PhysLean

Public

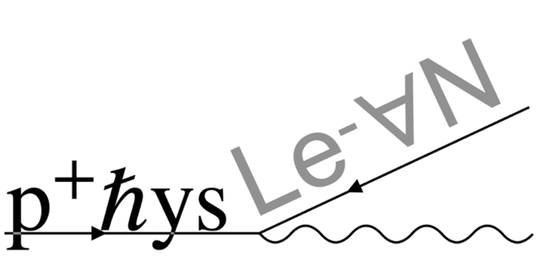
Edit Pins | Unwatch 6 | Fork 26 | Starred 278

master | 14 Branches | 9 Tags | Go to file t | Add file | Code

jstoobysmith feat: sorryful attribute and linter (#627) eBfbff6 · 2 days ago 2,089 Commits

.github	chore: Remove bot comment on PRs (#623)	last week
.vscode	feat: sorryful attribute and linter (#627)	2 days ago
PhysLean	feat: sorryful attribute and linter (#627)	2 days ago
docs	chore: Bump to v4.20.0-rc5 (#566)	last month
scripts	feat: sorryful attribute and linter (#627)	2 days ago
.gitignore	chore: Remove .DS_Store	11 months ago
.gitpod.yml	Add .gitpod.yml	8 months ago
CONTRIBUTING.md	Update CONTRIBUTING.md (#499)	2 months ago
LICENSE	Create LICENSE	last year
PhysLean.lean	feat: sorryful attribute and linter (#627)	2 days ago
README.md	Update README.md (#625)	5 days ago
lake-manifest.json	feat: Update (#593)	last month
lakefile.toml	feat: Update (#593)	last month
lean-toolchain	feat: Update (#593)	last month

README | Apache-2.0 license



An open-source, community, project to digitalize results from physics into Lean 4.

- Getting Started | The Website | How To Get Involved | PhysLean Zulip Discussion | TODO List
- PhysLean Search | PhysLean Online
- View The Stats | Lean v4.20.1 | Gitpod ready-to-code | Ask DeepWiki

Aims of this project

🎯 **Digitalize** results (meaning calculations, definitions, and theorems) from physics into Lean 4.

🎯 Develop structures to aid the **creation** of new results in physics using Lean, with the potential future use of AI.

🎯 Create good documentation so that the project can be used for **pedagogical** purposes.

How to get involved

See the Get Involved for more details. Some suggestions:

🛤️ write **informal** results - no need to learn Lean for this - see the Getting Started page for more details,

🛤️ tackle a TODO item,

🛤️ or, start formalizing an area that you find intresting.

Feel free to come to the PhysLean zulip to ask questions and advice.

Places in the project to start

Good places to start an exploration of the project.

- 📁💻 **Maxwell's equations** in electromagnetism.
- 📁💻 **Quantum Harmonic Oscillator** in quantum mechanics.
- 📁💻 The two state **canonical ensemble** in statistical mechanics.
- 📁💻 The **tight-binding model** in condensed matter physics
- 📁💻 The **twin paradox** in special relativity.
- 📁💻 The **two-Higgs doublet model** in particle physics
- 📁💻 **Wick's theorem** in quantum field theory.

Associated media and publications

- 📄 Joseph Tooby-Smith, **HepLean: Digitalising high energy physics**, Computer Physics Communications, Volume 308, 2025, 109457, ISSN 0010-4655, https://doi.org/10.1016/j.cpc.2024.109457. [arXiv:2405.08863]
- 📄 Joseph Tooby-Smith, **Formalization of physics index notation in Lean 4**, arXiv:2411.07667
- 📄 Joseph Tooby-Smith, **Digitalizing Wick's Theorem**, arXiv:2505.07939
- 📽️ Lean Together 2025: Joseph Tooby-Smith, Physics and Lean
- 📽️ Seminar recording of "HepLean: Lean and high energy physics" by J. Tooby-Smith

Papers referencing PhysLean

- Hu, Jiewen, Thomas Zhu, and Sean Welleck. "miniCTX: Neural Theorem Proving with (Long-) Contexts." arXiv preprint arXiv:2408.03350 (2024). Project page

How PhysLean (then called HepLean) was used: *Theorems from the space-time files of HepLean were included in a data set used to evaluate the ability of models to prove theorems from real-world repositories, which requires working with definitions, theorems, and other context not seen in training.*

Contributing

We would love to have you involved! See the Get Involved page to see how you can get involved. Any contributions are welcome! If you have any questions or want permission permission to create a pull-request for this repository contact Joseph Tooby-Smith on the Lean Zulip, or email.

Installation

If you want to play around with PhysLean, but do not want to download Lean, then you can use GitPod.

Installing Lean 4

Installation instructions for Lean 4 can be found:

- https://lean-lang.org/lean4/doc/quickstart.html

or

- https://leanprover-community.github.io/get_started.html

Installing PhysLean

- Clone this repository (or download the repository as a Zip file)
- Open a terminal at the top-level in the corresponding directory.
- Run `lake exe cache get` . The command `lake` should have been installed when you installed Lean.
- Run `lake build` .
- Open the directory (not a single file) in Visual Studio Code (or another Lean compatible code editor).

Optional extras

- Lean Copilot and LLMLean allow for the use of large language models in Lean
- tryAtEachStep allows one to apply a tactic, e.g. `exact?` at each step of a lemma in a file to see if it completes the goal. This is useful for golfing proofs.

About

A project to digitalise results from physics into Lean.

physlean.com

physics | theorem-proving | lean

Readme
Apache-2.0 license
Activity
Custom properties
278 stars
6 watching
26 forks
Report repository

Releases

9 tags

Create a new release

Packages

No packages published
Publish your first package

Contributors 13

Languages

Lean 99.4% | Other 0.6%