

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

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

УТВЕРЖДАЮ

Ректор МФТИ
д-р физ.-мат. наук, профессор



[Handwritten signature]
Д.В. Ливанов

« 29 » *[Handwritten date]* 2024 г.

**Дополнительная профессиональная
программа повышения квалификации
«Разработчик на Python»**

УГСН 01.00.00 Математика и механика

Направление подготовки 01.03.02 Прикладная математика и информатика

ОКВЭД – Деятельность в области информации и связи
63 - Деятельность в области информационных технологий

Москва 2024

1. Общая характеристика программы

1.1 Цель реализации программы

Целью реализации дополнительной профессиональной программы повышения квалификации «Разработчик на Python» является формирование компетенций слушателей в области решения задач в Python, применения шаблонов проектирования на Python, работы с Python библиотеками.

1.2 Совершенствуемые и/или приобретаемые компетенции

Компетенции, формируемые и совершенствуемые в результате обучения, представлены в таблицах 1 и 2.

таблица 1

№	Компетенция в соответствии с профессиональным стандартом 06.001 Программист	Код компетенции
1	Способен к написанию программного кода с использованием языков программирования, определению и манипулированию данными	ПК-1

таблица 2

№	Профессиональные компетенции (ПК) в соответствии с направлением подготовки 01.03.02 Прикладная математика и информатика (уровень бакалавриата)	Код компетенции
1.	Способен применять фундаментальные знания, полученные в области математических и (или) естественных наук, и использовать их в профессиональной деятельности	ОПК 1

1.3. Планируемые результаты обучения

Планируемые результаты обучения представлены в таблице 3.

таблица 3

№	Уметь - знать	«Прикладная математика и информатика» код: 01.03.02
		Квалификация: бакалавриат
		Код компетенции
1	Знать: Возможности языка Python и его особенности Уметь: Работать с данными и таблицами на Python	ОПК-1
2	Знать: Основы работы в Pandas Уметь: Работать с данными в Pandas	ОПК-1
		Профессиональный стандарт 06.001 Программист
		Код компетенции
3	Знать: Синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка	

	<p>программирования, методологии разработки программного обеспечения, методологии и технологии проектирования и использования баз данных, технологии программирования, особенности выбранной среды программирования и системы управления базами данных, компоненты программно-технических архитектур, существующие приложения и интерфейсы взаимодействия с ними</p> <p>Уметь: Применять выбранные языки программирования для написания программного кода, использовать выбранную среду программирования и средства системы управления базами данных. использовать возможности имеющейся технической и/или программной архитектуры</p>	
--	--	--

1.4 Категории обучающихся

Программа повышения квалификации предназначена для специалистов, имеющих высшее или среднее профессиональное образование, а также лиц, получающих высшее или среднее профессиональное образование.

1.5 Форма обучения

Очная с использованием дистанционных образовательных технологий

1.6 Объем программы

144 академических часа

1.7 Режим обучения

12 недель

2. Содержание программы

2.1. Учебный (тематический) план

Учебный (тематический) план программы представлен в таблице 4.

таблица 4

№ п/п	Наименование разделов (модулей) и тем	Всего, час.	Лекции	Практическая работа	Самост. работа	Форма контроля
1	Возможности языка Python и его особенности	64	16	24	24	Домашнее задание 1
2	Основы работы в Pandas	64	16	24	24	Домашнее задание 2
	Итоговая аттестация	16		6	10	Зачет на основании совокупности выполненных домашних

						заданий и защиты проекта
	Итого:	144	32	54	58	

2.2. Учебная программа

Содержание учебной программы приведено в таблице 5.

таблица 5

№ п/п	Наименование модуля, разделов и тем	Содержание обучения, наименование и тематика практических занятий (семинаров), самостоятельной работы	Объем, ак. час.
1	Возможности языка Python и его особенности		64
1.1	Установка Python. Знакомство со средой разработки	Знакомство с языком. Установка интерпретатора Python. Настройка окружения для работы и выбор среды разработки. Задачи: Развернуть окружение для программирования на Python. Узнать базовые сведения о языке. Узнать об организации кода на Python. Получить начальные сведения о внутреннем устройстве интерпретатора Python. Научиться простейшему вводу-выводу данных, применению арифметических операций. Самостоятельная работа: Самостоятельное выполнение заданий по теме лекции	4
1.2	Базовые типы. Конструкции управления потоком: условия. Модули и пакеты	Знакомство с базовыми типами языка и конструкцией <code>if_else</code> Задачи: Освоить базовые типы и конструкцию языка <code>if_else</code> . Структурировать знания о логических операциях и порядке их выполнения. Узнать об принципах импортирования и использования сторонних модулей и библиотек Python. Самостоятельная работа: Самостоятельное выполнение заданий по теме лекции	6
1.3	Конструкции управления потоком: циклы	Знакомство с циклическими конструкциями <code>for, while</code> Задачи: Освоить циклические конструкции <code>for..in, for .. in range, while</code> . Узнать об организации кода с помощью <code>pass, break, continue</code> . Научиться применять циклы на практике. Самостоятельная работа:	5

		Самостоятельное выполнение заданий по теме лекции	
1.4	Знакомство с коллекциями: множества, строки, списки, кортежи	<p>Знакомство с новыми типами данных – коллекциями и методами работы с ними. Использование в коде списочных выражений.</p> <p><i>Задачи:</i></p> <p>Научиться различать изменяемые и неизменяемые типы данных.</p> <p>Научиться работать со стандартными структурами данных в Python: множествами, строками, списками, кортежами.</p> <p>Узнать о методах списков и строк.</p> <p>Освоить использование синтаксической конструкции list comprehension.</p> <p>Самостоятельная работа:</p> <p>Самостоятельное выполнение заданий по теме лекции</p>	5
1.5	Работа со словарями. Списочные выражения Генераторы и генераторные выражения	<p>Знакомство с коллекцией Dict и OrderedDict стандартной библиотеки Python.</p> <p><i>Задачи:</i></p> <p>Продемонстрировать понимание задач, в которых оправдано применение множеств и словарей.</p> <p>Разработать решения задач с применением множеств и словарей.</p> <p>Использовать синтаксис списочных и генераторных выражений для ускорения написания кода</p> <p>Самостоятельная работа:</p> <p>Самостоятельное выполнение заданий по теме лекции</p>	5
1.6	Функции: возвращение значения из функции, области видимости переменных, передача параметра.	<p>Предпосылки появления функций в языках программирования, шаблоны их реализации и использования.</p> <p><i>Задачи:</i></p> <p>Изучить синтаксис пользовательских функций. Рассмотреть варианты вызова функций и возврата ими значений.</p> <p>Изучить понятие области видимости. Рассмотреть примеры использования параметров, в том числе по умолчанию</p> <p>Самостоятельная работа:</p> <p>Самостоятельное выполнение заданий по теме лекции</p>	5
1.7	Продвинутое использование функций. Функции как объект.	<p>Работа с функциями, получающими переменное количество аргументов. Рекурсия и лямбда-функции.</p> <p><i>Задачи:</i></p> <p>Изучить синтаксис обработки переменного числа аргументов пользовательскими функциями.</p>	5

		<p>Рассмотреть особенности использования функций как объекта. Научиться использовать рекурсивные функции Изучить синтаксис и сферы применения лямбда-функций. Самостоятельная работа: Самостоятельное выполнение заданий по теме лекции</p>	
1.8	Массивы Numpy	<p>Знакомство с реализацией массивов в библиотеке Numpy, содержащей большое количество полезных инструментов: от быстрых операций с многомерными массивами до реализации различных математических методов. Задачи: Использовать функции модуля numpy для выполнения операций с одномерными и двумерными массивами. Научиться строить графики различных типов</p> <p>Самостоятельная работа: Самостоятельное выполнение заданий по теме лекции</p>	5
1.9	Введение в объектно-ориентированное программирование	<p>Основные понятия ООП. Принципы, классы и объекты в Python Задачи: Изучить понятия и принципы реализации ООП в Python. Изучить синтаксис объявления классов и методов. Изучить создание объектов и их свойств. Самостоятельная работа: Самостоятельное выполнение заданий по теме лекции</p>	5
1.10	ООП: Абстрагирование и инкапсуляция	<p>Модель хранения данных в объектах и методология создания классов. Задачи: Рассмотреть понятия поля класса и свойства объекта. Изучить методы геттеры и сеттеры. Рассмотреть область видимости, скрытые поля и доступ к ним. Самостоятельная работа: Самостоятельное выполнение заданий по теме лекции</p>	5
1.11	ООП: Наследование	<p>Создание иерархии классов с наследуемыми свойствами. Задачи: Рассмотреть механизмы и синтаксис наследования в Python. Изучить особенности множественного наследования.</p>	5

		Рассмотреть синтаксис обращения к объектам, методам и свойствам внутри иерархии классов. Самостоятельная работа: Самостоятельное выполнение заданий по теме лекции	
1.12	ООП: Полиморфизм	Полиморфизм методов и их перегрузка в иерархии классов. Задачи: Рассмотреть применение принципа полиморфизма. Изучить перегрузку методов родительских классов. Перегрузка арифметических и логических операторов. Самостоятельная работа: Самостоятельное выполнение заданий по теме лекции	5
1.13	Библиотеки Python. Встроенные модули: math, random, datetime, pprint	Библиотеки Python. Встроенные модули: math, random, datetime, pprint. Инструменты работы с библиотеками Python Задачи: Рассмотреть функции и методы стандартных прекомпилированных модулей Python Самостоятельная работа: Самостоятельное выполнение заданий по теме лекции	4
2	Основы работы в Pandas		64
2.1	Инструменты Numpy	Знакомство с элементами линейной алгебры — векторами и матрицами, как основным математическим аппаратом для работы с данными. Использование функций модуля numpy для выполнения операций с матрицами и векторами Самостоятельная работа: Самостоятельное выполнение заданий по теме лекции	7
2.2	Знакомство с библиотекой Pandas. Объекты pandas.Series и pandas.DataFrame	Использование средств модуля pandas для загрузки данных и ознакомления с ними. Знакомство с основными функциями для работы с сериями и датафреймами в pandas. Самостоятельная работа: Самостоятельное выполнение заданий по теме лекции	7
2.3	Анализ взаимосвязей. Описательная статистика. Сводные таблицы	Корреляционный анализ данных и оценка силы и направления взаимосвязи. Меры средней тенденции, меры разброса. Механизм GroupBy, типы группировок. Агрегирование, преобразование, исключение групп. Подготовка данных для создания сводных таблиц. Представление сводных таблиц. Самостоятельная работа:	7

		Самостоятельное выполнение заданий по теме лекции	
2.4	Интеграция данных. Работа с несколькими таблицами	Слияние и конкатенация датафреймов, форматы "Long" и "Wide". Самостоятельная работа: Самостоятельное выполнение заданий по теме лекции	7
2.5	Визуализация с matplotlib	Рассмотрение объектного подхода к построению диаграмм и графиков различных типов. Самостоятельная работа: Самостоятельное выполнение заданий по теме лекции	7
2.6	Работа с файлами	Методы работы с файлами Самостоятельная работа: Самостоятельное выполнение заданий по теме лекции	7
2.7	Библиотека PyQt для разработки графического интерфейса	Использование библиотеки PyQt для создания графического интерфейса Самостоятельная работа: Самостоятельное выполнение заданий по теме лекции	7
2.8	Извлечение данных из веб-страниц	Извлечение данных из веб-страницы. Использовании модулей и сервисов, реализующих веб-запросы и облачную обработку. Самостоятельная работа: Самостоятельное выполнение заданий по теме лекции	8
2.9	Работа Python с API	Работа с API для доступа к социальным сетям и мессенджерам Самостоятельная работа: Самостоятельное выполнение заданий по теме лекции	7
	Итоговая аттестация	Зачет на основании совокупности выполненных домашних заданий и защиты проекта	16
	Итого:		144

3. Формы аттестации и оценочные материалы

3.1. Формы аттестации

Итоговая аттестация по программе является обязательной. Итоговая аттестация осуществляется на основании совокупности домашних заданий, выполненных на положительную оценку, и защиты проекта.

Форма итоговой аттестации: зачет на основании совокупности выполненных домашних заданий и защиты проекта.

Критерии оценивания:

- выполнение / не выполнение домашних заданий;
- выполнение / не выполнение итогового проекта.

Наименование модуля, разделов и тем	Основные показатели оценки	Формы и методы контроля и оценки
Разработчик на Python	ПК-1, ОПК-1	Зачет на основании совокупности выполненных домашних заданий и защиты проекта

Оценивание

таблица 7

Критерий	Соответствует	Не соответствует
выполнение / не выполнение домашних заданий	1 балл	0 баллов
выполнение / не выполнение итогового проекта	1 балл	0 баллов

Оценивание проводится преподавателем на основе представленных критериев и шкалы оценки.

Зачет выставляется слушателю, если по результатам оценивания слушатель получает 2 балла.

3.2. Оценочные материалы

таблица 8

Наименование модуля, разделов и тем	Основные показатели оценки	Формы и методы контроля и оценки	Вес задания
Возможности языка Python и его особенности	ПК-1, ОПК 1	Домашнее задание 1	30
Основы работы в Pandas	ПК-1, ОПК 1	Домашнее задание 2	30
Итоговая аттестация	ПК-1, ОПК 1	Проект	40

Примеры заданий

Практические задания

Домашнее задание №1 (по теме 1)

Воспользуйтесь кодом:

```
st = "Я помню чудное мгновенье:\n\
Передо мной явилась ты,\n\
Как мимолетное виденье,\n\
Как гений чистой красоты.\n\
\n\
В томленьях грусти безнадежной,\n\
В тревогах шумной суеты,\n\
```

Звучал мне долго голос нежный\n\
 И снились милые черты.\n\
 \n\
 Шли годы. Бурь порыв мятежный\n\
 Рассеял прежние мечты,\n\
 И я забыл твой голос нежный,\n\
 Твои небесные черты.\n\
 \n\
 В глуши, во мраке заточенья\n\
 Тянулись тихо дни мои\n\
 Без божества, без вдохновенья,\n\
 Без слез, без жизни, без любви.\n\
 \n\
 Душе настало пробужденье:\n\
 И вот опять явилась ты,\n\
 Как мимолетное виденье,\n\
 Как гений чистой красоты.\n\
 \n\
 И сердце бьется в упоенье,\n\
 И для него воскресли вновь\n\
 И божество, и вдохновенье,\n\
 И жизнь, и слезы, и любовь."

print(st)

1. Необходимо написать функцию, которая получает строку, заменяет в ней

- все знаки препинания и переходы строк на « »,
 - устраняет множественные пробелы,
- и возвращает эту строку.

2. Необходимо написать функцию, которая получает строку, разрезает её по переданному разделителю, возвращает список из подстрок – «слов», и количество подстрок.

3. Написать код, который использует описанные выше функции для создания упорядоченного по алфавиту словаря с частотой использованных слов. Словарь должен быть регистронезависимым («Без» и «без» - это одно слово)

Пример выполненного задания

```

def remove_everything(s):
# Эта функция заменяет в строке все знаки препинания на пробелы
# и убирает множественные пробелы
    s = s.replace("\n", " ")
    s = s.replace(".", " ")
    s = s.replace(",", " ")
  
```

```
s = s.replace(":", " ")
s = s.replace("; ", " ")
s = s.replace("-", " ")
s = s.replace(" ", " ")
s = s.replace(" ", " ")
s = s.replace(" ", " ")
```

```
# возвращаем изменённую строку
return s
```

```
def split_string(s):
```

```
# Эта функция "разрезает" строку по разделителю - знаку пробела
r = s.upper().split(" ")
```

```
# Возвращаем список из "слов" и их количество
return r, len(r)
```

```
st = "Я помню чудное мгновенье:\n\
Передо мной явилась ты,\n\
Как мимолетное виденье,\n\
Как гений чистой красоты.\n\
\n\
В томленьях грусти безнадежной,\n\
В тревогах шумной суеты,\n\
Звучал мне долго голос нежный\n\
И снились милые черты.\n\
\n\
Шли годы. Бурь порыв мятежный\n\
Рассеял прежние мечты,\n\
И я забыл твой голос нежный,\n\
Твои небесные черты.\n\
\n\
В глуши, во мраке заточенья\n\
Тянулись тихо дни мои\n\
Без божества, без вдохновенья,\n\
Без слез, без жизни, без любви.\n\
\n\
Душе настало пробужденье:\n\
И вот опять явилась ты,\n\
Как мимолетное виденье,\n\
Как гений чистой красоты.\n\
\n\
И сердце бьется в упоенье,\n\
И для него воскресли вновь\n\
\n"
```

```
И божество, и вдохновенье,\nИ жизнь, и слезы, и любовь."
```

```
print(st)
```

```
# Вызовем функцию, которая уберет знаки препинания
```

```
verse_string = remove_everything(st)
```

```
# Вызовем функцию, которая
```

```
verse_words, nwords = split_string(verse_string)
```

```
# Преобразуем список слов во множество - уберем дубликаты
```

```
words = set(verse_words)
```

```
# Множества не сортируются. Поэтому преобразуем множество обратно в список
```

```
# и отсортируем
```

```
words = sorted(list(words))
```

```
# Выведем статистику
```

```
print("\nСтихотворение содержит:\n\t", nwords, "слов\n\t", len(words), "уникальных\n\t", "слов\n")
```

```
# Создадим пустой словарь
```

```
d={ }
```

```
# В цикле for each переберем слова из списка
```

```
for word in words:
```

```
    # Заполним словарь
```

```
    # Ключ - слово, значение - количество вхождений
```

```
    d[word]=verse_words.count(word)
```

```
# Уберем "пустой символ", попавший случайно при "разрезании"
```

```
d.pop("")
```

```
# Посмотрим на результат
```

```
print(d)
```

Домашнее задание №2 (по теме 2)

Используйте файл data.csv в качестве источника данных.

Это данные о работе приёмной комиссии университета.

Вот структура объекта DataFrame:

```
Index(['№', 'ЛД', 'ФИО', 'Статус', 'Приоритет', 'Рейтинг', 'Факультет',
```

```
'Форма обучения', 'Период обучения', 'Срок обучения', 'Курс',  
'Код Специальности', 'Специальность', 'Образовательный уровень',  
'Комиссия', 'Тип док. об обр.', 'Серия', 'Номер', 'Ср. балл', 'Награды',  
'Оригинал', 'Комментарии', 'Бюджет', 'Общий конкурс', 'Целевой конкурс',  
'Дата создания', 'Тех. секретарь', 'Unnamed: 27'],  
dtype='object')
```

Столбец 'ЛД' содержит уникальный номер личного дела абитуриента. По уникальным записям этого столбца можно определить, сколько абитуриентов и в какое время обратились в приёмную комиссию.

Столбец 'Дата создания' содержит не только дату обращения, но и время. Абитуриенты, у которых дата и час совпадают, а различаются только минуты, обратились в приёмную комиссию в течение одного часа. Оцените нагрузку на приёмную комиссию.

Для этого, необходимо:

1. Преобразовать тип данных в столбце 'Дата создания' при помощи метода `pd.to_datetime`.
2. Добавить к данным два столбца: 'Date' и 'Hour', взяв данные из столбца 'Дата создания'.
3. Построить сводную таблицу, содержащую количество личных дел, принятых приёмной комиссией в течение каждого рабочего часа (`index = 'Hour', columns = 'Date'`).
4. Построить графическое представление данных в виде `heatmap` из библиотеки `SeaBorn`.

Пример выполненного задания

```
import numpy as np  
import pandas as pd  
import matplotlib as mpl  
import matplotlib.pyplot as plt  
import seaborn as sns  
  
data = pd.read_csv("data.csv", sep=';', encoding = "cp1251")  
  
df_dt = pd.to_datetime(data['Дата создания'], format="%d.%m.%Y %H:%M")  
df_dt = df_dt.dt.strftime('%d.%m.%y %H').str.split(" ", expand=True)  
  
df_dt.columns = ['Date', 'Hour']  
data = pd.concat([data, df_dt], axis = 1)  
  
fig = plt.figure(figsize=(8,6))  
h = data.pivot_table(index = 'Hour', columns = 'Date', values = 'ЛД', aggfunc = 'count')
```

```
hmp = sns.heatmap(h)
```

Проект

Загрузите файл salary_.csv. Кодировка: CP1251, разделители: «;».

В файле представлены данные о заработной плате профессорско-преподавательского состава университета:

'Т/Н' - табельный номер;

'Должность';

'Ставка' - часть ставки, на которую заключен трудовой договор;

'Код подразд.' - код подразделения по внутреннему классификатору;

'Оклад (ставка) по основному месту работы';

'Оклад (ставка) на работе по совместительству';

'Всего. руб.' - общая сумма заработной платы.

Обратите внимание, сотрудники идентифицируются по табельному номеру, на некоторых работников есть несколько записей – у них есть внутреннее совместительство, существуют сотрудники, у которых есть зарплата только по основному месту работы, есть основное и внутреннее совместительство, есть внешние совместители – у них нет оклада по основному месту работы в данной организации.

Необходимо написать программу на Python с использованием модулей Pandas и Rpyplot, которая позволяет получить следующую аналитику данных:

1. (1 балл) Выведите Фонд оплаты труда(ФОТ):

а) всей организации – float r1a;

б) в разрезе должностей – list r1b;

в) ФОТ организации без учёта внешних совместителей – float r1c;

г) ФОТ внешних совместителей – float r1d.

Ответ: list r1=[r1a, r1b, r1c, r1d]

2. (1 балл) Выведите данные о структуре персонала по должностям и постройте круговую диаграмму

Ответ: DataFrame r2 с columns=['Должность', 'Количество'],

Круговая диаграмма

3. (1 балл) Проанализируйте заработную плату ассистентов:

выведите минимальную (float r3a), среднюю (float r3b) и максимальную заработную плату (float r3c) по этой категории

Ответ: `list r3 = [r3a, r3b, r3c]`

4. (1 балл) Проанализируйте доцентов-внешних совместителей. Выведите таблицу с их группировкой по табельному номеру, представьте в таблице общую величину ставки и сумму заработной платы (DataFrame r4a с columns=['Т/Н', 'Ставка', 'Оклад']).

Выведите их численность с учётом группировки – int r4b.

Ответ: `list r4 = [r4a, r4b]`

5. (1 балл) Проанализируйте заработную плату работников (с учётом совместительства), работающих в данной организации по основному месту работы. Выведите минимальную (float r5a), среднюю (float r5b) и максимальную заработную плату (float r5c) по этой категории.

Ответ: `list r5 = [r5a, r5b, r5c]`

6. (1 балл) Проведите анализ ставок внешних совместителей: выведите минимальную (float r6a), среднюю (float r6b) и максимальную ставку (float r6c) по этой категории.

Ответ: `list r6 = [r6a, r6b, r6c]`

7. (1 балл) Выведите список доцентов, работающих на 1,0 ст.

Ответ: DataFrame r7 с columns=['Т/Н', 'Ставка']

8. (1 балл) Найдите самого незагруженного по основному месту работы сотрудника.

Ответ: DataFrame r8 с columns=['Т/Н', 'Ставка']

9. (2 балла) Постройте графики формирования ФОТ от меньших зарплат к большим для:

- а) всей организации;
- б) работников, чьё основное место работы находится в организации (с учётом внутреннего совместительства);
- в) внешних совместителей.

Ответ: 3 графика в одних осях

В качестве ответа на задания предоставить код программы в формате .py или .ipynb, формирующий структуру данных list r = [r1, r2, r3, r4, r5, r6, r7, r8], и 2 графика в одном полотне.

4. Организационно-педагогические условия реализации программы

4.1 Учебно-методическое обеспечение и информационное обеспечение программы

4.1.1. Список литературы

Основная литература:

Automate the Boring Stuff with Python [Электронный ресурс] – Режим доступа - <https://automatetheboringstuff.com/>

Dive Into Python 3 [Электронный ресурс] – Режим доступа - <http://diveintopython3.problemsolving.io/>

Problem Solving with Algorithms and Data Structures using Python [Электронный ресурс] – Режим доступа - <https://runestone.academy/runestone/static/pythonds/index.html>

Swaroop Chitlur. A Byte of Python [Электронный ресурс] – Режим доступа - <https://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.02.pdf> – 2020.

Федоров Д. Основы программирования на примере языка Python [Текст] : учебное пособие / Д. Федоров. - 2018.

Свейгарт Э. Автоматизация рутинных задач с помощью Python [Текст] : практическое руководство для начинающих / Эл Свейгарт. – 2019.

4.1.2. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)

Бэрри П. Изучаем программирование на Python [Текст] / П. Бэрри. – 2017.

Савельев В. Статистика и котики [Текст] / Владимир Савельев. – 2018.

Бослав С. Статистика для всех [Текст] / Сара Бослав. – 2015.

Хамидуллин Р. Я. Теория вероятностей и математическая статистика [Текст] / Р.Я. Хамидуллин. – 2020.

Справочник DAX функций для Power BI и Power Pivot [Электронный ресурс] / Антон Будуев – Режим доступа - <https://biprosto.ru/dax/functions.html>

Талер Р. Nudge. Архитектура выбора [Текст] / Ричард Талер. – 2017.

4.1.3. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)

Документация Postgres про сравнение строк <https://postgrespro.ru/docs/postgrespro/9.5/functions-matching>

Документация Postgres про другие функции работы со строками <https://postgrespro.ru/docs/postgrespro/9.5/functions-string>

Тестер регулярных выражений <https://www.regextester.com>

Интерактивный учебник по SQL <http://www.sql-tutorial.ru/ru/content.html>

Введение в анализ данных с помощью Pandas

Начало работы с Power BI - <https://docs.microsoft.com/ru-ru/power-bi/fundamentals/desktop-getting-started>

4.2. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

таблица 9

Наименование специализированных аудиторий, кабинетов, лабораторий	Вид занятий	Наименование оборудования, программного обеспечения
Аудитория	Лекция	Персональные компьютеры/ноутбуки, доступ в Интернет, принтеры, проектор, программное обеспечение Microsoft Visual Studio 2010 или выше
Аудитория	Практическая работа	Персональные компьютеры/ноутбуки, доступ в Интернет, принтеры, проектор, программное обеспечение Microsoft Visual Studio 2010 или выше
Аудитория	Итоговая аттестация	Персональные компьютеры/ноутбуки, доступ в Интернет, принтеры, проектор, программное обеспечение Microsoft Visual Studio 2010 или выше

5. Организация образовательного процесса

В таблице 9 описаны образовательные технологии.

Таблица 10

№ п/п	Вид занятия	Форма проведения занятий	Цель
1	Лекция	Ознакомление с теоретическими основами анализа данных	актуализация и систематизация теоретических знаний по дисциплине
2	Практическая работа	Выполнение заданий	осознание связей между теорией и практикой, повышение степени понимания материала

6. Составители программы


Тимохин Владимир Николаевич, д.э.н., профессор, специалист по УМР Центра развития ИТ-образования

Куклин Евгений Юрьевич, специалист по УМР Центра развития ИТ-образования

Созыкин Андрей Владимирович, к.т.н., заместитель директора Центра развития ИТ-образования

Согласовано
Эксперт отдела сопровождения образовательных программ

Согласовано
Директор ЦРИТО


Ж.И. Зубцова


А. В. Малеев



Федеральное государственное автономное
образовательное учреждение высшего образования
«Московский физико-технический институт
(национальный исследовательский
университет)»

Директору
по цифровизации образования
Д.И. Гриц

СЛУЖЕБНАЯ ЗАПИСКА

Рекомендую к реализации дополнительную профессиональную программу
повышения квалификации «Разработчик на Python», 144 ч.

Директор Центра развития ИТ-образования

А.В. Малеев



Федеральное государственное автономное
образовательное учреждение высшего образования
«Московский физико-технический институт
(национальный исследовательский
университет)»

Директору
по цифровизации образования
Д.И. Гриц

СЛУЖЕБНАЯ ЗАПИСКА

Прошу провести обучение по программе ПК ЦРИТО

№	Информация о программе и организаторе курса	Данные
1	Планируемое название дополнительной профессиональной программы (далее - ДПП)	Разработчик на Python
2	Вид ДПП (профессиональная переподготовка, свыше 250 ак.ч. или повышение квалификации, от 16 до 250 ак.ч.)	Повышение квалификации
3	Форма обучения (очно, заочно)	Очная (с применением дистанционных образовательных технологий)
4	Объем, в ак. ч.	144
5	Подразделение	ЦРИТО
6	Цель и целевая аудитория программы обучения	Целью реализации программы является формирование компетенций слушателей в области решения задач в Python, применения шаблонов проектирования на Python, работы с Python библиотеками. Целевая аудитория - лица, имеющие высшее или среднее профессиональное образование; студенты вузов
7	Сроки (период) обучения	По мере комплектования групп
8	Количество слушателей (список с электронными адресами в Приложении)	25
9	Источник финансирования и плановый размер финансовых средств	Субсидия на иные цели в рамках национального проекта «Демография»
10	ФИО и должность привлеченных к обучению лекторов	Тимохин Владимир Николаевич, д.э.н., профессор, специалист по УМР Центра развития ИТ-образования Куклин Евгений Юрьевич, специалист по УМР Центра развития ИТ-образования Созыкин Андрей Владимирович, к.т.н., заместитель директора Центра развития ИТ-образования

Исполнитель: руководитель проектов ЦРИТО Сырцова Е.Л.
email: syrtsova.el@mipt.ru Тел.: +7 (912) 709-64-78

Пояснительная записка
к разработке и реализации программы повышения квалификации
«Разработчик на Python»

Программа «Разработчик на Python»

направлена на решение следующих научно-образовательных задач:

формирование компетенций слушателей в области решения задач в Python, применения шаблонов проектирования на Python, работы с Python библиотеками.

Условия проведения:

Дистанционно – 12 недель

Оборудование: Персональные компьютеры/ноутбуки, доступ в Интернет, принтеры, проектор, программное обеспечение Microsoft Visual Studio 2010 или выше

Составители программы:

Тимохин Владимир Николаевич, д.э.н., профессор, специалист по УМР Центра развития ИТ-образования

Куклин Евгений Юрьевич, специалист по УМР Центра развития ИТ-образования

Созыкин Андрей Владимирович, к.т.н., заместитель директора Центра развития ИТ-образования

Директор Центра развития ИТ-образования



А.В. Малеев

Федеральное государственное автономное образовательное учреждение
высшего образования
«Московский физико-технический институт
(национальный исследовательский университет)»
(МФТИ, Физтех)

ВЫПИСКА ИЗ ПРОТОКОЛА № 6

заседания учебно-методического совета от 29 января 2024 года.

ПОВЕСТКА:

Рассмотрение дополнительных общеобразовательных и профессиональных программ.

Проректор по учебной работе А. А. Воронов

СЛУШАЛИ: заместителя директора (Центр дополнительного, дополнительного профессионального и онлайн-образования "Пуск") А. И. Рыбакову о представлении дополнительных общеобразовательных и профессиональных программ (Центр «Пуск», МФТИ).

ПОСТАНОВИЛИ:

Рекомендовать к утверждению в установленном порядке дополнительную профессиональную программу повышения квалификации «Разработчик на Python».

Решение принято единогласно.

Форма проведения заседания: заочная.

Председатель УМС МФТИ

А.А. Воронов

Ученый секретарь УМС МФТИ

М.В. Березникова

