

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

123242, город Москва, улица Зоологическая, дом 11, строение 2, помещение I, комната 11
ИНН 7701257303, ОГРН 1037739408189

Утверждаю:
Директор ОЧУ «Специалист»



/Т.С.Григорьева/
«01» июня 2018 года

**Дополнительная профессиональная программа
повышения квалификации
«Java SE10. Уровень 1. Основы
программирования»**

город Москва

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

Овладеть объектно-ориентированным языком программирования Java Standard Edition 10 (Java SE10), научиться использовать интегрированную среду разработки и приобрести базовые навыки создания объектно-ориентированных приложений.

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

Лица, успешно освоившие программу, должны овладеть следующими компетенциями:
Теоретическое и практическое освоение синтаксиса языка программирования Java.

После окончания обучения Слушатель будет знать:
Язык программирования Java SE10.

После окончания обучения Слушатель будет уметь:

- разбираться в Java технологиях;
- работать с примитивными типами;
- использовать ветвление и циклы;
- создавать методы на Java;
- работать с массивами и со строками;
- разрабатывать классы на Java;
- использовать наследование и полиморфизм;
- разрабатывать абстрактные классы и интерфейсы;
- обрабатывать ошибки, возникающие в программе;
- работать с файловой системой;
- использовать новые возможности классов из пакета NIO 2;
- пользоваться коллекциями Java;
- использовать в программе параметризованные типы;
- применять знания о паттернах проектирования.

2. Учебный план

Категория слушателей:

- Начинающие программисты
- Опытные программисты, желающие повысить уровень квалификации

Требования к предварительной подготовке: «Программирование на языке С (Си)».

Срок обучения: 40 академических часов с преподавателем.

Самостоятельные занятия: предусмотрены.

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	
			Лекций	Практических занятий
1	Введение в Java технологии	2	1	1
2	Введение в язык программирования Java	3	2	1
3	Методы и операторы Java	2	1	1
4	Создание и использование массивов Java	3	1	2
5	Работа со строками в Java	2	1	1
6	Разработка классов на Java	3	1	2
7	Наследование и полиморфизм	2	1	1
8	Интерфейсы и аннотации	3	2	1
9	Пакет java.lang	2	1	1
10	Обработка ошибок в Java	3	1	2
11	Потоки данных в Java	2	1	1
12	Работа с файловой системой в NIO 2	3	2	1
13	Пакет java.util	2	1	1
14	Коллекции в Java	2	1	1
15	Вложенные классы в Java	2	1	1
16	Лямбда-выражения	2	1	1
17	Паттерны проектирования	2	1	1
	Итого:	40	20	20
	Итоговая аттестация	тестирование		

Для всех видов аудиторных занятий академический час устанавливается продолжительностью 45 минут.

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 ИА	-	-	40
Итого:								40
Примечание: ИА – Итоговая аттестация (тестирование)								

4. Рабочие программы учебных предметов

Модуль 1. Введение в Java технологии

- Особенности языка программирования Java.
- Описание Java технологий.
- Использование интегрированной среды разработки.

Практика:

- Установка JDK/IDE, настройка параметров среды.
- Работа в командной строке.

Модуль 2. Введение в язык программирования Java

- Языковые лексемы Java.
- Введение в систему типов языка Java.
- Работа с примитивными типами и константами.
- Операции языка Java.
- Преобразование простых типов.

Модуль 3. Методы и операторы Java

- Создание и вызов методов.
- Перегрузка и методы с переменным числом аргументов.
- Операторы ветвления.
- Операторы циклов.

Модуль 4. Создание и использование массивов Java

- Одномерные массивы.
- Многомерные массивы.
- Работа с массивами и класс Arrays.

Практика:

- Конвертор дня года в месяц и число.

Модуль 5. Работа со строками в Java

- Основные методы класса String.

- Сложение и преобразование строк.
- Классы динамических строк.
- Ввод данных с клавиатуры и класс Scanner.
- Регулярные выражения.

Практика:

- Конвертор целых чисел в двоичную систему.

Модуль 6. Разработка классов на Java

- Обзор основных концепций ООП.
- Объявление класса.
- Члены класса и создание объектов класса.
- Модификаторы доступа.
- Модификаторы final & static.
- Использование пакетов, директив импорта и переменной среды CLASSPATH
- Модули Java, как единица развёртывания и безопасности

Практика:

- Разбор параметров командной строки

Модуль 7. Наследование и полиморфизм

- Наследование как механизм повторного использования кода.
- Конструктор при наследовании
- Преобразование типов и операция instanceof.
- Виртуальные методы и позднее связывание.
- Абстрактные классы и методы.

Практика:

- Разбор параметров командной строки. Часть 2. Наследование

Модуль 8. Интерфейсы и аннотации

- Концепция интерфейсов.
- Объявление интерфейса.
- Реализация интерфейса.
- Статические методы и методы по умолчанию в интерфейсах.
- Использование и создание аннотаций.

Практика:

- Сортировка массива объектов

Модуль 9. Пакет java.lang

- Класс Object и переопределение его методов.
- Метаданные и рефлексия.

- Классы System и Math.

Модуль 10. Обработка ошибок в Java

- Концепция исключений в Java.
- Использование операторов try, catch и finally.
- Проверяемые и непроверяемые исключения.
- Создание своих классов исключений.
- Оператор try для освобождения ресурсов.

Модуль 11. Поток данных в Java

- Обзор классов потоков.
- Работа с байтовыми потоками.
- Работа с потоками символов.
- Использование класса java.io.File.
- Сжатие файлов.
- Сериализация объектов в Java.

Практика:

- Частотный анализ слов в тексте

Модуль 12. Работа с файловой системой в NIO 2

- Использование интерфейса Path.
- Работа с атрибутами файлов.
- Основные возможности класса Files.
- Использование класса Files для обхода дерева каталогов.
- Мониторинг изменений в файловой системе.

Модуль 13. Пакет java.util

- Форматирование данных.
- Работа с датой и временем.
- Класс Locale и глобализация кода.
- Локализация и класс ResourceBundle.
- Генерация псевдослучайных чисел

Модуль 14. Коллекции в Java

- Иерархия классов коллекций.
- Концепция параметризованных типов данных.
- Работа с параметризованным методом и интерфейсом.
- Обзор возможностей списков, множеств и словарей в Java.

Модуль 15. Вложенные классы в Java

- Внутренние классы.
- Вложенные классы.
- Анонимные классы.
- Перечисления в Java.

Модуль 16. Лямбда-выражения

- Синтаксис лямбда-выражений.
- Ссылки на методы.
- Функциональные интерфейсы.

Модуль 17. Паттерны проектирования

- Обзор паттернов.
- Паттерн одиночка.
- Паттерн композиция.
- Паттерн наблюдатель.

Практика:

- Реализация паттерна наблюдатель.

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.

Вопрос 1

Какие имена переменных допустимы?

Выберите несколько ответов:

- Object
- int
- 7short
- short7
- _int

Вопрос 2

Какие объявления переменных допустимы?

Выберите несколько ответов:

- int x;
- boolean b1=true, b2=false;
- float y=7.0;
- short z=1.0;
- char c="\u042F";
- char c="A";

Вопрос 3

Какие объявления переменных допустимы?

Выберите несколько ответов:

- String s=new String("Hello");
- String s="Hello";
- Float f=5.2;
- Object o=new String("Hello");

- `String s="\u0426\u041A\u041E";`
- `String s=new String(75);`

Вопрос 4

Какие слова являются зарезервированными словами языка Java?

Выберите несколько ответов:

- `switch`
- `throws`
- `throw`
- `implement`
- `super`
- `synchronized`

Вопрос 5

Какие слова являются зарезервированными словами языка Java?

Выберите несколько ответов:

- `final`
- `friendly`
- `transient`
- `new`
- `old`
- `begin`
- `destroy`

Вопрос 6

Сколько байт оперативной памяти занимает число типа `long`?

Выберите один ответ:

- 2
- 4
- 6
- 8
- Объем выделяемой памяти зависит от компилятора

- Объем выделяемой памяти зависит от интерпретатора

Вопрос 7

Сколько байт оперативной памяти занимает число типа `double`?

Выберите один ответ:

- 2
- 4
- 6
- 8
- Объем выделяемой памяти зависит от компилятора
- Объем выделяемой памяти зависит от интерпретатора

Вопрос 8

Сколько бит оперативной памяти занимает число типа `int`?

Выберите один ответ:

- 4
- 8
- 16
- 32
- Объем выделяемой памяти зависит от компилятора
- Объем выделяемой памяти зависит от интерпретатора

Вопрос 9

Сколько бит оперативной памяти занимает переменная типа `char`?

Выберите один ответ:

- 4
- 8
- 16
- 32
- Объем выделяемой памяти зависит от компилятора
- Объем выделяемой памяти зависит от интерпретатора

Вопрос 10

Каково максимальное значение для переменной типа byte?

Выберите один ответ:

- 127
- 128
- 256
- зависит от компилятора
- зависит от интерпретатора

Вопрос 11

В программе объявлены три переменные: `int x=5; double y=7.1; short z=255;` Какие преобразования верны?

Выберите несколько ответов:

- `y=x;`
- `z=x;`
- `x=z;`
- `z=(short)(int)y;`
- `z=(double)(short)y;`

Вопрос 12

Какие формы записи оператора условного перехода верны?

Выберите несколько ответов:

- `if (x<y) { System.out.println("x<y"); }`
- `if x<y { System.out.println("x<y"); }`
- `if x<y System.out.println("x<y");`
- `if (x<y) System.out.println("x<y");`
- `if x<y then System.out.println("x<y");`

Вопрос 13

Какой результат появится на экране после выполнения следующей программы?

```
public class My{  
    public static void main(String arg[]){  
        char a='b';  
        switch (a){  
            case 'a':System.out.print('a');  
            case 'b':System.out.print('b');  
            case 'c':System.out.print('c');  
            default :System.out.print('d');  
        }  
    }  
}
```

Выберите один ответ:

- b
- d
- ab
- bc
- bcd
- abcd

Вопрос 14

Какой результат появится на экране после выполнения следующей программы?

```
public class My {
    public static void main(String arg[]) {
        for(int i=0;i<=2;i++){
            for(int j=0;j<=2;j++){
                System.out.print(j);
                if(j==1)break;
            }
        }
    }
}
```

Выберите один ответ:

- 0
- 01
- 000
- 010101
- программа не откомпилируется

Вопрос 15

Какой результат появится на экране после выполнения следующей программы?

```
class First {
    int sum(int x,int y){
        return(x+y);
    }
    double sum(double x,double y){
        return(x+y+1);
    }
}
public class My {
    public static void main(String arg[]){
        System.out.println((new First()).sum(5,6.0));
    }
}
```

Выберите один ответ:

- 11
- 11.0
- 12
- 12.0
- программа не откомпилируется

Вопрос 16

Какой результат появится на экране после выполнения следующей программы?

```
public class My {
    public static void main() {
        System.out.print('a');
    }
    public static void main(String arg) {
```

```
System.out.print('b');<br>    }<br>    public    static    void    main(String    arg[]){<br>System.out.print('c');<br> }<br>}<br>
```

Выберите один ответ:

- программа не откомпилируется
- программа откомпилируется, но не запустится
- a
- b
- c
- abc

Вопрос 17

Какие спецификаторы доступа можно использовать при объявлении класса?

Выберите несколько ответов:

- public
- protected
- private
- final
- friendly
- package

Вопрос 18

Какие спецификаторы доступа можно использовать при объявлении полей класса?

Выберите несколько ответов:

- public
- protected
- private
- final
- friendly
- static

Вопрос 19

Какие результаты можно увидеть на экране при попытке компиляции и запуска следующей программы?

```
private class My {  
    public static void main(String a[]) {  
        System.out.println("Hello!");  
    }  
}
```

Выберите один ответ:

- Hello!
- Программа не запустится из-за отсутствия метода main
- Программа не откомпилируется из-за недопустимого спецификатора доступа в описании класса
- Программа не откомпилируется из-за того, что компилятор не найдет классы System и String

Вопрос 20

Какие результаты можно увидеть на экране при попытке компиляции и запуска следующей программы?

```
final class My {  
    public static void main(String a[]) {  
        System.out.println("Hello!");  
    }  
}
```

Выберите один ответ:

- Hello!
- Программа не запустится из-за отсутствия метода main
- Программа не откомпилируется из-за недопустимого спецификатора доступа в описании класса
- Программа не откомпилируется из-за того, что компилятор не найдет классы System и String