

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Ливанов Дмитрий Викторович

Должность: Ректор

Дата подписания: 17.06.2025 14:07:49

Уникальный программный ключ:

c6d909c49c1d2034fa3a0156c4eaa51e7232a382

Федеральное государственное автономное образовательное учреждение
высшего образования

«Московский физико-технический институт
(национальный исследовательский университет)»

ВЫПИСКА ИЗ ПРОТОКОЛА № 12

заседания учебно-методического совета от 12 июля 2023

ПОВЕСТКА:

Рассмотрение дополнительных общеобразовательных и профессиональных программ.

Проректор по учебной работе А. А. Воронов.

СЛУШАЛИ: Директора Физтех-школы прикладной математики и информатики
А.М. Райгородского.

ПОСТАНОВИЛИ:

Рекомендовать к утверждению в установленном порядке дополнительную
общеобразовательную программу «Python-разработчик Junior».

Решение принято единогласно заочным голосованием.

Форма проведения заседания: заочная.

Председатель УМС МФТИ

А.А. Воронов

Ученый секретарь УМС МФТИ

М.В. Березникова

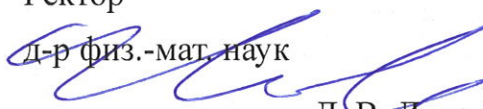
Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение высшего
образования

«Московский физико-технический институт
(национальный исследовательский университет)»

УТВЕРЖДАЮ

Ректор

д-р физ.-мат. наук



Д. В. Ливанов

«12» июля 2023 г.

Дополнительная общеобразовательная программа

«Python-разработчик Junior»

Москва 2023

Оглавление

1. Общая характеристика программы	2
2. Планируемые результаты обучения	3
3. Учебный план	3
4. Учебная программа	4
5. Организационно-педагогические условия реализации дополнительной профессиональной программы	16
5.1. Требования к материально-техническим условиям	16
5.2. Требования к информационным и учебно-методическим условиям	17
6. Оценка качества освоения программы	19
7. Составители программы:	20

1. Общая характеристика программы

- 1.1. Целью реализации дополнительной общеобразовательной программы «Python-разработчик Junior» (далее – программа) является получение навыков разработки программного обеспечения на языке Python, на технологиях, популярных в среде разработчиков Python. Будут рассмотрены основы программирования на Python, средства разработки, сборки и тестирования программ, основные алгоритмы и структуры данных, современные инструменты разработки программного обеспечения.
- 1.2. Категории слушателей, на обучение которых рассчитана программа: слушатели с отсутствующим опытом программирования и создания приложений на языке программирования Python. Участники получают основные сведения об объектно-ориентированном программировании, терминологии и синтаксисе языка, научатся создавать простые программы на языке программирования Python. Также участники программы освоят работу с базами данных при помощи языка запросов SQL, алгоритмы и структуры данных, инструменты Linux, Git, Docker, технологии и инструменты CI/CD.
- 1.3. Нормативный срок освоения программы – 545 академических часов.
- 1.4. Форма обучения: очная (с применением дистанционных образовательных технологий).
- 1.5. Режим обучения: дистанционно, 14 ак. часов в неделю.

2. Планируемые результаты обучения

В результате освоения программы слушатель должен:

владеть навыками:

- написание эффективного и безопасного кода на python;
- обработка / отладка приложений на python;
- работа с базами данных с помощью SQL;
- работа с операционной системой Linux;
- работа с системой контроля версий GIT;

знать:

- базовые понятия в программировании;
- область применения и синтаксис python;
- принципы написания эффективного и безопасного кода на python;
- средства обработки / отладки приложений на python;
- основные концепции ООП;
- основные алгоритмы и структуры данных;

уметь:

- писать эффективный и безопасный код на python;
- работать с базами данных с помощью SQL;
- работать с операционной системой Linux;
- работать с системой контроля версий GIT;
- работать с инструментом контейнеризации Docker;
- работать с инструментами CI/CD.

3. Учебный план

№ п/п*	Наименование Модулей, дисциплин	Всего, час.	В том числе:				
			Лекции	Практические занятия (семинары)	Самостоятельная работа	Контрольные задания	Форма контроля
1	Модуль 1. Основы программирования	60	12	12	30	6	проект
2	Модуль 2. Базы данных и SQL	100	10	5	70	15	проект
3	Модуль 3. Linux / Git для начинающих	20	7	6	5	2	тестирование
4	Модуль 4. Разработка на Python	200	51	24	110	15	проект

5	Модуль 5. Введение в алгоритмы и структуры данных	80	16	16	40	8	тестирование
6	Модуль 6. Docker и CI/CD для начинающих	20	2	1	10	7	тестирование
7	Итоговая квалификационная работа	65			65		проект
	ИТОГО:	545	98	64	330	53	

* Слушатели могут проходить модули в произвольном порядке.

4. Учебная программа

Учебно-тематический план программы

Модуль 1. Основы программирования

№ п/п	Содержание обучения, наименование и тематика практических занятий (вебинаров), самостоятельных работы	Объем, ак.час.
1.	Введение. Условные операторы	10
	Теория: 1. Компилируемые/интерпретируемые языка программирования. 2. Среды разработки, онлайн компиляторы, jupyter - notebook. 3. Ввод / вывод информации. 4. Типы данных и арифметические действия. 5. Переменная и ее место в памяти. Стек и куча. Имя переменной, значение переменной. 6. Условный оператор, решение задач. Блок if-elif-else. Операции сравнения, in.	2
	Практика: установка среды разработки, решение задач на условные операторы.	2
	Самостоятельная работа.	5
	Выполнение контрольных заданий.	1
2.	Вложенные условия. Циклы while / for	10
	Теория: 1. Логические операции or, and, not 2. Вложенные условия if, равносильность условию elif	2

	3. Цикл с предусловием и постусловием.	
	4. Операции break, continue.	
	5. Цикл for, зачем нужен цикл счетчик.	
	Практика: решения сложных задач, знакомство с трассировкой и отладкой, решение задач на ввод чисел вплоть до конкретного условия, решение задач на тему вложенных циклов.	2
	Самостоятельная работа.	5
	Выполнение контрольных заданий.	1
3.	Введение в коллекции. Вложенные списки	10
	Теория:	2
	1. Списки. Операции со списками и со строками. Индексация списков	
	2. Хранение данных, ввод данных.	
	3. Список списков. Хранение данных в памяти компьютера.	
	4. Срезки строк и списков.	
	Практика: решение задач на списки.	2
	Самостоятельная работа.	5
	Выполнение контрольных заданий.	1
4.	Функции	10
	Теория:	2
	1. Понятие функции и функционального программирования. Принимаемое константное число аргументов функции.	
	2. Область видимости переменных. Копирование или неkopирование элементов. Возвращаемые значения.	
	3. Вызов функции в памяти компьютера. Переменное число аргументов.	
	4. Лямбда функции.	
	Практика: решение задач на функции.	2
	Самостоятельная работа.	5
	Выполнение контрольных заданий.	1
5.	Введение в ООП	10

	Теория: 1. Понятие объектно-ориентированного программирования. Что такое класс, экземпляр и метод класса. 2. Основные принципы ООП. 3. Понятие полиморфизма. Определение и переопределение операторов. 4. Инкапсуляция. 5. Отличие Python от других языков программирования.	2
	Практика: реализация ООП в Python.	2
	Самостоятельная работа.	5
	Выполнение контрольных заданий.	1
6.	Наследование. Итоговый проект	10
	Теория: 1. Наследование. 2. Отличия наследования в Python от других языков программирования. 3. Проектирование и разработка продукта, методика. 4. Что такое Python, и почему он объектно ориентирован. В чем различие с другими языками программирования.	2
	Практика: реализация написание итогового проекта	2
	Самостоятельная работа.	5
	Выполнение контрольных заданий.	1
	Итого	60

Модуль 2. Базы данных и SQL

№ п/п	Содержание обучения, наименование и тематика практических занятий (вебинаров), самостоятельных работы	Объем, ак.час.
1.	Модуль I. Работа с базами данных с помощью SQL	80
1.1.	Основные понятия и базовый синтаксис выборки	6

1.2.	Создание таблицы и заполнение данными, ограничения на поля	5,5
1.3.	Строковые функции	6
1.4.	Изменение и удаление данных и UNION	5,5
1.5.	inner join и подзапросы	5,5
1.6.	другие типы join	5,5
1.7.	Агрегация	6,5
1.8.	Группировка	7,5
1.9.	Проектирование базы данных	5,5
1.10.	Работа со временем	5,5
1.11.	Оконные функции	5,5
1.12.	Логическое удаление	5,5
1.13.	Хранение истории	5,5
2.	Модуль II. NoSQL базы данных	20
2.1.	Знакомство с mongoDB	5,5
2.2.	Изменение данных в mongoDB	5,5
2.3.	Агрегация в mongoDB	6,5
2.4.	Соединение коллекций	7

Модуль 3. Linux / Git для начинающих

№ п/п	Содержание обучения, наименование и тематика практических занятий (вебинаров), самостоятельных работы	Объем, ак.час.
1	Базовый WSL и VirtualBox	3,5

	Теория: 1. Обзор курса. План занятий 2. Введение. Чем отличается Windows и Linux 3. Установка WSL. Запуск 4. VirtualBox установка и использование. Тест	1
	Практическая работа по теме лекции	1
	Самостоятельная работа	1
	Выполнение контрольных заданий	0,5
2	Основы Linux	6,5
	Теория: 1. Введение. Какие задачи он решает. Дистрибутив Ubuntu 2. Консоль. Основа работы с Linux. tmux 3. Скрипты. Продвинутая работа с консолью 4. Файловая система 5. Работа с сетью 6. Серверное взаимодействие. SSH	2,5
	Практическая работа по теме лекции	2,5
	Самостоятельная работа	1
	Выполнение контрольных заданий	0,5
3	WSL подробно	4
	Теория: 1. Краткий экскурс в историю появления. WSL1 и WSL 2 2. Основные команды работы с WSL 3. Файловая система 4. Сеть WSL 5. WSL USB 6. CUDA и Windows Докер	1,5
	Практическая работа по теме лекции	1
	Самостоятельная работа	1
	Выполнение контрольных заданий	0,5
4	Git	6
	Теория: 1. Введение. Цели. Разновидности систем контроля версий 2. Базовые основы GIT. Популярные сценарии работы 3. Серверные репозитории Git. Ветки. Merge request	2

	4. Сценарии отката изменений. Stash	
	Практическая работа по теме лекции	1,5
	Самостоятельная работа	2
	Выполнение контрольных заданий	0,5
	Итого	20

Модуль 4. Разработка на Python

№ п/п	Содержание обучения, наименование и тематика практических занятий (вебинаров), самостоятельных работы	Объем, ак.час.
1	Модуль 1. Базовый Python	40
1.1	Занятие 1. Введение в Python. Работа в Jupyter Notebook и Google Colab. Базовые типы данных	5
1.2	Занятие 2. Условия и циклы. Итераторы и генераторы. Контейнеры	5
1.3	Занятие 3. Функции. Области видимости. Замыкания. Декораторы	5
1.4	Занятие 4. Основы ООП. Работа с классами в Python	5
1.5	Занятие 5. Работа с файлами	5
1.6	Занятие 6. Модули, пакеты в Python. Встроенные модули. Работа со сторонними библиотеками	5
1.7	Занятие 7. Подготовка проекта	5
1.8	Занятие 8. Презентация проекта	5
2.	Модуль 2. Прикладной Python	60
2.1	Занятие 1. Введение в NumPy. SciPy	5
2.2	Занятие 2. Введение в Pandas	5

2.3	Занятие 3. Введение в визуализацию данных	5
2.4	Занятие 4. Работа с текстом в Python. Регулярные выражения	5
2.5	Занятие 5. Лабораторная работа	5
2.6	Занятие 6. Лабораторная работа	5
2.7	Занятие 7. Веб-скрапинг и парсинг	5
2.8	Занятие 8. Ускорение программ в Python. Cython. Numba	5
2.9	Занятие 9. Тестирование в Python (нормальное)	5
2.10	Занятие 10. Асинхронное программирование в Python	5
2.11	Занятие 11. Подготовка проекта	5
2.12	Занятие 12. Презентация проекта	5
3.	Модуль 3. Создание веб-приложений на Python	50
3.1	Занятие 1. Введение в сети, HTTP, HTML	5
3.2	Занятие 2. Введение в Flask	5
3.3	Занятие 3. Введение в Django. Общая структура, отличие от Flask. Обработка запросов. Шаблоны и теги	5
3.4	Занятие 4. Знакомство с базами данных. Знакомство с Django ORM	5
3.5	Занятие 5. API, REST	5
3.6	Занятие 6. Тестирование Django-приложений	5

3.7	Занятие 7. Дополнительное занятие на тему веб по темам от слушателей	5
3.8	Занятие 8. Подготовка проекта	5
3.9	Занятие 9. Подготовка проекта	5
3.10	Занятие 10. Презентация проекта	5
4.	Модуль 4. Продвинутая веб-разработка	50
4.1	Занятие 1. Асинхронная веб-разработка. Aiohttp	5
4.2	Занятие 2. Работа со статичными файлами	5
4.3	Занятие 3. Кеширование	5
4.4	Занятие 4. Event-based программирование. Django Signals	5
4.5	Занятие 5. Подготовка проекта	5
4.6	Занятие 6. Подготовка проекта	5
4.7	Занятие 7. Подготовка проекта	5
4.8	Занятие 8. Подготовка проекта	5
4.9	Занятие 9. Презентация проекта	5
4.10	Занятие 10. Презентация проекта	5
	Итого	200

Модуль 5. Введение в алгоритмы и структуры данных

№ п/п	Содержание обучения, наименование и тематика практических занятий (вебинаров), самостоятельных работы	Объем, ак.час.
1.	Введение	10
	Теория: 1. Орг. вопросы, Яндекс-контест, работа в системе, О-семантика, асимптотические классы, Мастер-теорема. 2. Обзор контейнеров в разных языках программирования, основные операции и их асимптотика. 3. Массивы, списки, двусвязные списки, стек и очередь.	2
	Практика: Реализация некоторых контейнеров. Правильные скобочные последовательности. Задача о книжках. Обратная польская нотация. Обработка запросов к серверу.	2
	Самостоятельная работа: Написать свой стек, список (двусвязанный и односвязный), очередь на стеке. Решение задач на очереди/стеки.	5
	Выполнение контрольных заданий.	1
2.	Сортировки	10
	Теория: 1. Понятие, примеры использования и свойства сортировок. 2. Квадратичные сортировки – сортировки вставками и пузырьком. 3. Merge Sort.	2
	Практика: Разбор дз. Семинар по Merge Sort / Quick Sort.	2
	Самостоятельная работа: Написать обе квадратичные сортировки, merge sort (quicksort - опционально).	5
	Выполнение контрольных заданий.	1
3.	Динамическое программирование. Бинарный (экспоненциальный) поиск.	10
	Теория: 1. Задача поиска числа в контейнере. Вычисление асимптотики. Бинарный поиск - зачем использовать. 2. Бинарный и экспоненциальный поиск.	2

	3. Динамическое программирование - что это и зачем надо. Задача о рюкзаке. (Не)точность дп.	
	4. Задача о рюкзаке с дополнительным условием. Двумерное динамическое программирование.	
	Практика: Бинарный и экспоненциальный поиск - реализация.	2
	Самостоятельная работа: Решение задач на бинарный поиск. Динамическое программирование одномерное и двумерное.	5
	Выполнение контрольных заданий.	1
4.	Хеширование. Метод цепочек.	10
	Теория:	2
	1. Хеш-функция. Ее свойства, реализация для различных типов данных. Проблема коллизии.	
	2. Хеш-таблица. Контейнеры в различных языках, использующие хеш-таблицу. Алгоритм работы хеш-таблицы.	
	3. ООП, атрибуты класса хеш-таблицы. Реализация хеш-таблицы методом закрытой адресации. Расчет ее асимптотик.	
	Практика: Ответы на вопросы, разбор домашнего задания.	2
	Самостоятельная работа: Задачи на хеш функцию. Построение своей хеш-таблицы с закрытой адресацией.	5
	Выполнение контрольных заданий.	1
5.	Хеш-таблицы с открытой адресацией	10
	Теория:	2
	1. Разрешение коллизии методом двойного хеширования.	
	2. Две хеш-функции, их свойства. Реализация алгоритма.	
	Практика: Обсуждение домашнего задания, его разбор. Решение задач по теме: хеш-таблицы / ответы на вопросы.	2
	Самостоятельная работа: Решить задачу для открытой адресации. Описать различия избежания коллизий.	5
	Выполнение контрольных заданий.	1
6.	Куча	10
	Теория:	2
	1. Определение и свойства кучи.	

	2. Вставка, Поиск максимума, Удаление. Реализация абстрактного типа данных.	
	3. Эффективное построение кучи. Взятие максимума/минимума. Какие задачи решает - в чем эффективность кучи.	
	4. Пирамидальная сортировка.	
	Практика: Построение и реализация кучи.	2
	Самостоятельная работа: Построение кучи, взятие максимума/минимума, пирамидальная сортировка.	5
	Выполнение контрольных заданий.	1
7.	Простые деревья и граф	10
	Теория: 1. Определение дерева. Высота дерева, родительские, дочерние узлы. Способы хранения дерева. 2. Обход в глубину и ширину. 3. Поиск ключа. Вставка/удаление. Проблема несбалансированности дерева. Оценка Эффективности. 4. Решаемые задачи. Поиск компонент связности, проверка на двудольность. Поиск цикла.	2
	Практика: Реализация обхода. Решение задач по теме.	2
	Самостоятельная работа: DFS, BFS, решение задач с урока.	5
	Выполнение контрольных заданий.	1
8.	Сбалансированные деревья	10
	Теория: 1. Напоминание: зачем балансировать деревья. Мотивация использовать графы. Определение AVL-дерева. 2. AVL - дерево. 3. Вращения дерева. Операции вставки и удаления. 4. Оценка эффективности AVL-дерева.	2
	Практика: Реализация конструкции класса. Реализация алгоритма AVL-дерева и ответы на вопросы.	2
	Самостоятельная работа: Реализация AVL-дерева.	5
	Выполнение контрольных заданий.	1

	Итого	80
--	--------------	-----------

Модуль 6. Docker и CI/CD для начинающих

№ п/п	Содержание обучения, наименование и тематика практических занятий (вебинаров), самостоятельных работы	Объем, ак.час.
1.	Docker	6,5
	Теория: 1. Введение 2. Что такое докер? Как его используют в организациях? Лицензирование и аналоги 3. Принципы работы докер. Докер vs VM 4. Контейнер vs образ. Dockerfile 5. Работа с контейнером. docker exec 6. Создание внешних дисков, подключение директорий 7. Создание сетей. Типы сетей 8. Docker Best Practices	1
	Практическая работа по теме лекции	0,5
	Самостоятельная работа	3
	Выполнение контрольных заданий	2
2.	Docker Compose	5,5
	Теория: 1. Что такое Docker Compose? Примеры и преимущества 2. Docker Compose. Сеть и диски в docker compose	0,5
	Практическая работа по теме лекции	0
	Самостоятельная работа	3
	Выполнение контрольных заданий	2
3.	DevOps. GitLab CI	8
	Теория: 1. Что такое DevOps? Какая связь с Agile, Scrum? Введение пайплайны в CI/CD 2. GitLab CI. Структура пайплайнов 3. CI/CD в GitLab CI 4. Подведение итогов	0,5

	Практическая работа по теме лекции	0,5
	Самостоятельная работа	4
	Выполнение контрольных заданий	3
	Итого	20

7.	Итоговая проектная работа	65
	Самостоятельная работа.	65
	Итого	545

5. Организационно-педагогические условия реализации дополнительной профессиональной программы

5.1. Требования к материально-техническим условиям

Перечень кабинетов, лабораторий, мастерских, тренажеров и др., обеспечивающих проведение всех предусмотренных программой видов занятий:

Наименование специализированных аудиторий, кабинетов, лабораторий	Вид занятий	Наименование оборудования, программного обеспечения
Система дистанционного обучения провайдера массовых открытых онлайн курсов	Лекции	Слушателю необходимо наличие доступа в сеть интернет, компьютер. Преподавателю курса необходимо наличие доступа администратора курса на LMS-платформе к материалам курса.
Информационно-коммуникационная платформа дистанционных семинаров	Практические занятия (дистанционные семинары)	Слушателю необходимо наличие доступа в сеть интернет, компьютер. Преподавателю курса необходимо оборудование для проведения дистанционных семинаров (вебинаров), качественный отказоустойчивый

		доступ в сеть интернет.
Система дистанционного обучения провайдера массовых открытых онлайн курсов	Самостоятельная работа	Наличие компьютера и доступа в сеть интернет.
Система дистанционного обучения провайдера массовых открытых онлайн курсов	Промежуточный, рубежный контроль, Итоговая аттестация	Наличие компьютера и доступа в сеть интернет.

5.2. Требования к информационным и учебно-методическим условиям

Список литературы:

МОДУЛЬ 1:

Основная литература

1. Никитина, Т. П. Программирование. Основы Python для инженеров / Т. П. Никитина, Л. В. Королёв. — Санкт-Петербург : Лань, 2023. — 156 с. — ISBN 978-5-507-45284-2. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/302720>. — Режим доступа: для авториз. пользователей.

Дополнительная литература

1. Задачи по программированию. Под ред. С. М. Окулова, - М.: БИНОМ. Лаборатория знаний, 2016.
2. Томас Х. Кормен, Чарльз И. Лейзерсон, Рональд Л. Ривест, Клиффорд Штайн. Алгоритмы: построение и анализ, 3-е издание. — М.: Вильямс, 2013.
3. Соколов А.П. Системы программирования: теория, методы, алгоритмы. - М.:ФиС, 2014.
4. С. М. Окулов. Основы программирования. - М.: Бином. Лаборатория знаний, 2012.
5. Совершенный код. Мастер-класс/ Пер. с англ. — М. : Издательство «Русская редакция», 2010. — 896 стр.

МОДУЛЬ 2:

Основная литература

1. Волк, В. К. Базы данных. Проектирование, программирование, управление и администрирование : учебник для вузов / В. К. Волк. — 3-е изд., стер. — Санкт-Петербург : Лань, 2022. — 244 с. — ISBN 978-5-8114-9368-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/193373> (дата обращения: 29.03.2023). — Режим доступа: для авториз. пользователей.
2. Нестеров, С. А. Интеллектуальный анализ данных с использованием SQL Server / С. А. Нестеров. — Санкт-Петербург : Лань, 2023. — 160 с. — ISBN 978-5-507-45535-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/311861> (дата обращения: 29.03.2023). — Режим доступа: для авториз. пользователей.

Дополнительная литература

1. Bob Ward. SQL Server 2019 Revealed: Including Big Data Clusters and Machine Learning.
2. Грофф, Джеймс . SQL : Энциклопедия : пер. с англ. / Д. Р. Грофф, П. Н. Вайнберг. — 3-е изд. — СПб. : Питер. 2016
3. Дэви Силен, Арно Мейсман, Мохамед Али. Основа Data Science и BigData. Python и наука о данных. Питер. 2017

МОДУЛЬ 3:

Основная литература

1. Орещенков, И. С. Операционные системы. Bodhi Linux 6.0: установка, настройка, эксплуатация / И. С. Орещенков. — Санкт-Петербург : Лань, 2023. — 364 с. — ISBN 978-5-507-44988-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/276659> (дата обращения: 29.03.2023). — Режим доступа: для авториз. пользователей.

Дополнительная литература

1. «Операционные системы. Программное обеспечение : учебник / составитель Т. П. Куль. — Санкт-Петербург : Лань, 2020. — 248 с. — ISBN 978-5-8114-4290-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/131045> (дата обращения: 29.03.2023). — Режим доступа: для авториз. пользователей.
2. Донцов, В. П. Linux на примерах : руководство / В. П. Донцов, И. В. Сафин. — Санкт-Петербург : Наука и Техника, 2017. — 352 с. — ISBN 978-5-94387-742-1. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/101550> (дата обращения: 29.03.2023). — Режим доступа: для авториз. пользователей.
3. Фишерман, Л. В. Git. Практическое руководство. Управление и контроль версий в разработке программного обеспечения : руководство / Л. В. Фишерман. — Санкт-Петербург : Наука и Техника, 2021. — 304 с. — ISBN 978-5-94387-547-2. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/191470> (дата обращения: 29.03.2023). — Режим доступа: для авториз. пользователей.

МОДУЛЬ 4:

Основная литература

1. Янцев, В. В. Web-программирование на Python / В. В. Янцев. — 2-е изд., стер. — Санкт-Петербург : Лань, 2023. — 180 с. — ISBN 978-5-507-46546-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/310289> (дата обращения: 30.03.2023). — Режим доступа: для авториз. пользователей.

Дополнительная литература

1. Эрик Мэтиз «Изучаем Python. Программирование игр, визуализация данных, веб-приложения»
2. Майкл Доусон «Программируем на Python»
3. Дэн Бейдер «Чистый Python. Тонкости программирования для профи»
4. Гарри Персиваль «Python. Разработка на основе тестирования»

МОДУЛЬ 5:

Основная литература

1. Седжвик, Р. Алгоритмы на C++ : учебное пособие / Р. Седжвик. — 2-е изд. — Москва : ИНТУИТ, 2016. — 1772 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/100565> (дата обращения: 29.03.2023). — Режим доступа: для авториз. пользователей.

Дополнительная литература

1. Бабенко, М. А. Введение в теорию алгоритмов и структур данных / М. А. Бабенко, М. В. Левин. — Москва : МЦНМО, 2016. — 144 с. — ISBN 978-5-4439-2396-3. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/80136> (дата обращения: 29.03.2023). — Режим доступа: для авториз. пользователей.
2. Хайнеман Д., Поллис Г., Селков С. Алгоритмы. Справочник с примерами на C, C++, Java и Python. — 2017.

МОДУЛЬ 6:

Основная литература

1. Сейерс, Э. Х. Docker на практике / Э. Х. Сейерс, А. Милл ; перевод с английского Д. А. Беликов. — Москва : ДМК Пресс, 2020. — 516 с. — ISBN 978-5-97060-772-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/131719> (дата обращения: 29.03.2023). — Режим доступа: для авториз. пользователей.

Дополнительная литература

1. Моуэт, Э. Использование Docker / Э. Моуэт ; научный редактор А. А. Маркелов ; перевод с английского А. В. Снастина. — Москва : ДМК Пресс, 2017. — 354 с. — ISBN 978-5-97060-426-7. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/93576> (дата обращения: 29.03.2023). — Режим доступа: для авториз. пользователей.

6. Оценка качества освоения программы

Оценка качества освоения программы осуществляется в процессе промежуточной аттестации. Формы и методы промежуточного контроля представлены в таблице.

Наименование модулей	Основные показатели оценки	Форма аттестации
Основы программирования	Качество выполнения проектной работы	проект
Базы данных и SQL	Количество правильных ответов	тестирование
Linux / Git для начинающих	Количество правильных ответов	тестирование
Разработка на Python	Качество выполнения проектной работы	проект

Введение в алгоритмы и структуры данных	Количество правильных ответов	тестирование
Docker и CI/CD для начинающих	Количество правильных ответов	тестирование
Итоговая проектная работа	Качество выполнения проектной работы	проект

7. Составители программы:

Райгородский Андрей Михайлович

Доктор физико-математических наук, директор ФПМИ МФТИ

Благодарный Евгений Владимирович

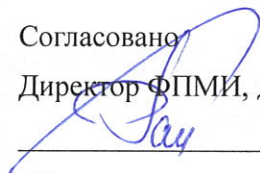
заведующий учебно-методической лабораторией инноватики

Иванова Анастасия Сергеевна

руководитель проектов учебно-методической лаборатории инноватики

Согласовано

Директор ФПМИ, д.ф.-м.н.

 А.М. Райгородский

**Пояснительная записка
к разработке и реализации дополнительной программы
профессиональной переподготовки**

«Python-разработчик Junior»

Целью программы профессиональной переподготовки является получение навыков разработки программного обеспечения на языке Python, на технологиях, популярных в среде разработчиков Python. Будут рассмотрены основы программирования на Python, средства разработки, сборки и тестирования программ, основные алгоритмы и структуры данных, современные инструменты разработки программного обеспечения.

Данный курс предназначен для слушателей с отсутствующим опытом программирования и создания приложений на языке программирования Python. Участники получают основные сведения об объектно-ориентированном программировании, терминологии и синтаксисе языка, научатся создавать простые программы на языке программирования Python. Также участники программы освоят работу с базами данных при помощи языка запросов SQL, алгоритмы и структуры данных, инструменты Linux, Git, Docker, технологии и инструменты CI/CD.

Важную часть программы данного курса составляют практические задания, поэтому учащиеся смогут получить реальный опыт разработки программного обеспечения.

В результате обучения слушатели освоят:

- базовые понятия в программировании;
- область применения и синтаксис python;
- принципы написания эффективного и безопасного кода на python;
- средства обработки/отладки приложений на python;
- основные концепции ООП;
- работу с базами данных с помощью SQL;
- алгоритмы и структуры данных;
- работу с операционной системой Linux;
- работу с системой контроля версий GIT;
- инструмент контейнеризации Docker;
- инструменты CI/CD.

Задачами профессиональной переподготовки являются:

- приобретение студентами навыков проектирования и реализации приложений на языке Python с использованием приемов объектно-ориентированного программирования;
- овладение студентами современных практик разработки: использование IDE, системы контроля версий, инструментов контейнеризации, инструментов CI/CD.

Целевая аудитория программы:

- специалисты без опыта программирования;
- специалисты, желающие переквалифицироваться в разработчиков программного обеспечения на языке Python.

Программа будет построена по очному принципу с применением дистанционных технологий (с применением электронного обучения и дистанционных образовательных технологий) в формате вебинаров, которые займут 162 ак.ч.

Преподавателями программы будут являться преподаватели МФТИ, имеющие опыт преподавания и консультирования по заявленным модулям, а также опыт практической работы в отечественных и/или зарубежных организациях в сфере деятельности, совпадающей с направлением преподавания.

Составители программы:

Райгородский Андрей Михайлович

доктор физико-математических наук, директор ФПМИ МФТИ

Благодарный Евгений Владимирович

заведующий учебно-методической лабораторией инноватики

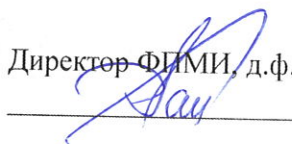
Ивченко Олег Николаевич

старший преподаватель кафедры АТП ФПМИ

Иванова Анастасия Сергеевна

руководитель проектов учебно-методической лаборатории инноватики ФПМИ

Директор ФПМИ, д.ф.-м.н.



А.М. Райгородский

КВАЛИФИКАЦИЯ И ОПЫТ ПРИВЛЕКАЕМОГО ПРЕПОДАВАТЕЛЬНОГО СОСТАВА ДЛЯ РЕАЛИЗАЦИИ ПРОГРАММЫ ПРОФЕССИОНАЛЬНОЙ ПЕРЕПОДГОТОВКИ «Python-разработчик Junior»

Ф.И.О. лектора, год рождения	Информация об образовании, полученном в соответствии с образовательными программами высшего профессионального образования, дополнительного профессионального образования (в т.ч. о наличии званий и ученых степеней) и т.д.	Место работы, занимаемая должность в настоящий момент, общий трудовой стаж, педагогический стаж	Опыт преподавания и консультирования по предмету, согласующемуся с направлением лота (перечислить), преподавательский стаж	Наличие опыта практической работы в отечественных и зарубежных организациях в сфере деятельности, совпадающей с направлением преподавания
Боярников Илья Алексеевич, 2001	Московский физико-технический институт (национальный исследовательский университет), Физтех-школа прикладной математики и информатики, Прикладная математика и информатика (бакалавриат).	Преподаватель курса «Программирование на языке Python» ФПМИ МФТИ. Общий трудовой и педагогический стаж 2,5 года.	Преподаватель курса «Программирование на языке Python» ФПМИ МФТИ. Педагогический стаж 2,5 года.	Разработчик, Яндекс.
Инанц Гайк Ашотович, 1992	ИИИТБ, Защиты информации, Комплексная защита объектов информатизации.	МФТИ. Должность: Преподаватель Высшая школа информационных технологий и безопасности HackerU. Должность: Преподаватель СПАО «Ингосстрах» (проект «Ингоскод»). Должность: преподаватель.	Преподавательский стаж: более 4 лет. Компании и проекты: HackerU, IThub, МФТИ, Сбербанк, MTS, Ингосстрах, Школа СТО яндекс.	Более 5 лет опыта работы в сфере работы с данными. Работа над КХД таких банков, как: ВТБ, Райффайзен банк, Альфа банк.
Чернецкий Аркадий Михайлович, 1992	Факультет инноваций и высоких технологий (ФИИВТ) МФТИ, Прикладная математика и информатика (бакалавриат, 2016).	Преподаватель кафедры алгоритмов и технологий программирования ФПМИ МФТИ. Общий трудовой и педагогический стаж 6 лет.	Преподаватель курсов «Разработка на C++», «Алгоритмы и структуры данных», «Технологии программирования», «Параллельные и распределенные вычисления» кафедры алгоритмов и технологий программирования ФПМИ	Администратор инфраструктуры кафедры алгоритмов и технологий программирования ФПМИ МФТИ.

				МФТИ. Общий педагогический стаж 6 лет.	
Кисляков Иван Дмитриевич, 2001	Физтех-школы прикладной математики и информатики МФТИ, ПМИ, кафедры дискретной математики (бакалавриат).	ГБОУ Школа 67, ПДО. Общий трудовой и педагогический стаж 1,5 года.	Преподаватель курса по алгоритмам и структурам данных в ГБОУ Школа 67 и Лицее Яндекса. Педагогический стаж 1,5 года.	ООО «Мир технологий», отдел разработки ПО, инженер-программист.	
Рябухин Илья Алексеевич, 1999	Бакалавриат (2021), специальность Software Engineering, ВУЗ Czech Technical University in Prague. Магистратура (2023), Prague University of Economics and Business, специальность Information Systems Management. Дополнительное образование: акселератор Физтех Старт (1 волна, 2017), интенсив по TexПреду от МФТИ.	DHL IT Services, Associate Consultant. Общий трудовой и педагогический стаж 4 года.	Преподаватель ФКН НИУ ВШЭ с 2021, дисциплина «Научно-исследовательский семинар «Введение в облачные технологии». Преподавательский стаж 2 года.	DHL IT Services, Associate Consultant.	