

(Django, React) также найдут отражение в курсе.

А затем и ты на курсе реализуешь свой большой веб-проект, применив все полученные знания и навыки.

## Оглавление курса

1. Подготовка
2. Как работает компьютер
3. Программное обеспечение
4. GNU/Linux
5. Как работает Интернет и веб: основы
6. DNS
7. HTTP, HTTPs
8. HTTP2
9. Сессии, cookies, безопасность
10. Быстрое погружение во frontend
11. SQL в PostgreSQL
12. PostgreSQL на практике
13. Python для самых маленьких хардкорщиков
14. Python для подрастающих хардкорщиков
15. Pythonic код
16. Алгоритмы и структуры данных
17. Как писать хорошо, а нехорошо не писать
18. Git
19. Контейнеризация — Docker, Podman и коллеги
20. Первая практика с FastAPI
21. Redis
22. Асинхронный Python
23. Django и как его готовить

- [24. FastAPI, SQLAlchemy, Alembic](#)
- [25. Автотесты](#)
- [26. Архитектура backend-приложений](#)
- [27. 12 факторов](#)
- [28. HTML](#)
- [29. CSS или как всех достать, раскрасить и распахать](#)
- [30. Мобильные приложения на фронтенд-технологиях](#)
- [31. Деплой веб-приложения](#)
- [32. JavaScript](#)
- [33. TypeScript](#)
- [34. React и SolidJS](#)
- [35. Архитектура frontend-приложения](#)
- [36. Взлом и защита](#)
- [37. CI/CD](#)
- [38. Облачные технологии](#)
- [39. Советы по работе в ИТ](#)
- [40. Rroom](#)
- [41. Финальный проект](#)
- [42. Что дальше](#)

## FAQ

Когда можно начать проходить курс?	+
Готовы не все главы?	+
Сколько времени нужно уделять курсу?	+
Можно оплатить от компании?	+
Когда закроется продажа?	+
Доступ к материалам курса вечный?	+