

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Ливанов Дмитрий Викторович

Должность: Ректор

Дата подписания: 19.03.2026 09:43:51

Уникальный программный ключ:

c6d909c49c1d2034fa3a0156c4eaa51e7232a3a

Федеральное государственное автономное образовательное учреждение
высшего образования

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

ВЫПИСКА ИЗ ПРОТОКОЛА № 2

заседания учебно-методического совета от « 30 » сентября 2022 г.

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 программу профессиональной
переподготовки «Python – разработчик»

Решение принято единогласно.

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

«30» сентября 2022 г.

**Дополнительная профессиональная программа
Программа профессиональной переподготовки**

«Python-разработчик»

Москва 2022

Оглавление

1. Рабочая группа	2
2. Профессиональные стандарты, учтённые в содержании программы	2
3. Квалификационные требования, учтённые в содержании программы	2
4. Требования федерального государственного стандарта высшего образования, на основании которых была разработана программа	2
5. Характеристика новой квалификации и связанных с ней видов профессиональной деятельности, трудовых функций и (или) уровней квалификации	2
6. Характеристика компетенций, подлежащих совершенствованию, и (или) перечень компетенций, формирующихся в результате освоения программы	5
7. Цель программы	5
8. Планируемые результаты обучения по программе	5
9. Учебный план	6
10. Календарный учебный график	8
11. Рабочие программы (учебная программа)	9
12. Организационно-педагогические условия реализации дополнительной профессиональной программы	22
12.1. Требования к квалификации педагогических кадров/представителей предприятий и организаций, обеспечивающих реализацию образовательного процесса	22
12.2. Требования к материально-техническим условиям	22
12.3. Требования к информационным и учебно-методическим условиям	23
12.4. Общие требования к организации образовательного процесса	25
13. Формы аттестации и оценочные материалы по программе	26

1. Рабочая группа

С целью повышения качества дополнительной профессиональной программы (ДПП) с учетом требований профессиональных стандартов (ПС) в группу разработчиков входили:

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

2. Профессиональные стандарты, учтённые в содержании программы

Настоящая дополнительная профессиональная программа (ДПП) разработана с учётом соответствующих ей профессиональных стандартов (ПС) из национального реестра профессиональных стандартов. Профессиональные стандарты (ПС), связанные с настоящей ДПП: 06.001 Программист

3. Квалификационные требования, учтённые в содержании программы

Квалификационные требования, учтённые в содержании программы, указаны в описании должностей единого квалификационного справочника должностей руководителей, специалистов и служащих. Причём учтены требования должностей, указанных в дополнительных характеристиках, отобранных для данной программы ОТФ (п. 5. настоящего документа), которые приведены в указанных в п. 2. настоящего документа профессиональных стандартах (ПС) в строке «ЕТКС или ЕКС». Квалификационные требования инкорпорированы в настоящий документ путём отсылки и являются его неотъемлемой частью.

4. Требования федерального государственного стандарта высшего образования, на основании которых была разработана программа

Настоящей дополнительной профессиональной программе соответствует федеральный государственный стандарт высшего образования (ФГОС ВО): 01.03.02 «Прикладная математика и информатика (уровень бакалавриата)».

Настоящая дополнительная профессиональная программа разработана на основании требований указанного ФГОС ВО.

Полный перечень требований приведён в тексте указанного ФГОС ВО, инкорпорирован в настоящий документ путём отсылки и является его неотъемлемой частью.

5. Характеристика новой квалификации и связанных с ней видов профессиональной деятельности, трудовых функций и (или) уровней квалификации

«Программа профессиональной переподготовки направлена на получение компетенции, необходимой для выполнения нового вида профессиональной деятельности, приобретение новой квалификации (часть 5 статьи 76 Федерального закона N 273-ФЗ). Для их определения и может использоваться ПС. При этом необходимо учесть различия терминологии, используемой в образовании и в профессиональных стандартах, о которых говорилось выше. Вид

профессиональной деятельности, квалификация, упоминаемые в цитируемой статье закона, в ПС в большинстве случаев соответствуют ОТФ, иногда - ТФ.»¹,

Новые квалификации (ОТФ/ТФ) и их уровни:

06.001 Программист

Обобщенные трудовые функции			Трудовые функции		
код	наименование	уровень квалификации	наименование	код	уровень (подуровень) квалификации
А	Разработка и отладка программного кода	3	Формализация и алгоритмизация поставленных задач	A/01.3	3
			Написание программного кода с использованием языков программирования, определения и манипулирования данными	A/02.3	3
			Оформление программного кода в соответствии с установленными требованиями	A/03.3	3
			Работа с системой контроля версий	A/04.3	3
			Проверка и отладка программного кода	A/05.3	3
			Проверка работоспособности и рефакторинг кода программного обеспечения	В	4

¹ Письмо Министерства образования и науки РФ от 22 апреля 2015 г. N ВК-1032/06 "О направлении методических рекомендаций"

			Разработка тестовых наборов данных	V/02.4	4
			Проверка работоспособности программного обеспечения	V/03.4	4
			Рефакторинг и оптимизация программного кода	V/04.4	4
			Исправление дефектов, зафиксированных в базе данных дефектов	V/04.5	4
C	Интеграция программных модулей и компонент и проверка работоспособности выпусков программного продукта	5	Разработка процедур интеграции программных модулей	C/01.5	5
			Осуществление интеграции программных модулей и компонент и верификации выпусков программного продукта	C/02.5	5
D	Разработка требований и проектирование программного обеспечения	6	Анализ требований к программному обеспечению	D/01.6	6
			Разработка технических спецификаций на программные компоненты и их взаимодействие	D/02.6	6
			Проектирование программного обеспечения	D/03.6	6

Характеристики данных ОТФ/ТФ приведены в профессиональном стандарте (ПС), указанном в п. 2. настоящего документа. инкорпорированы в настоящий документ путём отсылки, и являются его неотъемлемой частью.

Связанные виды профессиональной деятельности (ВПД):

научно-исследовательская;

производственно-технологическая;

проектная.

Характеристика данных ВПД приведена в федеральном государственном стандарте высшего образования (ФГОС ВО), указанном в п. 4. настоящего документа, инкорпорирована в настоящий документ путём отсылки, и является его неотъемлемой частью. Указанные ВПД, освоение которых предусмотрено указанным ФГОС ВО, являются «сквозными» и служат основой овладения выбранной квалификацией (ОТФ/ТФ).

ТФ, связанные с указанными в настоящем пункте ОТФ (в случае их указания), а также их характеристики приведены в профессиональном стандарте (ПС), указанном в п. 2. настоящего документа. инкорпорированы в настоящий документ путём отсылки, и являются его неотъемлемой частью.

6. Характеристика компетенций, подлежащих совершенствованию, и (или) перечень компетенций, формирующихся в результате освоения программы

Учитывая, что слушатели программы уже освоили или осваивают основную профессиональную образовательную программу, в том числе общие компетенции, соответствующего уровня профессионального образования, компетенции формирующиеся в результате освоения настоящей программы и компетенции, подлежащие совершенствованию, а также их характеристика, представлены в ФГОС, указанном в п. 4. настоящего документа, инкорпорированы в настоящий документ путём отсылки, и являются его неотъемлемой частью.

Данные компетенции указанного ФГОС ВО служат основой определения результатов обучения по программе.

7. Цель программы

Реализация программы профессиональной переподготовки направлена на получение компетенции, необходимой для выполнения нового вида профессиональной деятельности, приобретение новой квалификации.

В соответствии с Письмом Министерства образования и науки РФ от 22 апреля 2015 г. N ВК-1032/06 «О направлении методических рекомендаций», цель представляет собой осознанное представление (предвосхищение) результата деятельности. Планируемые результаты отражены ниже.

8. Планируемые результаты обучения по программе

Планируемые результаты обучения соответствуют результатам освоения соответствующей основной профессиональной образовательной программы, а также направлены на приобретение новой квалификации, требующей изменение направленности (профиля) или специализации в рамках направления подготовки (специальности) полученного ранее профессионального образования, определены на основе профессиональных компетенций соответствующего федерального государственного образовательного стандарта, указанного в п. 4. настоящего документа.

Имеющаяся квалификация и (или) уровень образования (требования к слушателям): Высшее, незаконченное высшее образование или среднее профессиональное

Новые виды профессиональной деятельности, и ПК, соответствующие им, которые планируются к получению в результате обучения по настоящей программе, а также их характеристики, представлены в ФГОС, указанном в п. 4. настоящего документа, инкорпорированы в настоящий документ путём отсылки, и являются его неотъемлемой частью.

Выпускник также должен обладать прочими компетенциями в соответствие с указанным ФГОС, которые приведены и охарактеризованы в нём, инкорпорированы в настоящий документ путём отсылки, и являются его неотъемлемой частью.

Новые квалификации (ОТФ/ТФ), планируемые к приобретению в результате обучения по настоящей программе, приведены и охарактеризованы в п. 5. настоящего документа.

Приведённая информация о цели и результатах обучения является основой для разработки рабочих программ, оценочных материалов и иных компонентов дополнительной профессиональной программы.

9. 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. Работа с базами данных и SQL. Linux / Git для начинающих	100	40	40	19	1	проект
2	Модуль 2. Разработка на Python	200	51	24	110	15	проект
3	Модуль 3. Введение в алгоритмы и структуры данных	80	16	16	40	8	тестирование
4	Модуль 4. Docker и CI/CD для начинающих	40	2	1	30	7	тестирование
5	Итоговая квалификационная работа	40			40		проект
	ИТОГО:	460	109	81	239	31	

* Слушатели могут проходить модули в произвольном порядке.

Срок освоения настоящей программы профессиональной переподготовки превышает минимально допустимый срок освоения 250 часов (либо другой установленный актуальными нормативными документами соответствующего Министерства срок).

10. Календарный учебный график

Модуль 1. Работа с базами данных и SQL. Linux / Git для начинающих

Обучение по установленной форме: <i>очная, с применением электронного обучения и дистанционных образовательных технологий</i>																					
Количество академических часов: 100																					
Порядковый № занятия	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Лекции	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	0
Практические занятия	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	0
Самостоятельная работа	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	0,5	9
Контрольные задания	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Итого	4,5	4,5	4,5	4,5	4,5	4,5	4,5	4,5	4,5	4,5	4,5	4,5	4,5	4,5	4,5	4,5	4,5	4,5	4,5	4,5	10

Модуль 2. Разработка на Python

Обучение по установленной форме: <i>очная, с применением электронного обучения и дистанционных образовательных технологий</i>																																								
Количество академических часов: 200																																								
Порядковый № занятия	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
Лекции	2	2	2	2	2	0	0	2	2	2	2	0	0	2	2	2	2	0	0	2	2	2	2	3	2	2	0	0	0	2	2	2	2	3	2	2	0	0	0	
практические	1	1	1	1	1	3	0	1	1	1	1	3	3	1	1	1	1	3	0	1	1	1	1	0	1	1	3	3	0	1	1	1	1	0	1	1	3	3	0	
самостоятельные	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
Контрольные задания	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	3
Итого	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5

Модуль 3. Введение в алгоритмы и структуры данных

Обучение по установленной форме: <i>очная, с применением электронного обучения и дистанционных образовательных технологий</i>								
Количество академических часов: 80								
Порядковый № занятия	1	2	3	4	5	6	7	8
Лекции	2	2	2	2	2	2	2	2
Практические занятия	2	2	2	2	2	2	2	2
Самостоятельная работа	5	5	5	5	5	5	5	5
Контрольные задания	1	1	1	1	1	1	1	1
Итого	10	10	10	10	10	10	10	10

Модуль 4. Docker и CI/CD для начинающих

<i>Обучение по установленной форме: заочная, с применением электронного обучения и дистанционных образовательных технологий</i>			
Количество академических часов: 40			
Порядковый № занятия	1	2	3
Лекции	1	0,5	0,5
практические	0,5	0	0,5
самостоятельные	10	10	10
Контрольные задания	2	2	3
Итого	13,5	12,5	14

Итоговая квалификационная работа

<i>Обучение по установленной форме: очная, с применением электронного обучения и дистанционных образовательных технологий</i>	
Количество академических часов: 40	
самостоятельная работа	40
Итого	40

11. Рабочие программы (учебная программа)

Модуль 1. Работа с базами данных и SQL. Linux / Git для начинающих

№ п/п	Содержание обучения, наименование и тематика практических занятий (вебинаров), самостоятельных работы	Объем, ак. час.
1.1	Модуль I. SQL базовый	40,5
1.1.1	Базовый синтаксис SQL	4,5
	Лекция. Базовые понятия и команды для формирования таблицы в базе данных	2
	Практическое занятие по теме лекции	2
	Самостоятельная работа	0,5
	Выполнение контрольных заданий	0

1.1.2	Создание и заполнение таблиц	4,5
	Лекция. Создание и заполнение таблиц (понятие DDL и DML, создание таблиц, изменение таблиц, добавление данных, изменение данных, удаление данных)	2
	Практическое занятие по теме лекции	2
	Самостоятельная работа	0,5
	Выполнение контрольных заданий	0
1.1.3	Объединение данных	4,5
	Лекция. Union / Union All ,Особенности Join	2
	Практическое занятие по теме лекции	2
	Самостоятельная работа	0,5
	Выполнение контрольных заданий	0
1.1.4	Объединение данных (продолжение)	4,5
	Лекция. Объединение данных (Типы Join и различия между ними)	2
	Практическое занятие по теме лекции	2
	Самостоятельная работа	0,5
	Выполнение контрольных заданий	0
1.1.5	Агрегация	4,5
	Лекция. Агрегация (агрегация, группировка having, применение подзапросов, оператор case)	2
	Практическое занятие по теме лекции	2

	Самостоятельная работа	0,5
	Выполнение контрольных заданий	0
1.1.6	Повторение пройденного материала	4,5
	Лекция. Повторение пройденного материала (повторение пройденных тем)	2
	Практическое занятие по теме лекции	2
	Самостоятельная работа	0,5
	Выполнение контрольных заданий	0
1.1.7	Очистка данных	4,5
	Лекция. Очистка данных (строковые методы, очистка данных, отработка очистки данных)	2
	Практическое занятие по теме лекции	2
	Самостоятельная работа	0,5
	Выполнение контрольных заданий	0
1.1.8	Регулярные выражения	4,5
	Лекция. Регулярные выражения (синтаксис регулярных выражений, методы работы с регулярными выражениями, отработка очистки данных)	2
	Практическое занятие по теме лекции	2
	Самостоятельная работа	0,5
	Выполнение контрольных заданий	0
1.1.9	Проектирование базы данных	4,5
	Лекция. Проектирование баз данных	2
	Практическое занятие по теме лекции	2

	Самостоятельная работа	0,5
	Выполнение контрольных заданий	0
1.2	Модуль II: SQL продвинутый. Linux / Git для начинающих.	49,5
1.2.1.	Знакомство с Python	4,5
	Лекция. Знакомство с Python	2
	Практическое занятие по теме лекции	2
	Самостоятельная работа	0,5
	Выполнение контрольных заданий	0
1.2.2	Работа со словарями	4,5
	Лекция. Работа со словарями	2
	Практическое занятие по теме лекции	2
	Самостоятельная работа	0,5
	Выполнение контрольных заданий	0
1.2.3	Функции	4,5
	Лекция. Функции	2
	Практическое занятие по теме лекции	2
	Самостоятельная работа	0,5
	Выполнение контрольных заданий	0
1.2.4	Работа с файловой системой	4,5
	Лекция. Работа с файловой системой	2

	Практическое занятие по теме лекции	2
	Самостоятельная работа	0,5
	Выполнение контрольных заданий	0
1.2.5	Взаимодействие SQL и Python	4,5
	Лекция. Взаимодействие SQL и Python	2
	Практическое занятие по теме лекции. решение прикладных задач	2
	Самостоятельная работа	0,5
	Выполнение контрольных заданий	0
1.2.6	Паттерны хранения данных	4,5
	Лекция. Паттерны хранения данных (логическое удаление, исторические данные)	2
	Практическое занятие по теме лекции	2
	Самостоятельная работа	0,5
	Выполнение контрольных заданий	0
1.2.7	Инкрементальная загрузка	4,5
	Лекция. Инкрементальная загрузка (процесс инкрементальной загрузки, реализация процесса с нуля на примере данных, обсуждение сложностей в процессе)	2
	Практическое занятие по теме лекции	2
	Самостоятельная работа	0,5
	Выполнение контрольных заданий	0
1.2.8	Инкрементальная загрузка. Работа с bash-терминалом	4,5
	Лекция. Инкрементальная загрузка. Работа с bash-терминалом	2

	Практическое занятие по теме лекции	2
	Самостоятельная работа	0,5
	Выполнение контрольных заданий	0
1.2.9	Оконные функции	4,5
	Лекция. Оконные функции	2
	Практическое занятие по теме лекции	2
	Самостоятельная работа	0,5
	Выполнение контрольных заданий	0
1.2.10	Git и Github	4,5
	Лекция. Git и Github	2
	Практическое занятие по теме лекции	2
	Самостоятельная работа	0,5
	Выполнение контрольных заданий	0
1.2.11	Знакомство с Linux. Работа с BASH командами	4,5
	Лекция. Знакомство с Linux. Работа с BASH командами	2
	Практическое занятие по теме лекции	2
	Самостоятельная работа.	0,5
	Выполнение контрольных заданий.	0
1.3.	Итоговая аттестация - Финальный проект	10
	Самостоятельная работа. Выполнение проекта	9
	Выполнение контрольных заданий	1

Итого:	100
---------------	------------

Модуль 2. 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 3. Введение в алгоритмы и структуры данных

№ п/п	Содержание обучения, наименование и тематика практических занятий (вебинаров), самостоятельных работы	Объем, ак.час.
3.1.	Введение	10
	Теория:	2

	<ol style="list-style-type: none"> 1. Орг. вопросы, Яндекс-контеcт, работа в системе, O-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. 	
	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
3.2.	Сортировки	10
	Теория: <ol style="list-style-type: none"> 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
3.3.	Динамическое программирование. Бинарный (экспоненциальный) поиск.	10
	Теория: <ol style="list-style-type: none"> 1. Задача поиска числа в контейнере. Вычисление асимптотики. Бинарный поиск - зачем использовать. 2. Бинарный и экспоненциальный поиск. 3. Динамическое программирование - что это и зачем надо. Задача о рюкзаке. (He)точность дп. 4. Задача о рюкзаке с дополнительным условием. Двумерное динамическое программирование. 	2
	Практика: Бинарный и экспоненциальный поиск - реализация.	2
	Самостоятельная работа: Решение задач на бинарный поиск. Динамическое программирование одномерное и двумерное.	5

	Выполнение контрольных заданий.	1
3.4.	Хеширование. Метод цепочек.	10
	Теория: <ol style="list-style-type: none"> 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. 	2
	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
3.5.	Хеш-таблицы с открытой адресацией	10
	Теория: <ol style="list-style-type: none"> 1. Разрешение коллизии методом двойного хеширования. 2. Две хеш-функции, их свойства. Реализация алгоритма. 	2
	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
3.6.	Куча	10
	Теория: <ol style="list-style-type: none"> 1. Определение и свойства кучи. 2. Вставка, Поиск максимума, Удаление. Реализация абстрактного типа данных. 3. Эффективное построение кучи. Взятие максимуму/минимума. Какие задачи решает - в чем эффективность кучи. 4. Пирамидальная сортировка. 	2
	Практика: Построение и реализация кучи.	2

	Самостоятельная работа: Построение кучи, взятие максимума/минимума, пирамидальная сортировка.	5
	Выполнение контрольных заданий.	1
3.7.	Простые деревья и граф	10
	Теория: <ol style="list-style-type: none"> 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
3.8.	Сбалансированные деревья	10
	Теория: <ol style="list-style-type: none"> 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 4. Docker и CI/CD для начинающих

№ п/п	Содержание обучения, наименование и тематика практических занятий (вебинаров), самостоятельных работы	Объем, ак.час.
4.1	Docker	
	Теория: <ol style="list-style-type: none"> 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	10
	Выполнение контрольных заданий	2
4.2	Docker Compose	
	Теория: <ol style="list-style-type: none"> 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	10
	Выполнение контрольных заданий	2
4.3	DevOps. GitLab CI	
	Теория: <ol style="list-style-type: none"> 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	10

	Выполнение контрольных заданий	3
	Итого	40

5.	Итоговая квалификационная работа	40
	Самостоятельная работа.	40
	Итого	460

Содержание дополнительной профессиональной программы направлено на достижение результатов её целей (планируемых результатов).

12. Организационно-педагогические условия реализации дополнительной профессиональной программы

12.1. Требования к квалификации педагогических кадров/представителей предприятий и организаций, обеспечивающих реализацию образовательного процесса

Требования к образованию педагогических и иных работников, а также (при наличии) требования к освоению ими дополнительных профессиональных программ, опыту работы в области профессиональной деятельности, соответствующей направленности ДПП: высшее образование, трудовой и (или) педагогический стаж от полугода, наличие опыта практической работы в сфере деятельности, совпадающей с направлением преподавания.

12.2. 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	Практические занятия (дистанционные)	Слушателю необходимо наличие доступа в сеть интернет,

дистанционных семинаров	семинары)	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.

12.3. Требования к информационным и учебно-методическим условиям

Список литературы:

МОДУЛЬ 1:

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.
3. 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. 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

4. «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.
5. 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.
6. 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 2:

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 3:

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 4:

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.

12.4. Общие требования к организации образовательного процесса

Форма обучения – очная / заочная, с применением дистанционных технологий.

Режим обучения – 2-3 раз(а) в неделю по 3-4 академических часа.

Преподаватель проводит практические занятия дистанционно в форме вебинаров с использованием платформы ZOOM (или аналогичной).

Видеоматериалы и дополнительные материалы выкладываются на обучающую платформу SkillSpace.

Самостоятельная работа выполняется слушателем в удобном для слушателя режиме.

В Таблице ниже описаны образовательные технологии.

№ п/п	Вид занятия	Форма проведения занятий	Цель
1	Лекция	Изложение материала посредством лекций, обсуждение общих вопросов по тематике курса / Самостоятельный просмотр видеолекций на обучающей платформе	Ознакомление слушателей с базовым материалом по тематике курса
2	Практическое занятие	Выполнение практических заданий, получение обратной связи от преподавателя.	Практическое освоение теоретических знаний, Разбор ситуаций, а также углубление знаний по курсу
3	Самостоятельная работа	Самостоятельное изучение дополнительных материалов и литературы. Выполнение тренировочных тестов и заданий.	Углубление знаний по курсу. Применение знаний к своей организации / проекту
4	Выполнение контрольных	Выполнение тестов, проверочных заданий.	Практическое освоение теоретических знаний, контроль

	заданий		освоения материалов.
5	Итоговая аттестация	Подготовка итоговой аттестационной работы.	Практическое освоение теоретических знаний, контроль освоения материалов. Получение практического результата в своих организациях / проектах

13. Формы аттестации и оценочные материалы по программе

Оценка качества освоения программы проводится по двухбалльной системе: «зачтено», «не зачтено» по результатам промежуточного контроля (тестирование, проверочные задания на взаимную оценку), контроля посещаемости практических занятий (вебинаров) и результатам итоговой аттестации.

Слушатель считается аттестованным в случае положительных результатов работы (не менее 70% баллов от итоговой оценки) в процессе обучения и успешной сдачи экзамена. При этом баллы за экзамены начисляются только при достижении 50% порога при прохождении каждого экзаменационного испытания. После аттестации слушатель получает оценку «отлично», если набрано не меньше 80% баллов от возможного максимума, «хорошо», если набрано не меньше 65%, «удовлетворительно», если набрано не меньше 40%, в противном случае слушатель курс не сдает.

Результат тестирования, решения проверочных заданий и написания кода проверяется автоматически системой на образовательной платформе. Итоговая аттестационная работа проверяется преподавателем

Составляющие процесса обучения, которые оцениваются в ходе обучения, и их вклад в итоговую оценку представлены в Таблице ниже.

Таблица – Составляющие процесса обучения

	Основные показатели оценки	Вклад в итоговую оценку
1	Основной курс обучения на образовательной платформе	80%
2	Итоговая аттестация	20%

Оценочные материалы:

1. Пример заданий текущего контроля к модулю 1 «Работа с базами данных и SQL. Linux / Git для начинающих»

1) Схема БД состоит из четырех таблиц:

- Product(maker, model, type)
- PC(code, model, speed, ram, hd, cd, price)
- Laptop(code, model, speed, ram, hd, price, screen)
- Printer(code, model, color, type, price)

Таблица Product представляет производителя (maker), номер модели (model) и тип ('PC' - ПК, 'Laptop' - ПК-блокнот или 'Printer' - принтер). Предполагается, что номера моделей в таблице Product уникальны для всех производителей и типов продуктов. В таблице PC для каждого ПК, однозначно определяемого уникальным кодом – code, указаны модель –

model (внешний ключ к таблице Product), скорость - speed (процессора в мегагерцах), объем памяти - ram (в мегабайтах), размер диска - hd (в гигабайтах), скорость считывающего устройства - cd (например, '4x') и цена - price (в долларах). Таблица Laptop аналогична таблице PC за исключением того, что вместо скорости CD содержит размер экрана - screen (в дюймах). В таблице Printer для каждой модели принтера указывается, является ли он цветным - color ('y', если цветной), тип принтера - type (лазерный - 'Laser', струйный - 'Jet' или матричный - 'Matrix') и цена - price.

Найдите производителей ПК с процессором не менее 450 МГц. Вывести: Maker

- 2) Необходимо разработать python приложение, у которого есть следующие функции:
 1. Функция, которая создает таблицу client. Если она уже есть, необходимо вывести об этом сообщение в консоль
 - a. Таблица client должна содержать в себе следующие поля:
 - b. name
 - c. lastname
 - d. age
 2. Функция, которая добавляет клиентов в таблицу (если клиента с такими фамилией и именем нет в таблице)
 3. Функция, которая получив путь до JSON файла и добавляет клиентов в таблицу (только тех, которые по фамилии и имени отсутствуют в таблице)

2. Пример тестовых заданий к модулю 2 «Разработка на Python»

- 1) Создайте лист L2 с числами от 0 до 9 (включительно).
- 2) Напишите генераторную функцию generate(x), которая создаёт генераторы которые бесконечно возвращают число x при итерировании.
- 3) Напишите генератор only_even(X), который берёт на вход контейнер или генератор и возвращает из него только чётные по порядку элементы.

3. Пример тестовых заданий к модулю 3 «Введение в алгоритмы и структуры данных»

- 4) Реализовать АТД очередь на двух АТД стеках. Реализовать методы push(), pop, size, is_empty.
- 5) Реализуйте сортировку слиянием merge_sort().
- 6) Решите задачу: Используя АТД стек, по обратной Польской нотации выражения найдите его значение.

4. Пример тестовых заданий к модулю 4 «Docker и CI/CD для начинающих»

- 1) Напишите Dockerfile к приложению на фласке. Потом разверните контейнер на порту 8000
 1. Скачать python и pip
 2. Запустить команду pip install flask, а потом запустить команду python main.py

Пример main.py можно найти тут:

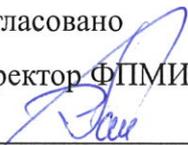
<https://flask.palletsprojects.com/en/1.1.x/quickstart/>

- 2) Сделайте CI/CD пайплан, в котором будет
 - Stage build
 - Job docker build & push - сборка докер образа и отправка в container registry

- Stage test
- Job run test 1 - эмуляция запуска тестов (внутри должен быть любой непустой скрипт)
- Job run test 2 - эмуляция запуска тестов (внутри должен быть любой непустой скрипт)
- Job run test 3 - эмуляция запуска тестов (внутри должен быть любой непустой скрипт)

Согласовано

Директор ФПМИ, д.ф.-м.н.



_____ А.М. Райгородский

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

«Python-разработчик»

Целью программы повышения квалификации 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 небольшим или 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:

- 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;
- ИТ-специалисты, желающие овладеть правилами языка программирования Python и современными инструментами разработки, являющимися стандартами для разработчиков Python;
- 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 190 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

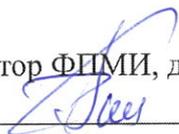
Ивченко Олег Николаевич

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 ПРЕПОДАВАТЕЛЬСКОГО СОСТАВА ДЛЯ РЕАЛИЗАЦИИ ПРОГРАММЫ
ПРОФЕССИОНАЛЬНОЙ ПЕРЕПОДГОТОВКИ «Python-разработчик»**

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

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.	МФТИ. Общий педагогический стаж 6 лет.	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 ТехПреду 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.