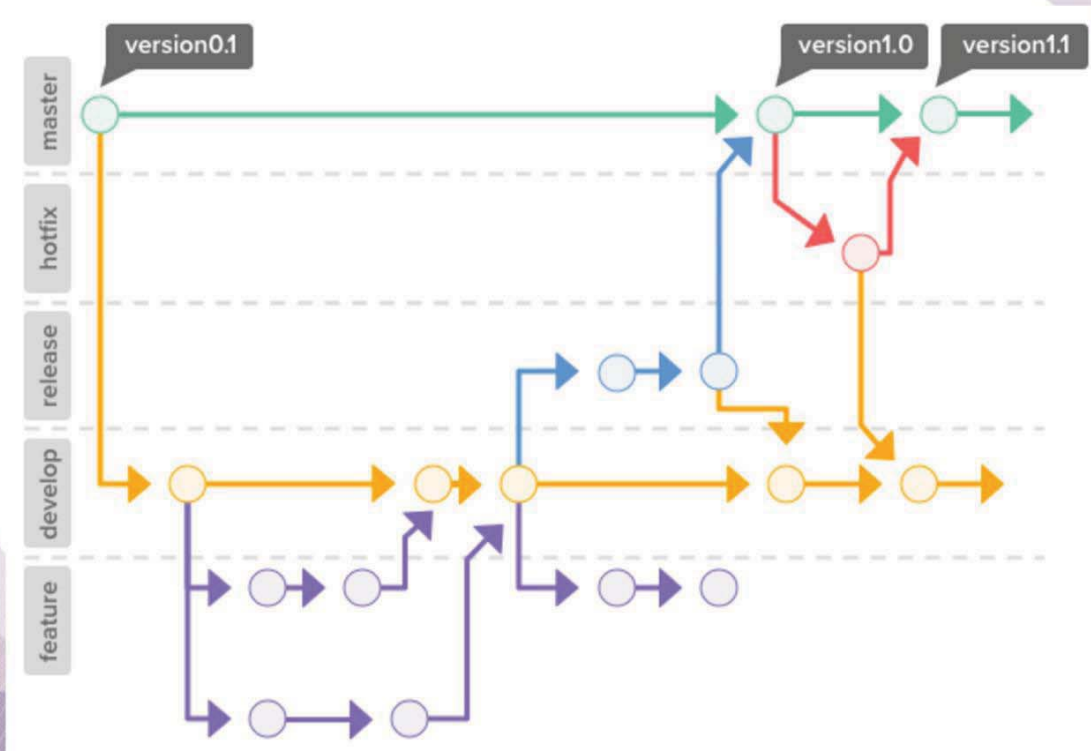


The screenshot shows the GitHub interface for a pull request titled "Change Xenopus to C. elegans #1". The repository is "hyeshik / snubioinfo-test-git". The pull request is from "ChangLabSNU:master" to "hyeshik:master". The main comment by "hyeshik" says: "I think `C. elegans` is more frequently used to study microRNAs. Why don't we change?". Below the comment, there is a link to the file "Change Xenopus to C. elegans". At the bottom, there is a notification: "Continuous integration has not been set up. GitHub Actions and several other apps can be used to automatically catch bugs and enforce style." The right sidebar shows the same metadata as the previous screenshot.

Branching

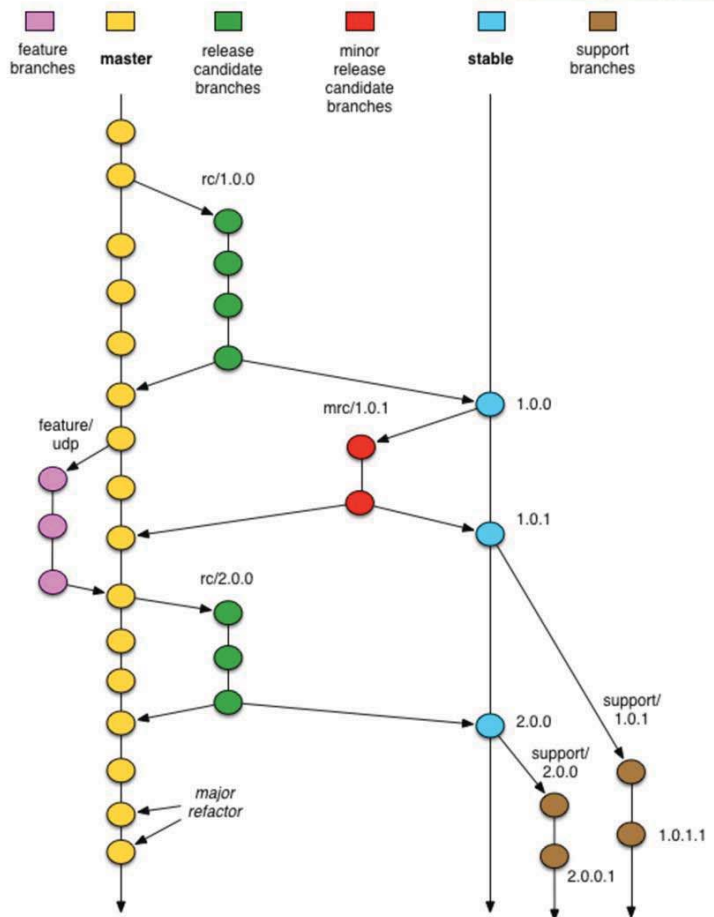
71

브랜칭 (Branching)



72

Master를 개발 브랜치로 쓰는 경우



73

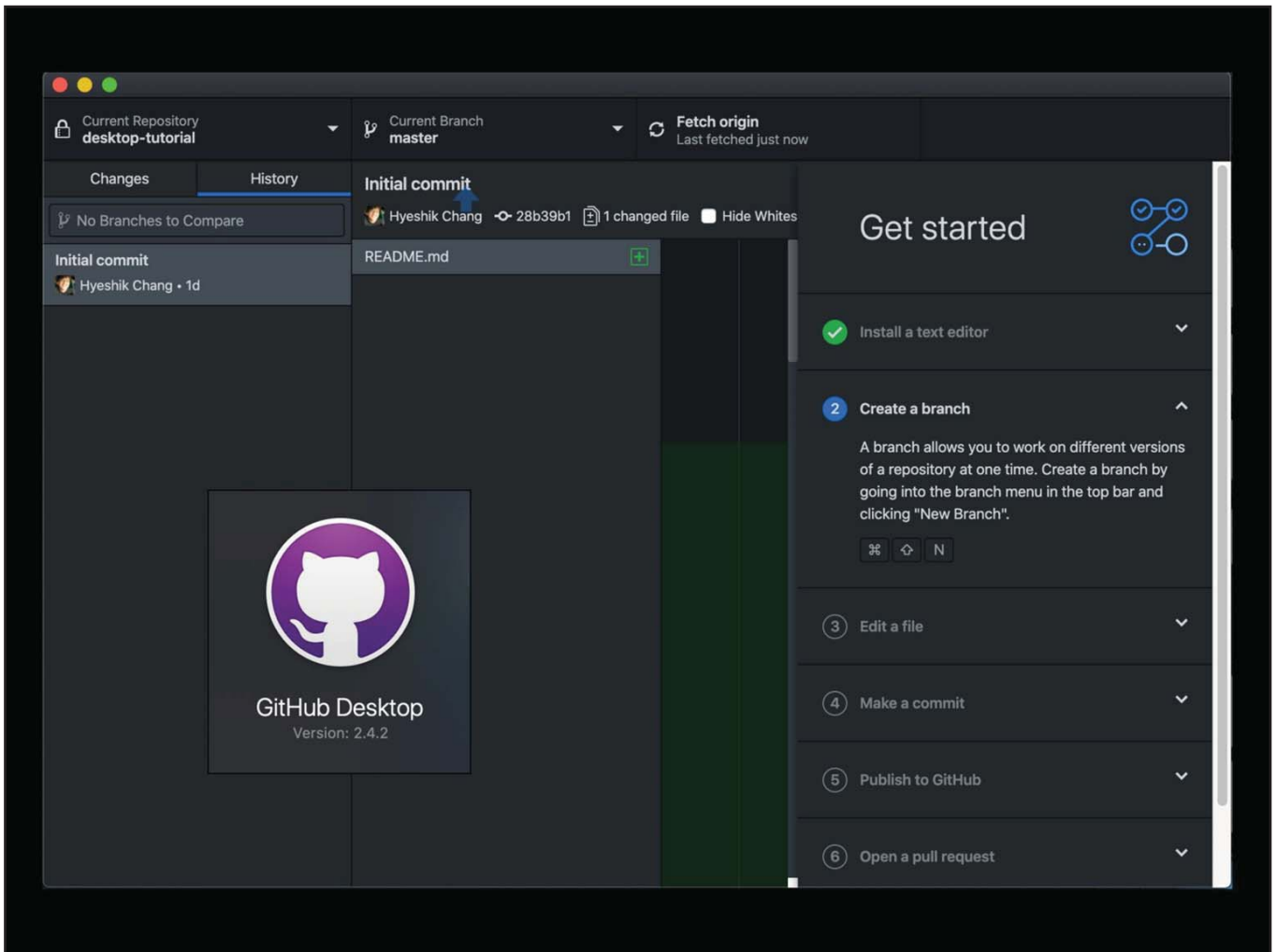
The screenshot shows the 'Learn Git Branching' website in a browser. On the left, a terminal window displays the following content:

```

$ level rampup3
$ hint
You'll need to use at least one direct
reference (hash) to complete this
level
$ delay 2000
$ show goal
  
```

On the right, a branching diagram shows a central 'HEAD' pointing to commit C1. C1 branches into C2 (HEAD) and C3. C2 branches into C4 (master) and C5. C3 branches into C5 and C6. A 'bugFix' branch is shown branching off from C5. A 'Fork me on GitHub' banner is visible in the top right corner.

<https://learngitbranching.js.org>



버전 컨트롤

- 코드의 역사를 기록한다.
- 공동작업 뿐만 아니라 홀로 작업할 때도 매우 유용하다.
- 최대한 기록은 자주 남기는 것이 좋다.
- GitHub, GitLab 등을 이용하면 이슈 트래킹, 자동 테스트, 웹 사이트 호스팅 등과 연계 가능하다.

KSBi-BIML 2022

Beginning Version Control for Bioinformatics

장혜식

서울대학교 생명과학부