



empenoso

18 фев в 03:25

# Автоматизируем учёт облигаций: как избавиться от рутины

👉 Простой

🕒 7 мин

👁️ 5.8K

Python\*, API\*, Open source\*, Финансы в IT

Кейс

В преддверии одной большой конференции про облигации я, Михаил Шардин и Екатерина Кутняк решили написать статью о том, почему облигации важны и как можно автоматизировать рутинные действия с ними.



Передаю слово Екатерине:

Сейчас, наверно, уже самые начинающие инвесторы понимают насколько важны облигации в портфеле. Ведь они выполняют целый ряд функций, помимо зарабатывания денег.

Улучшают ликвидность, оптимизирую показатель риск/доходность, придают гибкость при принятии решений. А на сегодняшний день они ещё и самый доходный и предсказуемый инструмент.

Мало кто из частных инвесторов заморачивается ведением своего портфеля. Конечно, это неправильно, но вполне можно понять. Слишком много времени уходит на учет, а его всегда и всем не хватает. Поэтому и доверяют своему торговому приложению. Но не видя картинки в целом, очень трудно принимать как стратегические, так и тактические решения по портфелю.

Сейчас есть приложения, которые помогают вести портфель. Они не бесплатные, но порядок в портфеле того стоит.

Это статья скорее для тех, кто понимает важность учета и пытается сам вести свой портфель в Excel. У меня было четыре статьи о том, как вести учет облигаций в Excel. Их

РЕКЛАМА

**До 1 000 000 бонусов**

на перенос IT-инфраструктуры



Как вести учет облигаций в Excel

[smart-lab.ru](https://smart-lab.ru)



Как вести учет облигаций в Excel. Расчет доходности

[smart-lab.ru](https://smart-lab.ru)



Как вести учет облигаций в Excel. Часть 3. Дюрация

[smart-lab.ru](https://smart-lab.ru)



Как рассчитать дюрацию «на коленке». Часть 4

[smart-lab.ru](https://smart-lab.ru)

Но всегда есть способ что-то улучшить и оптимизировать. Например сервисы по учету бумаг, [такие как Snowball или другие](#), предлагают помимо прочих опций еще и график купонных выплат. Вещь, безусловно очень важная. Почему?

Чтобы максимально эффективно управлять облигационной частью портфеля, важно учитывать и анализировать все выплаты.

Почему учёт купонов так важен?

- ✔ Позволяет прогнозировать денежные потоки.
- ✔ Упрощает реинвестирование.
- ✔ Помогает оценить реальную доходность портфеля.

**Например, брокерские приложения принципиально не учитывают выплаченные купоны, что вносит дикую путаницу в умах инвесторов. Они бесконечно жалуются на “красные” портфели, и сетуют на убытки. Забывая о том, что облигации - это еще и полученные купоны.**



**До 1 000 000 бонусов**  
на перенос IT-инфраструктуры

того, о чем мы расскажем в этой статье.

### **Итак, у нас есть проблема:**

Как учитывать купонные выплаты, если в портфеле несколько облигаций с разными датами выплат? Вручную отслеживать их сложно, особенно если бумаги погашаются и покупаются новые.

Обычные способы учёта:

- Вести записи вручную в Excel.
- Использовать специализированные приложения (платно)

### **Ручной способ**

В Excel трудоёмко, легко что-то упустить. От ручного вбивания каждой выплаты, а перешла к сводным таблицам. Это гораздо нагляднее. В результате, получается отлично отслеживать не только купоны, но и погашения, и оферты.

Можно смотреть выплаты по месяцам



**До 1 000 000 бонусов**  
на перенос IT-инфраструктуры

А можно раскрыть таблицу, и смотреть по дням в конкретном месяце:



**До 1 000 000 бонусов**  
на перенос IT-инфраструктуры

Можно добавить наименование выпусков, и тогда картинка получается максимально информативная:



**До 1 000 000 бонусов**  
на перенос IT-инфраструктуры

Безусловно, так гораздо нагляднее и удобнее. Но остается одна проблема, чтобы сделать сводные таблицы все еще нужна исходная таблица, содержащая всю необходимую информацию. Вот ее то по-прежнему приходилось делать руками.

Но и у этой проблемы есть техническое решение.

**Передаём слово Михаилу:**

## **Автоматизированное получение данных о купонах и выплатах номинала облигаций**

Разработал Python скрипт, который автоматически скачивает данные о купонах и выплатах номинала через API Московской биржи для списка облигаций из первой вкладки Excel-файла и записывает результат обратно в этот же файл на вкладку "Ден.поток". Третья вкладка содержит формулы анализа полученных данных.



**До 1 000 000 бонусов**  
на перенос IT-инфраструктуры

Расчёт для портфеля на 5 млн рублей

Я выбрал использовать Python скрипт отдельно от встроенной функции Python в Excel, потому что встроенная функция накладывает множество ограничений. Например, у меня был Microsoft Office 2021 - в нём функции Python нет. Пришлось установить Microsoft Office 2024 - только в нём есть встроенный Python в Excel. Но и этого оказалось недостаточно чтобы запускать встроенный Python в Excel. Необходимо иметь Office 365 (Mondo 2016) подписку, чтобы функция активировалась. И будет работать только под Windows. Да и сам



**До 1 000 000 бонусов**  
на перенос IT-инфраструктуры

Исходя из этого я выбрал написать отдельный Python скрипт - ведь тогда уже не важно какая именно версия Эксель у вас установлена. И даже не важно какая операционная система - Windows или macOS. Да даже если и Linux - Python поддерживает множество операционных систем, а свободный аналог Экселя под Linux называется LibreOffice Calc.

### Как это помогает?

- Видно, когда и сколько денег поступит.
- Можно заранее планировать реинвестирование.
- Повышается контроль над кэшем в портфеле.

Все это касается уже существующих портфелей.

Но есть еще один неочевидный плюс для тех, кто прежде чем купить облигационный портфель, планирует его структуру. Можно не прилагая больших усилий найти оптимальное соотношение бумаг, чтобы получить планируемый поток кэша. Это очень удобно именно на этапе выбора структуры.

### Как это выглядит на практике?



**До 1 000 000 бонусов**

на перенос IT-инфраструктуры

через отправку pull request (PR) на GitHub.

Для того чтобы Вы могли воспользоваться скриптом - сначала скачайте его на свой компьютер с Гитхаба по кнопке < > Code / Download ZIP.

Весь скрипт находится в файле « `2_bonds_search_coupons.py` ». Вы можете познакомиться с его содержимым перед запуском - для этого откройте его, например, блокнотом.

```
# 💰 Скачивание и обработка данных о денежном потоке облигаций 💰  
#  
# Этот Python скрипт автоматически скачивает данные о купонах и выплатах номинала  
# через API Московской биржи для списка облигаций из Excel-файла bonds.xlsx и  
# записывает результат обратно в этот же файл.  
#  
# Установка зависимостей перед использованием: pip install requests openpyxl  
#  
# Автор: Михаил Шардин https://shardin.name/  
# Дата создания: 29.01.2025  
# Версия: 1.1  
#  
# Актуальная версия скрипта всегда здесь: https://github.com/empenoso/moex-bond-search-  
#
```



**До 1 000 000 бонусов**  
на перенос IT-инфраструктуры

```
from datetime import datetime

# Настройка кодировки для корректного вывода русского текста
import sys
sys.stdout.reconfigure(encoding='utf-8')

# Загружаем Excel-файл
file_path = "bonds.xlsx"
wb = openpyxl.load_workbook(file_path)
sheet_data = wb["Исходные данные"]
sheet_result = wb["Ден.поток"]

# Очищаем лист с результатами
sheet_result.delete_rows(1, sheet_result.max_row)
sheet_result.append(["Название", "Идентификатор", "Дата выплаты", "Денежный поток, Р (к

# Считываем данные из листа "Исходные данные"
ArraySymbolQuantity = []
for row in sheet_data.iter_rows(min_row=2, max_row=sheet_data.max_row, values_only=True):
    if row[0] and row[1]: # Проверяем, что данные не пустые
        ArraySymbolQuantity.append(row)

print(f"Считано {len(ArraySymbolQuantity)} облигаций для обработки.")

CashFlow = []

# Обрабатываем каждую облигацию
for ID, number in ArraySymbolQuantity:
    print(f"\nОбрабатываем {ID}, количество: {number} шт.")
    url = f"https://iss.moex.com/iss/statistics/engines/stock/markets/bonds/bondization
    print(f"Запрос к {url}")

    response = requests.get(url)
    json_data = response.json()

# Обработка купонов
for coupon in json_data.get("coupons", {}).get("data", []):
    name = coupon[1].replace("'", '').replace('"', '').replace("\\", '')
    isin = coupon[0]
    coupon_date = coupon[3]
```



**До 1 000 000 бонусов**  
на перенос IT-инфраструктуры

```

    if coupon_datetime > datetime.now():
        value_rub = (coupon[9] or 0) * number
        CashFlow.append([f"{name} (купон 💎)", isin, coupon_datetime, value_rub])
        print(f"Добавлен купон: {CashFlow[-1]}")

# Обработка выплат номинала
for amort in json_data.get("amortizations", {}).get("data", []):
    name = amort[1].replace("'", '').replace('"', '').replace("\\", '')
    isin = amort[0]
    amort_date = amort[3]

    # Преобразуем дату в объект datetime
    amort_datetime = datetime.strptime(amort_date, "%Y-%m-%d")

    if amort_datetime > datetime.now():
        value_rub = (amort[9] or 0) * number
        CashFlow.append([f"{name} (номинал 💰)", isin, amort_datetime, value_rub])
        print(f"Добавлена выплата номинала: {CashFlow[-1]}")

# Записываем данные в Excel
for row in CashFlow:
    sheet_result.append(row)

# Устанавливаем формат ячеек
for cell in sheet_result["C"][1:]: # Пропускаем заголовок
    cell.number_format = "DD.MM.YYYY"

for cell in sheet_result["D"][1:]:
    cell.number_format = '# ##0,00 P'

# Добавляем запись об обновлении
update_message = f"\nДанные автоматически обновлены {datetime.now().strftime('%d.%m.%Y')}
sheet_result.append(["", update_message])
print(update_message)

# Сохраняем изменения в файле
wb.save(file_path)
print(f"Файл {file_path} успешно обновлён.")

```



**До 1 000 000 бонусов**  
на перенос IT-инфраструктуры

Хотя нумерация скрипта начинается с двойки, это полностью самостоятельный скрипт, который ищет и скачивает данные о купонах и выплатах номинала через API Московской биржи.

Теперь чтобы запустить скрипт сначала нужно установить Python. Скачайте и установите последнюю версию [с официального сайта Python](#) для Вашей операционной системы.

Затем, найдите в меню "Пуск" "Командную строку" или "Терминал" и откройте её. В командной строке введите:

```
pip install requests openpyxl
```

Это установит необходимые для работы скрипта инструменты - без их установки скрипт работать не будет:

- Requests: библиотека для отправки HTTP-запросов и работы с ответами от серверов.
- OpenPyXL: библиотека для чтения и записи данных в файлы Excel (.xlsx).



**До 1 000 000 бонусов**  
на перенос IT-инфраструктуры

Заполните файл-шаблон « `bonds.xlsx` » Вашими собственными данными.  
Переименовывать файл и менять название вкладок нельзя - в скрипте жестко заданы названия файла и вкладки. Убедитесь, что Excel-файл « `bonds.xlsx` » закрыт перед запуском.



**До 1 000 000 бонусов**  
на перенос IT-инфраструктуры



**До 1 000 000 бонусов**  
на перенос IT-инфраструктуры

Теперь, найдите « `2_bonds_search_coupons.py` » в проводнике, дважды кликните по нему. Это запустит скрипт, и он обновит файл Excel. После окончания работы можно посмотреть логи работы - скрипт ждёт нажатия кнопки.

Все эти данные, полученные через API Мосбиржи доступны только вам и для ознакомления.



**До 1 000 000 бонусов**  
на перенос IT-инфраструктуры

Скрипт автоматически заполняет вкладку с денежным потоком, а все сводные данные в человеческом понятном виде находится на вкладке аналитика. Надеюсь этот скрипт облегчит Вашу работу!

## Итоги: почему стоит учитывать облигации и как это сделать эффективно?

- ◆ Без учёта облигаций инвестор упускает важную часть доходности.
- ◆ Системный учёт купонных выплат даёт чёткое понимание будущих денежных потоков.
- ◆ Автоматизация позволяет не тратить время на рутину.



**До 1 000 000 бонусов**  
на перенос IT-инфраструктуры

Авторы статьи: [Михаил Шардин](#) и [Екатерина Кутняк](#)

Код и техническая реализация: [Михаил Шардин](#)

---

P.S. Я планирую написать и опубликовать ещё три продолжения.

P.P.S. Небольшой анонс для публикации следующей недели: для меня, как для частного инвестора многие варианты облигаций с большими процентами окажутся мусорными. И у меня, как частного инвестора, много других проблем и забот - и мне не всегда есть время разбираться чем занят тот или иной эмитент. Нет времени знакомиться с эмитентом или проводить какую-то аналитику.

Публикация следующей недели будет посвящена скрипту, который по коду ценной бумаги с Московской биржи внутри файла `bonds.xlsx` автоматически собирает все последние новости об этой компании.

**Теги:** [моех](#), [облигации](#), [сезон open source](#)

**Хэбы:** [Python](#), [API](#), [Open source](#), [Финансы в IT](#)

## Редакторский дайджест

Присылаем лучшие статьи раз в месяц

**189**

Карма

**17.5**

Рейтинг

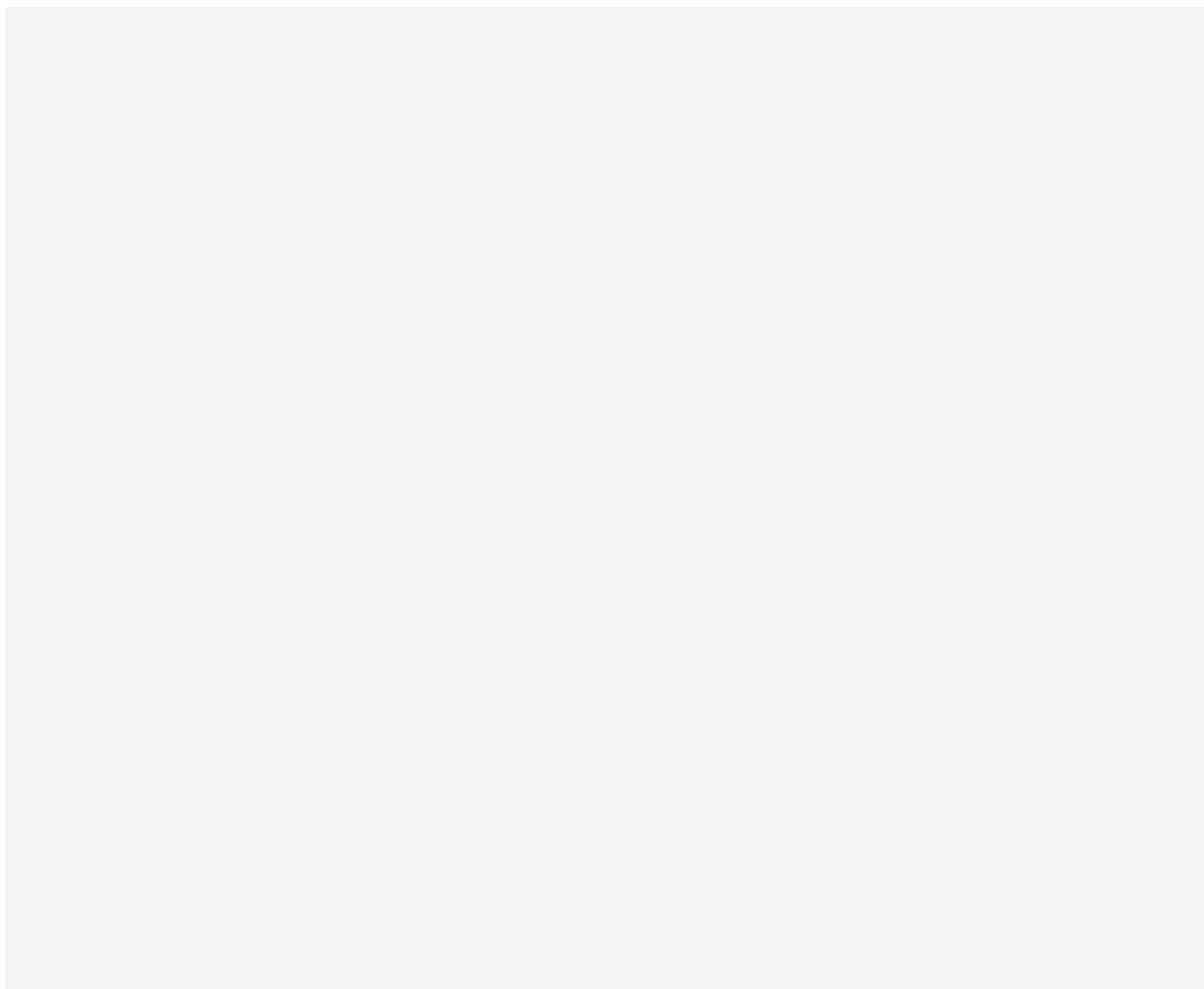
**Михаил Шардин** [@empenoso](#)

[Автоматизация](#) / [Данные](#) / [Финансы](#) / [Умные дома](#)

**До 1 000 000 бонусов**

на перенос IT-инфраструктуры

[Хабр Карьера](#) [Сайт](#) [Сайт](#) [Github](#)



 [Комментарии 3](#)

## Публикации

**ЛУЧШИЕ ЗА СУТКИ**

[ПОХОЖИЕ](#)



**Exosphere**

15 часов назад



**До 1 000 000 бонусов**  
на перенос IT-инфраструктуры

 +93 34 64

vital\_pavlenko

22 часа назад

## Больше нет входа в IT. Только выход

 2 мин  76K +49 78 321

duran-duran

21 час назад

## Трамплин в интернет: как мы ускорили запуск Яндекс Браузера

 6 мин  3.2K +42 9 29

Nickmob

16 часов назад

## Введение в Angie: краткая история и отличия от Nginx

 Простой  7 мин  2.8K

Ретроспектива

 +37 29 8

Myskat\_90

21 час назад

## Распределённый инференс и шардирование LLM. Часть 1: настройка GPU, проброс в Proxmox и настройка Kubernetes

 Сложный  14 мин  1.8K

Тutorial

 +33 46 0**До 1 000 000 бонусов**

на перенос IT-инфраструктуры

## Мое производство электрощитов приносит 40 млн в год. Спасибо нейросетям и СССР за конструкторскую школу

 Простой  14 мин  21K

Интервью

 +29

 45

 53



AlexeyNadezhin

22 часа назад

## Важное обновление BatteryTest 2

 Простой  3 мин  2.5K

 +27

 37

 9



ru\_vds

16 часов назад

## Как создавались вокальные эффекты Daft Punk

 Средний  13 мин  836

Обзор

Перевод

 +23

 5

 1



GordienkoAnd

19 часов назад

## Как настраивать сети: готовые решения Selectel для максимальной отказоустойчивости

 Средний  20 мин  2.5K

Обзор

 +21

 28

 0



alizar

20 часов назад



До 1 000 000 бонусов  
на перенос IT-инфраструктуры

Обзор

+21

7

5

## Превращаем бездушные тикеты в эпическое приключение

Промо

Показать еще

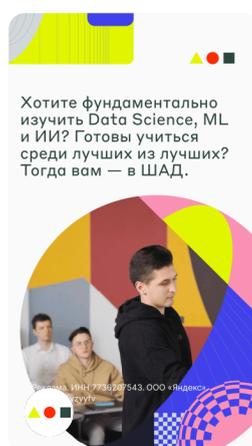
### ИСТОРИИ



**Старт гонки разработчиков**



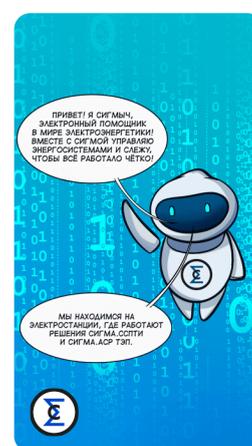
**Торопись в сезон Open source**



**Открыт приём в Школу анализа данных**



**С Днём радио!**



**HELLO, WORLD — теперь на 110 киловольтах**



**Будущее**

### ВАКАНСИИ

**Ведущий разработчик C (Open source)**

от 300 000 Р · СберТех · Москва · Можно удаленно

**Python/Django разработчик**

от 800 до 2 000 \$ · Floship · Можно удаленно

**Python-разработчик**

от 2 000 до 3 500 \$ · BCraft · Можно удаленно



**До 1 000 000 бонусов**

на перенос IT-инфраструктуры

## Python разработчик Senior

от 200 000 до 300 000 Р · Туроператор «Русь» · Москва

[Больше вакансий на Хабр Карьере](#)

### МИНУТОЧКУ ВНИМАНИЯ



Экономим деньги со скидками в Промокодусе



Как избежать ошибок при бизнес-миграции и сохранить данные?



Как упорядочить хаос тикетов на спринте

### РАБОТА

Data Scientist

43 вакансии

Python разработчик

62 вакансии

Django разработчик

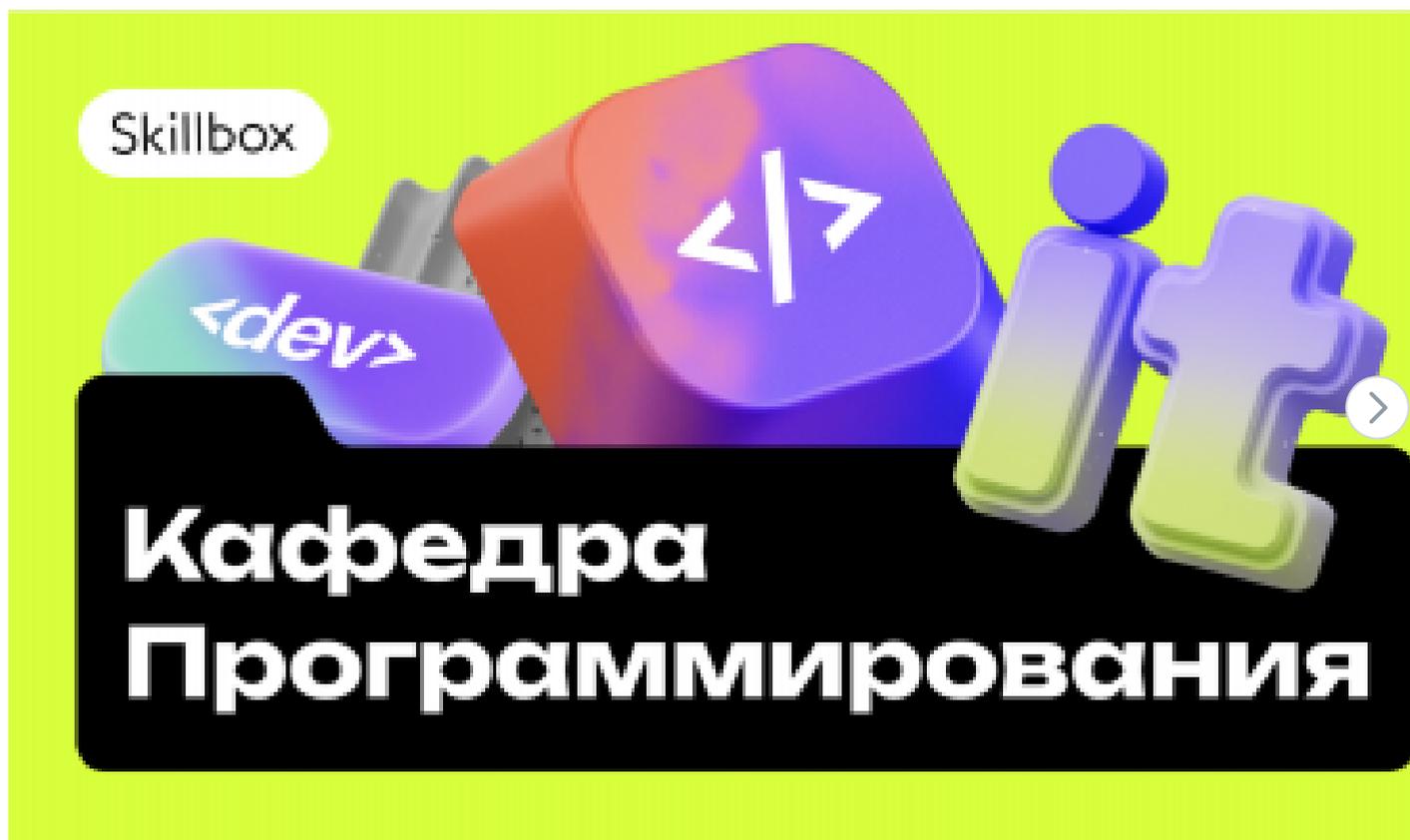
17 вакансий

[Все вакансии](#)

### БЛИЖАЙШИЕ СОБЫТИЯ



**До 1 000 000 бонусов**  
на перенос IT-инфраструктуры



17 апреля – 29 мая

## Серия бесплатных офлайн-конференций «Кафедра Программирования» от Skillbox

Москва

Разработка

Больше событий в календаре

Хабр



До 1 000 000 бонусов  
на перенос IT-инфраструктуры

техническая поддержка

© 2006–2025, Habr



**До 1 000 000 бонусов**  
на перенос IT-инфраструктуры