

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ

Зачем изучать программирование?	14
Почему Python?	15
Почему Minecraft?	15
Что вы найдете в этой книге?	15
Интернет-ресурсы	17
Приключение начинается!	18

1. ГОТОВИМСЯ К ПРИКЛЮЧЕНИЯМ

Установка и настройка программ для Windows	20
Установка Minecraft	20
Установка Python	21
Установка Java	23
Установка Minecraft Python API и Spigot	25
Запуск Spigot и создание профиля игры	26
Создание нового мира	29
Игра без доступа к интернету	31
Переключение в режим выживания	31
Установка и настройка программ для Mac OS	33
Установка Minecraft	33
Установка Python	35
Установка Java	36

Установка Minecraft Python API и Spigot	37
Запуск Spigot и создание профиля игры	39
Выбор подходящей версии Minecraft	40
Создание мира	41
Создание нового мира	42
Игра без доступа к интернету	43
Переключение в режим выживания	43
Установка и настройка программ для Raspberry Pi	44
Знакомство с IDLE	45
Знакомство с окном консоли Python	46
Поприветствуйте окно программы	47
Когда нужно использовать окно консоли, а когда окно программы	49
Подсказки	50
Проверяем работу Minecraft и Python	51

2. ТЕЛЕПОРТАЦИЯ С ПОМОЩЬЮ ПЕРЕМЕННЫХ

Что такое программа?	53
Хранение информации в переменных	54
Как устроены языки программирования	55
Синтаксис для переменных	56
Изменение значений переменных	57
Целые числа	57
<i>Миссия 1. Телепортация игрока</i>	58
Вещественные числа	64
<i>Миссия 2. Перемещение в точности туда, куда надо</i>	65
Замедление телепортации с помощью модуля time	67
<i>Миссия 3. Телепортационный тур</i>	68
Отладка	70
<i>Миссия 4. Исправьте неработающий телепортатор</i>	71
Что вы узнали	73

3. МАТЕМАТИКА, МОМЕНТАЛЬНОЕ СТРОИТЕЛЬСТВО И СУПЕРПРЫЖКИ

Выражения и команды	74
Операции	75
Сложение	75
<i>Миссия 5. Башенка из блоков</i>	76
<i>Миссия 6. Суперпрыжок</i>	78
Вычитание	79

Миссия 7. Измените блок под ногами игрока	80
Математические операции и аргументы	81
Миссия 8. Быстрое строительство	83
Умножение	86
Деление	86
Миссия 9. Потрясающие шпиды	87
Возведение в степень	89
Скобки и порядок выполнения операций	89
Полезные математические хитрости	90
Сокращенные операции	90
Случайные числа	91
Миссия 10. Суперпрыжок в неизвестность	92
Что вы узнали	93

4. ОБЩАЕМСЯ С ПОМОЩЬЮ СТРОК

Что такое строки?	95
Функция print()	95
Миссия 11. Привет, мир Minecraft!	96
Функция input()	97
Миссия 12. Отправьте в чат сообщение	99
Склейка строк	100
Преобразование числа в строку	101
Склейка целых и вещественных чисел	102
Миссия 13. Добавьте перед сообщениями имена	102
Преобразование строки в целое число	104
Миссия 14. Позвольте пользователю выбрать тип блока	104
Обработка исключений	106
Миссия 15. Допускаются только числа	107
Миссия 16. Отчет о перемещениях	109
Что вы узнали	112

5. «ИСТИНА» И «ЛОЖЬ» БУЛЕВЫХ ЗНАЧЕНИЙ

Булевы значения: основы	114
Миссия 17. Отставить разрушение блоков!	114
Склейка строк и булевых значений	115
Операции сравнения	116
«Равно»	116
Миссия 18. Игрок в воде?	117
«Не равно»	119
Миссия 19. Игрок в воздухе?	119
«Больше» и «меньше»	121

«Больше или равно» и «меньше или равно»	122
Миссия 20. Игрок над землей?	123
Миссия 21. Далеко ли игрок от дома?	124
Логические операции	126
Логическое «и»	126
Миссия 22. Игрок под водой?	127
Логическое «или»	129
Миссия 23. Игрок на дереве?	129
Логическое «не»	131
Миссия 24. Это не арбуз?	131
Порядок выполнения логических операций	133
Мое число между двумя другими?	134
Миссия 25. Игрок в доме?	135
Что вы узнали	136

6. КОНСТРУКЦИЯ IF, ДУШ И ПОТАЙНАЯ ДВЕРЬ

Конструкция if	138
Миссия 26. Как сделать кратер	139
Конструкция else	141
Миссия 27. Предотвратить разрушения или нет?	142
Конструкция elif	144
Миссия 28. Подарок	146
Цепочки конструкций elif	148
Миссия 29. Телепортация в нужное место	149
Вложенные конструкции if	151
Миссия 30. Потайная дверь	151
Проверка диапазона значений с помощью if	153
Миссия 31. Ограничьте область телепортации	154
Логические операции и конструкция if	156
Миссия 32. Душ	158
Что вы узнали	160

7. ЦИКЛ WHILE, ДИСКОТЕКА И ЦВЕТОЧНЫЙ ДОЖДЬ

Простейший цикл while	161
Миссия 33. Телепортация в случайные места	163
Управление циклами с помощью переменной count	165
Миссия 34. Водяное проклятие	167
Бесконечный цикл while	169
Миссия 35. Цветочный след	169
Замысловатые условия	171
Миссия 36. Состязание ныряльщиков	171

Логические операции и цикл while	174
Проверка диапазона значений в условии while	174
<i>Миссия 37. Постройте танцпол</i>	175
Вложенные конструкции if и циклы while	177
<i>Миссия 38. Прикосновение Мидаса</i>	178
Выход из цикла while с помощью break	179
<i>Миссия 39. Постоянный чат на основе цикла</i>	180
Конструкция while-else	181
<i>Миссия 40. «Горячо или холодно»</i>	182
Что вы узнали	185

8. ФУНКЦИИ КАК ИСТОЧНИК БОЛЬШИХ ВОЗМОЖНОСТЕЙ

Создание собственных функций	187
Вызов функции	187
Функции принимают аргументы	188
<i>Миссия 41. Посадите лес</i>	190
Рефакторинг кода	191
<i>Миссия 42. Да здравствует рефакторинг!</i>	192
Комментирование с помощью строк документации	194
Переносы строк в списке аргументов	195
Возвращаемое значение функции	195
<i>Миссия 43. Напоминалка типов блоков</i>	197
If и while внутри функций	199
Конструкция if	200
<i>Миссия 44. Цвет шерсти</i>	201
Цикл while	202
<i>Миссия 45. Блоки повсюду</i>	204
Глобальные и локальные переменные	205
<i>Миссия 46. Самодвижущийся блок</i>	207
Что вы узнали	209

9. СПИСКИ, СЛОВАРИ И УДАРЫ ПО БЛОКАМ

Работа со списками	210
Доступ к элементам списка	211
Изменение элементов списка	212
<i>Миссия 47. Высоко и низко</i>	213
Изменение структуры списка	215
Добавление элемента	215
Вставка элемента	215
Удаление элемента	216
<i>Миссия 48. Столбик-секундомер</i>	217

Работа со строками как со списками	219
Кортежи	220
Присвоение значений переменным с помощью кортежей	220
<i>Миссия 49. Скольжение</i>	221
Функции, возвращающие кортеж	223
Другие полезные свойства списков	224
Длина списка	224
<i>Миссия 50. Удары по блокам</i>	225
Выбор случайного элемента	228
<i>Миссия 51. Случайный блок</i>	228
Копирование списков	229
Проверка элементов и конструкция if	231
<i>Миссия 52. Меч ночного видения</i>	232
Словари	234
Создание словаря	234
Доступ к элементам словаря	235
<i>Миссия 53. Путеводитель</i>	236
Изменение и добавление элементов словаря	237
Удаление элементов словаря	239
<i>Миссия 54. Удары по блокам и таблица результатов</i>	239
Что вы узнали	241

10. ЦИКЛЫ FOR И ВОЛШЕБСТВО В MINECRAFT

Простой цикл for	242
<i>Миссия 55. Волшебная палочка</i>	243
Функция range()	245
<i>Миссия 56. Волшебная лестница</i>	246
Эксперименты с функцией range()	247
Другие функции для работы со списками	248
<i>Миссия 57. Колонны</i>	250
<i>Миссия 58. Пирамида</i>	251
Перебор элементов словаря в цикле	252
<i>Миссия 59. Таблица результатов</i>	254
Конструкция else и цикл for	255
Выход из цикла for-else с помощью break	255
<i>Миссия 60. Алмазоискатель</i>	256
Вложенные циклы for и многомерные списки	257
Думаем в двух измерениях	257
Доступ к элементам 2D-списка	262
<i>Миссия 61. Пиксель-арт</i>	263

Генерация 2D-списка с помощью циклов	265
<i>Миссия 62. Обветшалая стена</i>	266
Думаем в трех измерениях	268
Отображение 3D-списков	269
Доступ к элементам 3D-списка	275
<i>Миссия 63. Копирование конструкций</i>	276
Что вы узнали	282

11. КОПИРОВАНИЕ КОНСТРУКЦИЙ С ПОМОЩЬЮ ФАЙЛОВ И МОДУЛЕЙ

Работа с файлами	284
Открытие файла	284
Запись данных и сохранение файла	285
Чтение данных из файла	286
Чтение строки из файла	287
<i>Миссия 64. Перечень дел</i>	288
Часть 1: сохранение записей	288
Часть 2: вывод перечня дел на экран	290
Модули	291
Модуль pickle	291
Импортирование pickle	291
Импортирование одиночной функции с помощью from	293
Импортирование всех функций с помощью *	294
Псевдоним модуля	294
<i>Миссия 65. Сохранение и загрузка конструкции</i>	295
Часть 1: сохранение конструкции	295
Часть 2: загрузка конструкции	299
Модуль shelve и хранение наборов данных	301
Открытие файла с помощью shelve	301
Добавление, изменение и доступ к данным файла при помощи shelve	301
<i>Миссия 66: сохранение набора конструкций</i>	302
Часть 1: запись конструкции в коллекцию	303
Часть 2: загрузка конструкции из коллекции	303
Установка новых модулей с помощью pip	307
Работа с pip в Windows	308
Работа с pip в Mac OS и Raspberry Pi	309
Модуль для создания веб-сайтов Flask	309
<i>Миссия 67. сайт с координатами игрока</i>	311
Что вы узнали	311

12. ОБЪЕКТНО ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ – ЭТО КЛАССНО!

Основы	314
Создание класса	314
Создание объектов	315
Доступ к свойствам	316
<i>Миссия 68. Объекты-места</i>	316
Что такое методы	318
<i>Миссия 69. Дом-призрак</i>	320
Методы, возвращающие значение	323
<i>Миссия 70. Замок-призрак</i>	324
Создание нескольких объектов	326
<i>Миссия 71. Поселок-призрак</i>	327
Свойства класса	328
Наследование	331
Наследование классов	332
Добавление новых методов в производный класс	333
<i>Миссия 72. Гостиница-призрак</i>	334
Переопределение методов и свойств	336
<i>Миссия 73. Дерево-призрак</i>	339
Что вы узнали	341
ПОСЛЕСЛОВИЕ	342
УСТРАНЕНИЕ НЕПОЛАДОК	343
ИДЕНТИФИКАТОРЫ БЛОКОВ	350
ПРЕДМЕТНЫЙ УКАЗАТЕЛЬ	356
ОБ АВТОРЕ И ТЕХНИЧЕСКОМ РЕДАКТОРЕ	363
БЛАГОДАРНОСТИ	364
РЕСУРСЫ	365

*Посвящается всем взрослым и детям, которые читали
ранние черновики этой книги, пользовались моими советами
и инструкциями и посещали мои семинары. Огромное спасибо
за ваш энтузиазм и поддержку!
Эта книга — для вас*

ВВЕДЕНИЕ

Эта книга научит вас писать программы на языке программирования Python и управлять с их помощью событиями в мире Minecraft. Освоив основы программирования, вы тут же пустите новые знания в ход: соорудите постройки, создадите мини-игры и превратите обычные предметы в потрясающие артефакты. Навыков, которые вы получите, будет достаточно для воплощения в жизнь самых смелых идей! И не только в мире Minecraft. Если захотите, сможете писать на Python игры, приложения и полезные утилиты. Итак, сейчас вы делаете первый шаг на пути к тому, чтобы стать выдающимся программистом и повелителем мира Minecraft!

Зачем изучать программирование?

Одна из главных причин, по которым люди решают освоить программирование, — тренировка ума для решения сложных задач. Вы научитесь разбивать большие задачи на маленькие, с которыми проще иметь дело, и подключать при этом творческое мышление.

Еще один плюс программирования в том, что оно учит мыслить логически. Логика нужна, чтобы тщательно спланировать структуру программы и сценарий ее работы. Навыки решения задач, творческое и логическое мышление пригодятся вам в самых разных областях жизни, а не только при написании кодов.

Профессия программиста — отличная перспектива: каждый день вы будете решать интересные задачи, требующие нестандартного подхода. А если вы предпочтете другую профессию, программирование

может стать необыкновенно увлекательным хобби. Кстати, я начал писать программы в свободное от работы время, а в итоге стал профессиональным программистом.

И самое главное: программирование может доставить уйму радости! Мало что сравнится с удовольствием, которое испытываешь, глядя, как созданная тобою программа делает нечто прекрасное!

Почему Python?

Все ясно, но зачем вам изучать программирование именно на языке Python? Начнем с того, что этот язык отлично подходит для новичков. Коды на Python легко писать и читать, при этом мощности языка хватает, чтобы создавать на нем серьезные компьютерные программы. Неспроста Python является одним из самых распространенных языков программирования в мире!

Почему Minecraft?

Игра Minecraft очень популярна, ведь она такая увлекательная! Вы можете создавать в игровом мире все, что захотите, — лишь дайте волю воображению! А управляя миром Minecraft с помощью Python, вы еще больше раскроете свой творческий потенциал — сделаете такое, что просто невозможно повторить обычными средствами игры (например, в считанные секунды возведете огромное здание).

Начать программировать бывает непросто, ведь сначала приходится изучать примеры кодов, которые не делают ничего интересного. Однако, совместив Python с Minecraft, вы сможете сразу писать интересные программы и наблюдать результат их работы прямо в игре.

Что вы найдете в этой книге?

Каждая из 12 глав посвящена определенным возможностям языка Python. Знакомясь с ними, вы будете постепенно наращивать свой багаж знаний. Главы состоят из объяснения устройства языка, рабочих примеров кода и миссий. В ходе миссий вы будете писать программы, взаимодействующие с Minecraft. Основы их кодов я уже подготовил, так что вам потребуется лишь добавить недостающие фрагменты. В результате вы обретете навык решения задач, необходимый каждому программисту.

Рассмотрим вкратце, что ждет вас в каждой из глав.

Глава 1 «Готовимся к приключениям» поможет настроить на вашем компьютере Python и Minecraft, после чего вы сможете приступить к программированию!

Глава 2 «Телепортация с помощью переменных» покажет, как мгновенно переместить игрока. Вы узнаете, что такое переменные и как они помогают хранить данные. Затем углубите полученные навыки, отправив игрока в телепортационное путешествие по игровому миру.

Глава 3 «Математика, моментальное строительство и суперпрыжки» научит использовать математические операции для получения игроком суперспособностей и мгновенной постройки зданий. Хотите создать дом за секунду? Математические операции помогут вам в этом! Хотите подпрыгнуть высоко-высоко над землей? И здесь пригодятся математические операции!

В **главе 4 «Общаемся с помощью строк»** вы узнаете о строках и о том, как с их помощью создать интерактивный чат, а также научитесь писать программы, отправляющие текстовые сообщения, которые смогут прочитать другие пользователи.

Глава 5 «“Истина” и “ложь” булевых значений» покажет, что при помощи булевых значений и логических операций ваши программы для Minecraft могут отвечать на однозначные вопросы, например такие: «Игрок под водой?», «Игрок на дереве?», «Игрок рядом с домом?»

Глава 6 «Конструкция if, душ и потайная дверь» — здесь мы перейдем на новый уровень булевой логики. Вы узнаете, как с помощью конструкции `if` создавать программы, выполняющие разные действия в зависимости от введенных данных. Хотите сделать потайную дверь, которая открывается, если поставить определенный блок в определенное место? С конструкцией `if` это возможно!

Глава 7 «Цикл while, дискотека и цветочный дождь» расскажет, как научить программу многократно выполнять код с помощью циклов. Вы сможете автоматизировать работу программ и получить фантастические результаты. Например, след из цветов, который будет тянуться за игроком, или волшебный танцпол, переливающийся всеми цветами радуги! Если показать все это друзьям, они будут поражены!

В **главе 8 «Функции как источник больших возможностей»** вы научитесь мгновенно создавать целые леса и поселки при помощи

функций. А еще узнаете, как облегчить себе жизнь, используя части кода повторно.

В главе 9 «Списки, словари и удары по блокам» вы создадите мини-игры с помощью списков. Список — важный инструмент программирования, позволяющий хранить множество значений в одном месте. Вам предстоит использовать списки, чтобы программа запоминала, по каким блокам вы ударили мечом. А если добавить в эту программу еще несколько строчек кода, получится забавная мини-игра!

Глава 10 «Циклы `for` и волшебство в `Minecraft`» покажет, как возводить постройки (например, пирамиды) с помощью циклов `for`. Используя их, вы сможете рисовать пиксельные картинки и создавать копии самых разных объектов. Представьте, что вы изваяли великолепную статую. Теперь вы можете скопировать и воссоздать ее многократно, получив целую армию статуй!

В главе 11 «Копирование построек с помощью файлов и модулей» вы научитесь создавать и изменять файлы прямо из кода программы, что позволит сохранять постройки и переносить их в другие миры `Minecraft`. Иными словами, скопировав постройку из игрового мира в файл, вы сможете перенести ее куда пожелаете. Хотите сохранить великолепный особняк, в который вложено столько сил и времени? Без проблем! Просто записываете здание в файл, а потом загружаете эти данные куда угодно.

Глава 12 «Объектно ориентированное программирование — это классно!» расскажет о продвинутых методах программирования — наследовании классов и объектов. Изучив эту главу, вы станете настоящим знатоком Python! Выполняя миссии, вы постройте здание, а затем с помощью классов, объектов и наследования будете создавать его копии и вариации (например, поселки и гостиницы) — и все это с помощью нескольких дополнительных строк кода!

«Идентификаторы блоков» — удобная справка по идентификаторам блоков `Minecraft`, которые вы можете использовать в своих программах.

Интернет-ресурсы

Все тексты программ, приведенные в этой книге, доступны по ссылке <http://mif.to/minecraft/>. Вы можете сверять с ними коды ваших программ,

если они работают некорректно или не запускаются, а также использовать эти коды как основу для создания собственных замечательных программ! На странице книги на сайте издательства также выложены установочные файлы — о процессе загрузки и проверки программного обеспечения я подробно расскажу в главе 1.

Приключение начинается!

Надеюсь, вам уже не терпится начать — так же как и мне. Я получил истинное удовольствие, работая над этой книгой и придумывая миссии, которые помогут вам научиться программировать. Хочется верить, что они вам понравятся. Итак, поехали!

1

ГОТОВИМСЯ К ПРИКЛЮЧЕНИЯМ



Прежде чем приступить к написанию интересных Python-программ для игры Minecraft, вам нужно установить и настроить на компьютере Minecraft, Python и еще несколько других программ. В этой главе я расскажу, как установить и запустить все необходимое программное обеспечение. Установка программ и настройка компьютера — самое сложное, с чем вы столкнетесь в этой книге, так что не стесняйтесь звать на помощь

родителей или других взрослых, которые разбираются в компьютерах. Не спешите и внимательно выполняйте инструкции шаг за шагом, иначе что-нибудь может пойти не так.

На следующих страницах вы найдете три блока инструкций для трех разных операционных систем. Если на вашем компьютере установлена Windows, просто читайте дальше. Если ваша операционная система Mac OS, перейдите к разделу «Установка и настройка программ для Mac OS» на с. 33. Если же вы планируете работать на микрокомпьютере Raspberry Pi, перейдите на с. 44 к разделу «Установка и настройка программ для Raspberry Pi». Если в процессе установки и настройки у вас возникнут проблемы, обратитесь к приложению на с. 343.



На странице книги по адресу <http://mif.to/minecraft/> можно найти дополнительную информацию и последние правки к этим инструкциям.

Установка и настройка программ для Windows

Чтобы управлять миром Minecraft с помощью языка Python, вам нужно установить пять программных пакетов:

- Minecraft
- Python 3
- Java
- Minecraft Python API
- Minecraft-сервер Spigot

В этом разделе я расскажу, как установить эти программы на компьютер с операционной системой Windows. Итак, начнем с Minecraft.

Установка Minecraft

Если у вас уже установлена последняя версия Minecraft, переходите к разделу «Установка Python» на с. 21. А если вы в этом не уверены, выполните пошаговые инструкции из этого раздела.

В случае если у вас еще нет Minecraft, игру можно приобрести на официальном сайте <https://minecraft.net/ru-ru/>. При необходимости попросите взрослых вам помочь. Запомните или запишите имя пользователя и пароль, которые вы укажете при покупке Minecraft, — они понадобятся позже, чтобы войти в игру.

После покупки установите игру на компьютер. Для этого выполните следующее:

1. Перейдите по ссылке <https://minecraft.net/ru-ru/download/>.
2. В разделе «Загрузить Minecraft для Windows» найдите кнопку **Загрузить** и кликните по ней, чтобы скачать файл *MinecraftInstaller.msi*. Если всплывет панель с вопросом, сохранить файл или открыть его, выберите **Сохранить (Save File)**.
3. Дождитесь окончания загрузки файла, а затем откройте его. Если появится диалоговое окно с вопросом, хотите ли вы запустить файл, выберите **Запустить (Run)**. Не волнуйтесь, этот файл безопасен для компьютера!
4. Когда откроется окно установщика Minecraft, кликните **Next**, а затем кликните **Next** еще раз. После этого кликните **Install**.

Download —
скачать

Next — далее

Install — устано-
вить

5. Возможно, программа установки спросит, действительно ли вы хотите установить Minecraft. Разумеется, хотите! Кликните **Да (Yes)**. Дождитесь окончания установки. За это время я успел съесть печенье и выпить стакан воды.
6. Когда установка завершится, нажмите **Finish**.

Finish — завершить

Ну вот, Minecraft установлен. Знаете, чем стоит заняться теперь? Разумеется, поиграть! Для этого нужно выполнить всего несколько действий:

1. Кликните по кнопке **Пуск (Start)** или нажмите на клавишу **WINDOWS** на клавиатуре, найдите Minecraft в списке программ и кликните по его иконке.
2. Minecraft запустится и, возможно, установит обновления.
3. После этого появится окно входа в игру. Введите имя пользователя и пароль, которые вы использовали при покупке Minecraft, и нажмите **Войти**.
4. Нажмите **Играть**. Перед запуском игры Minecraft загрузит еще несколько обновлений.
5. И наконец, кликните **Одиночная игра (Single Player) → Создать новый мир (Create New World)**. Введите название вашего мира и кликните **Создать новый мир (Create New World)**. После этого программа создаст мир — можете играть, сколько пожелаете.

Повеселитесь! Если прежде вы не играли в Minecraft, играйте, пока в мире Minecraft не стемнеет. Берегитесь монстров! Имейте в виду, что программировать на Python мы будем в многопользовательском режиме игры, который отличается от однопользовательского. Мы поговорим об этом подробнее в разделе «Запуск Spigot и создание профиля игры» на с. 26.

А теперь за работу! Пора установить Python. Находясь в игре, нажмите клавишу **ESC**, чтобы появился курсор. В открывшемся меню выберите **Сохранить и выйти (Save and Quit to Title) → Выйти из игры (Quit Game)** и закройте Minecraft.

Установка Python

Python — это язык программирования, который вы будете изучать с помощью этой книги. Давайте установим программное обеспечение Python прямо сейчас.

1. Перейдите по ссылке <http://www.python.org/downloads/>.
2. Кликните по кнопке **Download Python 3.5.2** (на момент написания этой книги версия 3.5.2 была последней, однако, если вы обнаружите на сайте более позднюю версию, установите ее).
3. Начнется скачивание Python. Если браузер задал вам вопрос, сохранить файл или открыть его, выберите **Сохранить файл (Save File)**.
4. Когда загрузка программы-установщика завершится, кликните по ее иконке. Если возникнет диалоговое окно с вопросом, действительно ли вы хотите запустить этот файл, кликните **Запустить (Run)**.
5. Когда установщик Python запустится, поставьте галочку в чекбоксе Add Python 3.5 to Path, как на рис. 1.1. Затем кликните по **Install Now**.

Install now — установить сейчас



Рис. 1.1. Убедитесь, что вы поставили галочку в чекбоксе Add Python 3.5 to PATH

6. Может появиться окно с вопросом, разрешить ли программе устанавливать программное обеспечение на этом компьютере. Кликните **Да (Yes)** и дождитесь окончания установки Python. За это время я успел подойти к окну, а когда вернулся за компьютер, дело было сделано.
7. Кликните **Close**. Python установлен!

Close — закрыть