

**Образовательное частное учреждение  
Дополнительного профессионального образования «Центр  
компьютерного обучения «Специалист» Учебно-научного центра при  
МГТУ им. Н.Э. Баумана»  
(ОЧУ «Специалист»)**

123317 Москва, Пресненская набережная, д 8, стр. 1, этаж 48, помещение 484с, комната 5  
ИНН 7701257303, ОГРН 1037739408189

Утверждаю:  
Директор ОЧУ «Специалист»



/Т.С.Григорьева/

«20» февраля 2018 года

**Дополнительная профессиональная программа  
повышения квалификации  
«Программирование на языке Python. Часть 2.  
Продвинутый курс»**

Программа разработана в соответствии с приказом Министерства образования и науки Российской Федерации от 1 июля 2013 г. N 499 "Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным профессиональным программам".

Повышение квалификации слушателей, осуществляемое в соответствии с программой, проводится с использованием модульного принципа построения учебного плана с применением различных образовательных технологий, в том числе дистанционных образовательных технологий и электронного обучения в соответствии с законодательством об образовании.

Дополнительная профессиональная программа повышения квалификации, разработана образовательной организацией в соответствии с законодательством Российской Федерации, включает все модули, указанные в учебном плане.

Содержание оценочных и методических материалов определяется образовательной организацией самостоятельно с учетом положений законодательства об образовании Российской Федерации.

Структура дополнительной профессиональной программы соответствует требованиям Порядка организации и осуществления образовательной деятельности по дополнительным профессиональным программам, утвержденного приказом Минобрнауки России от 1 июля 2013 г. N 499.

Объем дополнительной профессиональной программы вне зависимости от применяемых образовательных технологий, должен быть не менее 16 академических часов. Сроки ее освоения определяются образовательной организацией самостоятельно.

Формы обучения слушателей (очная, очно-заочная, заочная) определяются образовательной организацией самостоятельно.

К освоению дополнительных профессиональных программ допускаются:

- лица, имеющие среднее профессиональное и (или) высшее образование;
- лица, получающие среднее профессиональное и (или) высшее образование.

Для определения структуры дополнительной профессиональной программы и трудоемкости ее освоения может применяться система зачетных единиц. Количество зачетных единиц по дополнительной профессиональной программе устанавливается организацией.

Образовательная деятельность слушателей предусматривает следующие виды учебных занятий и учебных работ: лекции, практические и семинарские занятия, лабораторные работы, круглые столы, мастер-классы, мастерские, деловые игры, ролевые игры, тренинги, семинары по обмену опытом, выездные занятия, консультации, выполнение аттестационной, дипломной, проектной работы и другие виды учебных занятий и учебных работ, определенные учебным планом.

### 1. Цель программы

Предоставить слушателям полноценный комплекс знаний и навыков, необходимых для разработки программного обеспечения на языке Python.

#### Совершенствуемые компетенции

№	Компетенция	Направление подготовки		
		ФГОС	ВО	ПО
		НАПРАВЛЕНИЮ ПОДГОТОВКИ 09.03.02 «ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ»		

		(УРОВЕНЬ БАКАЛАВРИАТА)
		Код компетенции
1	способностью участвовать в работах по доводке и освоению информационных технологий в ходе внедрения и эксплуатации информационных систем	ПК-15
2	способность адаптировать приложения к изменяющимся условиям функционирования	ПК-32

**Совершенствуемые компетенции в соответствии с трудовыми функциями профессионального стандарта «ПРОГРАММИСТ», утвержденного приказом Минтруда и социальной защиты РФ от 18 ноября 2013 г. N 679н**

№	Компетенция	Направление подготовки
		Трудовые функции (код)
1	Разработка программного кода (Формализация и алгоритмизация поставленных задач, Написание программного кода с использованием языков программирования, определения и манипулирования данными, Оформление программного кода в соответствии с установленными требованиями)	A/01.3; A/02.3; A/03.3

### Планируемый результат обучения

После окончания обучения Слушатель будет знать:

- Основы объектно-ориентированного программирования
- Базовые элементы языка Python

После окончания обучения Слушатель будет уметь:

- создавать собственные классы;
- оценивать сложность алгоритма;
- использовать важнейшие стандартные структуры данных;
- создавать собственные структуры данных на основе стандартных.

## 2. Учебный план

Категория слушателей:

- Программисты
- Научные работники
- Специалисты по анализу данных и машинному обучению
- Преподаватели средней школы

Требования к предварительной подготовке: успешное окончание курса «Программирование на языке Python. Уровень 1. Базовый курс» или эквивалентная подготовка.

Срок обучения: 40 академических часов с преподавателем, 20 самостоятельно.

Форма обучения: очная, очно-заочная, заочная. По желанию слушателя форма обучения может быть изменена и/или дополнена.

Режим занятий: дневной, вечерний, группы выходного дня.

№ п/п	Наименование модулей по программе	Общая трудоемкость (акад. часов)	В том числе		СРС,ч	ПА*
			Лекций	Практических занятий		
1	Понятие класса. Инкапсуляция	6	2	2	2	Практ. (лаб) работа
2	Наследование	6	2	2	2	Практ. (лаб) работа
3	Полиморфизм и абстракция	6	2	2	2	Практ. (лаб) работа
4	Практическая работа	6	2	2	2	-
5	Алгоритмы сортировки и поиска	6	2	2	2	Практ. (лаб) работа
6	Алгоритмы поиска на графах	6	2	2	2	Практ. (лаб) работа
7	Структуры данных	6	2	2	2	Практ. (лаб) работа
8	Стандартные абстрактные типы данных	6	2	2	2	Практ. (лаб) работа
9	Хранение данных вне программы	6	2	2	2	Практ. (лаб) работа
10	Практическая работа	6	2	2	2	-
	Итого:	60	20	20	20	
	ПА* - Форма промежуточной аттестации					
	Итоговая аттестация	тестирование				

Для всех видов аудиторных занятий академический час устанавливается продолжительностью 45 минут. Форма промежуточной аттестации – см. п.3.3 в ЛНА «Положение о проведении промежуточной аттестации слушателей и осуществлении текущего контроля их успеваемости».

### 3. Календарный учебный график

Календарный учебный график формируется при осуществлении обучения в течение всего календарного года. По мере набора групп слушателей по программе составляется календарный график, учитывающий объемы лекций, практики, самоподготовки, выезды на объекты.

Неделя обучения	1	2	3	4	5	6	7	Итого часов
	пн	вт	ср	чт	пт	сб	вс	
1 неделя	8	8	8	8	8	-	-	40
СРС	4	4	4	4	4	-	-	20
Итого:								60
Примечание: ИА – Итоговая аттестация (тестирование)								

#### 4. Рабочие программы учебных предметов

##### Модуль 1. Понятие класса. Инкапсуляция

- Класс и экземпляр класса.
- Данные экземпляра, методы экземпляра и свойства экземпляра.
- Понятие атрибута.
- Практикум: Создание класса и его экземпляров.

##### Модуль 2. Наследование

- Принцип утиной типизации.
- Понятия базового класса и производного класса.
- Функция `isinstance` и ее применение.
- Практикум: Создание производного класса, применение экземпляров базового и производного класса.

##### Модуль 3. Полиморфизм и абстракция

- Подмена методов в производном классе.
- Понятие абстрактного класса.
- Практикум: Применение полиморфных классов.

##### Модуль 4. Практическая работа

##### Модуль 5. Алгоритмы сортировки и поиска

- Основные алгоритмы сортировки и поиска.
- Понятие сложности алгоритма.
- Практикум. Поиск информации по заданному критерию.

##### Модуль 6. Алгоритмы поиска на графах

- Поиск в ширину.
- Поиск в глубину.
- Практикум. Построение пути в графе.

##### Модуль 7. Структуры данных

- Понятие о структуре данных.
- Список и запись как основные структуры данных.
- Таблица и другие практически важные структуры данных.
- Практикум. Представление записей и таблиц в программе на языке Python.

##### Модуль 8. Стандартные абстрактные типы данных

- Контейнерные типы.
- Библиотечные модули `collections` и `collections.abc`
- Практикум. Схема «Компонент-Контейнер» и ее применение.

##### Модуль 9. Хранение данных вне программы

- Сохранение и восстановление данных. Библиотечный модуль `pickle`.
- Понятие об объектно-реляционном соответствии.
- Практикум. Применение модуля `pickle` для хранения и передачи данных.

##### Модуль 10. Практическая работа

## 5. Организационно- педагогические условия

Требования к кадровым условиям реализации дополнительной профессиональной программы:

а) преподавательский состав образовательной организации, обеспечивающий образовательный процесс, должен обладать высшим образованием и стажем преподавания по изучаемой тематике не менее 1 года и (или) практической работы в областях знаний, предусмотренных модулями программы, не менее 3 (трех) лет;

б) образовательной организацией наряду с традиционными лекционно-семинарскими занятиями должны применяться современные эффективные методики преподавания с применением интерактивных форм обучения, аудиовизуальных средств, информационно-телекоммуникационных ресурсов и наглядных учебных пособий.

Требования к материально-техническому и учебно-методическому обеспечению дополнительной профессиональной программы:

а) образовательная организация располагает необходимой материально-технической базой, включая современные аудитории, библиотеку, аудиовизуальные средства обучения, мультимедийную аппаратуру, оргтехнику, копировальные аппараты. Материальная база соответствует санитарным и техническим нормам и правилам и обеспечивает проведение всех видов практической и дисциплинарной подготовки слушателей, предусмотренных учебным планом реализуемой дополнительной профессиональной программы.

б) в случае применения электронного обучения, дистанционных образовательных технологий каждый обучающийся в течение всего периода обучения должен быть обеспечен индивидуальным неограниченным доступом к электронной информационно-образовательной среде, содержащей все электронные образовательные ресурсы, перечисленные в модулях дополнительной профессиональной программы.

## 6. Формы аттестации и оценочные материалы

Образовательная организация несет ответственность за качество подготовки слушателей и реализацию дополнительной профессиональной программы в полном объеме в соответствии с учебным планом.

Оценка качества освоения дополнительной профессиональной программы слушателей включает текущий контроль успеваемости и итоговую аттестацию.

Конкретные формы и процедуры текущего контроля успеваемости, промежуточной аттестации и итоговой аттестации слушателей устанавливаются образовательной организацией самостоятельно.

Слушателям, успешно освоившим дополнительную профессиональную программу и прошедшим итоговую аттестацию, выдается удостоверение о повышении квалификации.

Слушателям, не прошедшим итоговой аттестации или получившим на итоговой аттестации неудовлетворительные результаты, а также лицам, освоившим часть дополнительной профессиональной программы и (или) отчисленным из образовательной организации, выдается справка об обучении или о периоде обучения по образцу, самостоятельно устанавливаемому образовательной организацией.

Итоговая аттестация проводится по форме тестирования в соответствии с учебным планом.

### Промежуточная аттестация

№	Тематика практического занятия	Форма ПА
1	На усмотрение преподавателя	Практическая работа (лабораторная работа №1)
2	На усмотрение преподавателя	Практическая работа (лабораторная работа №2)
3	На усмотрение преподавателя	Практическая работа

		(лабораторная работа №3)
4	На усмотрение преподавателя	Практическая работа (лабораторная работа №4)
5	На усмотрение преподавателя	Практическая работа (лабораторная работа №5)
6	На усмотрение преподавателя	Практическая работа (лабораторная работа №6)
7	На усмотрение преподавателя	Практическая работа (лабораторная работа №7)
8	На усмотрение преподавателя	Практическая работа (лабораторная работа №8)

Примеры практических задач, которые могут быть рассмотрены на промежуточной аттестации. Список не является исчерпывающим. Конкретные темы определяются преподавателем по согласованию со слушателями.

- Поиск маршрута между двумя пунктами.
- Обработка XML-документа.
- Комплектование и обработка заказа.
- Игра в шахматы (программная часть).
- Веб-скрейпинг.
- Составление расписания учебных занятий.
- Написание игрового сервера.
- Расчет простой электрической сети постоянного тока.
- Обработка видео с использованием внешнего ПО.

### **Итоговая аттестация**

#### **Вопрос 1/10**

В чем состоит принцип инкапсуляции?

Выберите один ответ:

- Содержимое файла не может быть прочитано, если файл не открыт.
- Нет необходимости знать особенности внутренней реализации объекта, чтобы пользоваться его свойствами и методами.
- Данные объекта описываются отдельно от его методов
- Метод объекта не может быть вызван, если не заданы начальные условия
- Каждый класс должен быть описан в отдельном файле

#### **Вопрос 2/10**

Что такое конструктор?

Выберите один ответ:

- Специальный метод, который автоматически вызывается всякий раз, когда создается новый объект этого класса.
- Функция, возвращающая новосозданный экземпляр класса.
- Часть интерпретатора, создающая новые классы.
- Функция, возвращающая другую функцию.
- Все приведенные ответы неверны

### Вопрос 3/10

Что будет выведено, если пользователь введет отрицательное число?

```
class SignError( ValueError ) : pass
```

```
try:
    A = input( u': ' )
    if A < 0 : raise SignError
except ValueError :
    print u'Неверное значение'
except SignError :
    print u'Неверный знак числа'
except Exception :
    print u'Неизвестная ошибка'
else:
    print u'Число введено правильно'
```

Выберите один ответ:

- Неверное значение
- Неверный знак числа
- Неизвестная ошибка
- Число введено правильно
- Будет выведено сообщение об ошибке из-за неперехваченного исключения

### Вопрос 4/10

Найдите ошибку в следующем фрагменте.

```
class Transport( object ) :
    year = property( lambda self : self.__Year )
```

```
class Auto ( Transport ) :
    pass
```

```
X = Auto()
X.year = 1997
print X.year
```

Выберите один ответ:

- В классе `Auto` отсутствует свойство `year`
- Значение свойства `year` нельзя выводить на печать
- Свойство `year` работает только на чтение — ему нельзя присваивать значение.
- Класс `Auto` является абстрактным — создавать его экземпляры нельзя.
- В этом фрагменте нет ошибок

### Вопрос 5/10

Можно ли использовать экземпляр класса `User` там, где по контексту требуется экземпляр класса `Auto`?

```
class Auto( object ) :
    def save( self ) :
        ...
```

```
class User ( object ) :
    def save( self ) :
        ...
```

Выберите один ответ:

- Нельзя, так как класс `User` не является производным от класса `Auto`
- Нельзя ни при каких обстоятельствах.
- Можно в любом случае.
- Можно при условии, что в этом контексте не вызываются никакие другие методы, кроме `save()`.
- Все приведенные ответы неверны

### Вопрос 6/10

Что такое файлоподобный объект?

Выберите один ответ:

- Любой объект, осуществляющий запись в файл.
- Любой объект, осуществляющий чтение из файла.
- Объект, содержащий метод `close` и хотя бы один из методов `read()` или `write()`.
- Объект, содержащий любые методы, считывающие и/или записывающие данные.
- Объект, осуществляющий запись в базу данных

### Вопрос 7/10

Можно ли в производном классе повторно создать метод, который уже был создан в базовом классе?

Выберите один ответ:

- Нельзя
- Можно без ограничений
- Можно, при условии, что такая возможность предусмотрена в базовом классе.
- Можно при условии, что новый метод будет иметь те же параметры, что и старый.
- Можно без ограничений, но если новый метод будет иметь другие параметры, это приведет к ошибке в дальнейшем.

### Вопрос 8/10

Можно ли использовать числа с плавающей точкой для работы с денежными суммами?

Выберите один ответ:

- Нельзя, так как денежная сумма всегда представлена точным числом.
- Нельзя, так как число с плавающей точкой имеет слишком низкую точность.
- Можно без ограничений
- Можно при условии, что денежная сумма не отрицательна.
- Можно при условии, что денежная сумма не превосходит некоторой определенной величины.

### Вопрос 9/10

Найдите ошибку в следующем фрагменте:

```
def step( C ) :  
    return C // 10.0  
  
for K in step(20.0) :  
    print K
```

Выберите один ответ:

- Функцию `step()` нельзя использовать в цикле после слова `in`, так как она не возвращает списка и не является генератор-функцией
- Тело цикла никогда не будет выполнено
- Значение переменной `K` нельзя выводить на печать
- Фактический параметр функции `step()` не должен быть с плавающей точкой.

- В этом фрагменте нет ошибок

### **Вопрос 10/10**

Можно ли использовать экземпляр производного класса там, где по контексту требуется экземпляр базового класса?

Выберите один ответ:

- Нельзя ни при каких условиях.
- Можно без ограничений.
- Можно при условии, что базовый класс не абстрактный.
- Можно при условии, что производный класс не абстрактный.
- Можно при условии, что производный класс не имеет других базовых классов.