



АВТОНОМНАЯ НЕКОММЕРЧЕСКАЯ ОРГАНИЗАЦИЯ  
**НАЦИОНАЛЬНЫЙ ЦЕНТР  
ИННОВАЦИЙ В ОБРАЗОВАНИИ**  
ИНН 7708241976, КПП 770801001, ОГРН 1147799018696

107045, Россия, Москва, ул. Сретенка, д. 24/2, стр. 1, Тел: +7(495)114-56-28, www.ncio.ru, E-mail: info@ncio.ru



**Программа  
дополнительного профессионального образования  
(повышение квалификации)**

**«Основы программирования на языке JavaScript»**

Автор: А.А. Герасимова,

инженер-разработчик ООО «Экзамен-Технолаб»

Москва, 2023 г.

## Тема: «Основы программирования на языке JavaScript»

### Раздел 1. «Характеристика программы»

#### 1.1. Цель реализации программы

**Цель:** совершенствование профессиональных компетенций слушателей в области изучения основ программирования на языке JavaScript для использования в образовательном процессе.

#### 1.2. Совершенствуемые компетенции

№ п/п	Компетенции	Направление подготовки Педагогическое образование 44.03.01 Бакалавриат
		Код компетенции
1.	Способен осуществлять педагогическую деятельность на основе специальных научных знаний	ОПК-8

#### 1.3. Планируемые результаты

№ п/п	Уметь-Знать	Направление подготовки Педагогическое образование 44.03.01 Бакалавриат
		Код компетенции
1.	<b>Уметь:</b> создавать алгоритмы работы программ; писать программный код на языке JavaScript; решать алгоритмические задачи с помощью языка JavaScript. <b>Знать:</b> принципы написания программного кода; синтаксис, структуры и инструменты, область применения и особенности языка JavaScript.	ОПК-8
2.	<b>Уметь:</b> разрабатывать учебные занятия по изучению основ программирования на языке JavaScript <b>Знать:</b> алгоритм разработки учебных занятий по	ОПК-8

изучению основ программирования на языке JavaScript	
---	--

#### 1.4. Категория слушателей

*Уровень образования* – высшее образование;

*Направление подготовки* – педагогическое образование;

*Область профессиональной деятельности* – обучение на уровне общего образования, дополнительного образования.

**1.5. Форма обучения:** Заочная с ДОТ и ЭО.

**1.6. Режим занятий:** 3 раза в неделю на протяжении 8 недель.

**1.7. Трудоёмкость программы:** 29 часов.

## Раздел 2. «Содержание программы»

### 2.1. Учебный план

№ п/п	Наименование разделов/тем	Всего, внеаудит. часов	Внеаудиторные учебные занятия		Форма контроля	Трудоёмкость
			Онлайн-лекция	Практ. занятия		
1.	Раздел 1. Знакомство с JavaScript.	9	3	6		9
1.1.	Область применения и особенности языка.	1	1			1
1.2.	Примитивы и базовые операторы.	3	1	2	Тест № 1	3
1.3.	Условные операторы.	5	1	4	Тест № 2 Проект № 1	5
2.	Раздел 2. Базовый JavaScript.	11	3	8		11

2.1.	Циклы.	3	1	2	Тест № 3	3
2.2.	Массивы и их методы.	3	1	2	Тест № 4	3
2.3.	Функции.	5	1	4	Тест № 5 Проект № 2	5
3.	<b>Раздел 3. Решение алгоритмических задач.</b>	<b>5</b>	<b>1</b>	<b>4</b>		<b>5</b>
3.1.	Как решать задачи? Разбор примеров и решение задач.	5	1	4	Проект № 3	5
4.	<b>Раздел 4. Разработка учебных занятий по основам языка JavaScript.</b>	<b>3</b>	<b>1</b>	<b>2</b>		<b>3</b>
4.1.	Построение учебных занятий по изучению основ языка JavaScript	1	1			1
4.2.	Разработка учебного занятия по изучению основ языка JavaScript.	2		2	Проект № 4	2
5.	<b>Итоговая аттестация.</b>	<b>1</b>		<b>1</b>	Зачёт по совокупности выполненных работ и итогового тестирования	<b>1</b>
<b>Итого:</b>		<b>29</b>	<b>8</b>	<b>21</b>		<b>29</b>

## 2.2. Календарный учебный график

Программа повышения квалификации реализуется в течение двух месяцев по мере комплектования группы в соответствии с календарным учебным графиком. Продолжительность реализации программы по графику – восемь недель с момента укомплектованности группы.

Наименование разделов/тем	Объём нагрузки, часов	Учебные недели			
		1 неделя	2 неделя	3 неделя	4 неделя

<b>Раздел 1. Знакомство с JavaScript.</b>	9	2 часа (онлайн-лекция), 2 часа (практика)	1 час (онлайн-лекция), 4 часа (практика)	-	-
1.1. Область применения и особенности языка.	1	1 час (онлайн-лекция)			
1.2. Прimitives и базовые операторы.	3	1 час (онлайн-лекция), 2 часа (практика)			
1.3. Условные операторы.	5		1 час (онлайн-лекция), 4 часа (практика)		
<b>Раздел 2. Базовый JavaScript.</b>	11	-	-	1 час (онлайн-лекция), 2 часа (практика)	1 час (онлайн-лекция), 2 часа (практика)
2.1. Циклы.	3			1 час (онлайн-лекция), 2 часа (практика)	
2.2. Массивы и их методы.	3				1 час (онлайн-лекция), 2 часа (практика)

Наименование разделов/тем	Объём нагрузки, часов	Учебные недели			
		5 неделя	6 неделя	7 неделя	8 неделя
<b>Раздел 2. Базовый JavaScript.</b>	11	1 час (онлайн-лекция), 2 часа (практика)	2 часа (практика)	-	-
2.3. Функции.	5	1 час (онлайн-лекция), 2 часа (практика)	2 часа (практика)		
<b>Раздел 3. Решение алгоритмических задач.</b>	5	-	-	1 час (онлайн-лекция), 4 часа (практика)	-
3.1. Как решать задачи? Разбор	5			1 час (онлайн-лекция),	

примеров и решение задач.				4 часа (практика)	
<b>Раздел 4. Разработка учебных занятий по основам языка JavaScript.</b>	3	-	-	-	1 час (онлайн-лекция), 2 часа (практика)
4.1. Построение учебных занятий по изучению основ языка JavaScript	1				1 час (онлайн-лекция)
4.2. Разработка учебного занятия по изучению основ языка JavaScript.	2				2 часа (практика)
<b>5. Итоговая аттестация.</b>	1	-	-	-	1 час (практика)

### 2.3. Рабочая программа

№ п/п	Наименование разделов, тем	Виды учебных занятий, час	Содержание
<b>Раздел 1. Знакомство с JavaScript.</b>			
1.1.	Область применения и особенности языка.	Онлайн-лекция – 1 час	Использование JavaScript в веб-разработке. Особенности языка. Среда разработки.
1.2.	Примитивы и базовые операторы.	Онлайн-лекция – 1 час	Переменные. Знакомство с математическими и логическими операторами. Типы данных (целые, вещественные, логические).

		Практическое занятие – 2 час	Практика на представленных в лекции примерах по примитивам и базовым операторам.  Тест № 1. Выполнение тестирования по примитивам и базовым операторам.
1.3.	Условные операторы.	Онлайн-лекция – 1 час	Типы условных операторов. Ветвления. Операторы с дополнительным условием. Оператор выбора. Тернарный оператор.
		Практическое занятие – 4 часа	Практика на представленных в лекции примерах по условным операторам.  Тест № 2. Выполнение тестирования по условным операторам.  Проект №1. Написание программы для решения задачи с использованием условного оператора.
<b>Раздел 2. Базовый JavaScript.</b>			
2.1.	Циклы.	Онлайн-лекция – 1 час	Знакомство с циклами. Циклы с постусловием и предусловием. Цикл с шагом.
		Практическое занятие – 2 час	Практика на представленных в лекции примерах по циклам.  Тест № 3. Выполнение тестирования по циклам.
2.2.	Массивы и их методы.	Онлайн-лекция – 1 часа	Массив. Определение и область применения. Массивы и строки. Методы перебора. Сортировка, отбор и сведение.
		Практическое занятие – 2 часа	Практика на представленных в лекции примерах по массивам и их методам.  Тест № 4. Выполнение тестирования по массивам и их методам.

2.3	Функции.	Онлайн-лекция – 1 часа	Функция и ее область видимости. Способы создания функции. Стрелочные функции.
		Практическое занятие – 4 часа	Практика на представленных в лекции примерах по функциям. Тест № 5. Выполнение тестирования по функциям. Проект №2. Написание программы для решения задачи с использованием циклов и функций.
<b>Раздел 3. Решение алгоритмических задач.</b>			
3.1.	Как решать задачи? Разбор примеров и решение задач.	Онлайн-лекция – 1 час	Алгоритм решения задач. Разбор решения алгоритмических задач.
		Практическое занятие – 4 часа	Практика на представленных в лекции примерах. Проект №3. Написание программы для решения алгоритмической задачи с использованием изученных структур языка JavaScript.
<b>Раздел 4. Разработка учебных занятий по основам языка JavaScript.</b>			
4.1.	Построение учебных занятий по изучению основ языка JavaScript	Онлайн-лекция – 1 час	Особенности построения учебных занятий по изучению основ языка JavaScript
4.2.	Разработка учебного занятия по изучению основ языка JavaScript.	Практическое занятие – 2 часа	Проект № 4 Разработка учебного занятия по изучению основ языка JavaScript (тема по выбору обучающихся).
5.	<b>Итоговая аттестация</b>	Практическое занятие – 1 час	Зачёт по совокупности выполненных работ и итогового тестирования.



### Раздел 3. «Формы аттестации и оценочные материалы»

Программой предусмотрены: промежуточный контроль – тесты и проекты; итоговая аттестация – итоговое тестирование, зачёты промежуточного контроля.

#### 3.1. Промежуточный контроль

##### Проекты:

**Проект №1:** «Написание программы для решения задачи с использованием условного оператора»

**Требования к выполнению проекта:** написать программный код на языке JavaScript, который будет соответствовать решению задачи, предлагаемой преподавателем. Ответ должен быть представлен в виде ссылки на решение.

##### Критерии оценивания:

1. Ответ представлен в требуемом виде.
2. Алгоритм решения задачи оптимален.
3. В программном коде используется хотя бы один условный оператор.
4. Задача решена в полной мере.

##### Оценивание:

Зачтено	Не зачтено
Выполнены все пункты оценочных материалов.	Не выполнен хотя бы один из оценочных пунктов.

**Проект №2:** «Написание программы для решения задачи с использованием циклов и функций»

**Требования к выполнению проекта:** написать программный код на языке JavaScript, который будет соответствовать решению задачи, предлагаемой преподавателем. Ответ должен быть представлен в виде ссылки на решение.

##### Критерии оценивания:

1. Ответ представлен в требуемом виде.
2. Алгоритм решение задачи оптимален.
3. В программном коде используется хотя бы один цикл и функцию.
4. Задача решена в полной мере.

##### Оценивание:

Зачтено	Не зачтено
Выполнены все пункты оценочных	Не выполнен хотя бы один из оценочных

материалов.	пунктов.
-------------	----------

**Проект №3:** «Написание программы для решения алгоритмической задачи с использованием изученных структур языка JavaScript»

**Требования к выполнению проекта:** написать программный код на языке JavaScript, который соответствует решению алгоритмической задачи, предлагаемой преподавателем. Ответ должен быть представлен в виде ссылки на решение.

**Критерии оценивания:**

1. Ответ представлен в требуемом виде.
2. Алгоритм решение задачи оптимален.
3. Задача решена в полной мере.
4. Все тест-кейсы выполнены успешно.

**Оценивание:**

Зачтено	Не зачтено
Выполнены все пункты оценочных материалов.	Не выполнен хотя бы один из оценочных пунктов.

**Проект №4:** «Разработка учебного занятия по изучению основ языка JavaScript»

**Требования к выполнению проекта:** составить учебное занятие (тему выбирает слушатель) по изучению основ языка JavaScript, в которое должны быть включены, изученные технологии. Ответ должен быть представлен в виде файла в формате doc, docx или pdf.

**Критерии оценивания:**

1. Ответ представлен в требуемом виде.
2. Представлено учебное занятие по изучению основ языка JavaScript.
3. Содержание занятия соответствует одной или нескольким технологиям, изученным на занятиях пройденного учебного курса.

**Оценивание:**

Зачтено	Не зачтено
Выполнены все пункты оценочных материалов.	Не выполнен хотя бы один из оценочных пунктов.

**Тесты:**

**Тест № 1 (образец);**

1. Выберите из списка логические операторы, существующие в языке JavaScript:

- ||
- And
- Or
- &
- !!

2. Выберите из списка операторы сравнения, существующие в языке JavaScript:

- =
- ==
- !=
- >=
- =>
- <>

3. Выберите оператор присваивания языка JavaScript:

- =
- :=
- ==
- :

4. Какими ключевыми словами может объявляться переменная в JavaScript?

- var, let, const
- let
- var, let
- let, const
- var, const

5. Что будет, если изменить значение const?

- Ничего, данные просто не изменятся
- Ошибка
- Там будет храниться новое значение

**Критерии оценивания:**

Тест оценивается по системе зачтено/не зачтено. Ответ на вопрос теста должен соответствовать действительности.

**Оценивание:**

Зачтено	Не зачтено
Правильные ответы слушателя составляют не менее 75% вопросов компьютерного проверочного теста.	Правильные ответы слушателя составляют менее 75% вопросов компьютерного проверочного теста.

**Тест № 2 (образец);**

1. Выберите из списка условные конструкции, существующие в языке JavaScript:

- if
- ?
- If...else
- switch

2. Где верный вариант?

- if a=5 console.log("a=5")
- if (a=5) console.log("a=5")
- if a==5 console.log("a=5")
- if (a==5) console.log("a=5")

3. Где верный вариант?

- if a==5 console.log("a равно 5")  
else console.log("a не равно 5");
- if (a==5) console.log("a равно 5")  
else console.log("a не равно 5");
- if (a==5) console.log("a равно 5");  
else console.log("a не равно 5");
- if a==5 console.log("a равно 5");  
else console.log("a не равно 5");

4. Где верный вариант?

- let b = (a === 5) ? a\*2 ; a;
- let b = (a === 5) ? a\*2 : a;

- let b = (a === 5) ? a\*2 , a;
- let b = (a === 5) ? (a\*2 , a);
- 5. Где допущена ошибка?
- if (a !== 5) console.log(a);
- if (a===5) console.log(“a равно 5”)  
else console.log(“a не равно 5”);
- console.log((a == 5) ? 'a=5' : 'a!=5');
- console.log(a == 5) ? 'a=5' : 'a!=5';

### Критерии оценивания:

Тест оценивается по системе зачтено/не зачтено. Ответ на вопрос теста должен соответствовать действительности.

### Оценивание:

Зачтено	Не зачтено
Правильные ответы слушателя составляют не менее 75% вопросов компьютерного проверочного теста.	Правильные ответы слушателя составляют менее 75% вопросов компьютерного проверочного теста.

### Тест № 3 (образец);

1. Выберите из списка циклы, существующие в языке JavaScript:

- for
- repeat
- do...while
- while
- loop

2. Выберите верный вариант:

- While a==5 {  
    console.log(a);  
    a--;  
}
- While a==5  
    console.log(a);  
    a--;

While (a==5) {  
     console.log(a);  
     a--;  
 }

While a==5 (  
     console.log(a);  
     a--;  
 )

While(a===5) (  
     console.log(a);  
     a--;  
 )

3. Выберите верный вариант:

for (let i; i<10; i++) console.log(i);

for (let i=0; i<10; i++){  
     console.log(i);  
 }

for (let i=0, i<10, i++){  
     console.log(i);  
 }

4. Где реализован бесконечный цикл?

While(true){ /\*тело цикла\*/ }

For(let i=10; i>0; i++){ /\*тело цикла\*/ }

While(1){ /\*тело цикла\*/ }

While('false'){ /\*тело цикла\*/ }

5. Тело какого цикла не будет выполнено?

While(false){ /\*тело цикла\*/ }

For(let i=0; i>0; i++){ /\*тело цикла\*/ }

For(let i=0; i>=0; i++){ /\*тело цикла\*/ }

While(''){ /\*тело цикла\*/ }

**Критерии оценивания:**

Тест оценивается по системе зачтено/не зачтено. Ответ на вопрос теста должен соответствовать действительности.

**Оценивание:**

Зачтено	Не зачтено
Правильные ответы слушателя составляют не менее 75% вопросов компьютерного проверочного теста.	Правильные ответы слушателя составляют менее 75% вопросов компьютерного проверочного теста.

**Тест № 4 (образец);**

1. Выберите метод, который запускает функцию для каждого элемента массива:
  - map
  - forEach
  - filter
  - reduce
  - splice
2. Выберите метод, который запускает функцию для каждого элемента массива и возвращает массив результатов:
  - map
  - forEach
  - filter
  - reduce
  - splice
3. Выберите метод, который вычисляет значение на основе всех элементов массива:
  - map
  - forEach
  - filter
  - reduce
  - splice
4. Выберите метод, который разбивает строку на массив по заданному разделителю:

- map
- forEach
- filter
- find
- split

5. Где объявлен массив?

- const a=[];
- let a = array(4, 5, 6)
- let a = new Array();

**Критерии оценивания:**

Тест оценивается по системе зачтено/не зачтено. Ответ на вопрос теста должен соответствовать действительности.

**Оценивание:**

Зачтено	Не зачтено
Правильные ответы слушателя составляют не менее 75% вопросов компьютерного проверочного теста.	Правильные ответы слушателя составляют менее 75% вопросов компьютерного проверочного теста.

**Тест № 5 (образец);**

1. Выберите неверный вариант объявления функции:

- ```
function func(param){
  console.log(param)
}
```
- ```
let func(param) => {
  console.log(param)
}
```
- ```
let func = (param) => {
  console.log(param)
}
```

2. Выберите верный вариант вызова функции:

```
function func(){
  console.log('function')
```



```
}
```

- func;
- func();
- function func();
- function func;

3. Выберите неверный вариант объявления функции:

function func(){  
    console.log('function')  
}

let func(param) => {  
    console.log('function')  
}

let func = (param) => {  
    console.log('function')  
}

4. Выберите верный вариант вызова функции:

```
function func(param1, param2){  
    console.log(param1 + param2)  
}
```

- func();
- func(1, '2');
- function();
- function(param1, param2);

5. С помощью какого слова можно сделать так, чтобы функция возвращала результат?

- result
- return
- revert

**Критерии оценивания:**

Тест оценивается по системе зачтено/не зачтено. Ответ на вопрос теста должен соответствовать действительности.

### Оценивание:

| Зачтено                                                                                        | Не зачтено                                                                                  |
|------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------|
| Правильные ответы слушателя составляют не менее 75% вопросов компьютерного проверочного теста. | Правильные ответы слушателя составляют менее 75% вопросов компьютерного проверочного теста. |

## 3.2. Итоговая аттестация

Зачёт по совокупности выполненных работ и итогового тестирования.

### Итоговое тестирование (образец):

1. Что будет выведено в результате работы программы?

```

1  let m=6;
2  let str='hello';
3
4  if (m<0){
5    | str=str+m;
6  }
7  else if (m>0){
8    | m--
9  }
10 str=m+m+str+m;
11 console.log(str);

```

- Ошибка
- 12hello6
- 10hello5
- Mmhellom
- 55hello5
- 66hello6

2. Что будет выведено в результате работы программы?

```

1  let m=123456;
2  let str='';
3  while(m>=1){
4      str=str+(m%10);
5      m=(m-(m%10))/10;
6  }
7
8  console.log(str);

```

- 65432
- Ошибка
- m%10
- Ничего
- 654321

3. Что делает данная функция?

```

1  function func(n, result=n)
2  {
3      if(n>1)
4      {
5          result=result*(--n);
6          return func(n,result);
7      }
8      return result;
9  }
10

```

- Возвращает факториал числа, переданного в параметрах
- Выводит факториал числа, переданного в параметрах
- Возвращает количество разрядов числа, переданного в параметрах
- Выводит количество разрядов числа, переданного в параметрах

4. Функция должна считать количество разрядов числа, но работает некорректно. В какой строке ошибка (введите число)?

```

1  function getNumberOfDigits(n, result=0)
2  {
3      if (n === 0){
4          result = 1;
5      } else {
6          while(n > 1){
7              n = n/10;
8              result++;
9          }
10     }
11     return result;
12 }
13

```

5. Функция должна считать количество разрядов числа, но работает некорректно. Исправьте ошибку. Введите полностью исправленную строку без пробелов, где их можно не ставить.

```

1  function getNumberOfDigits(n, result=0)
2  {
3      if (n === 0){
4          result = 1;
5      } else {
6          while(n > 1){
7              n = n/10;
8              result++;
9          }
10     }
11     return result;
12 }
13

```

### Критерии оценивания:

Тест оценивается по системе зачтено/не зачтено. Ответ на вопрос теста должен соответствовать действительности.

### Оценивание:

| Зачтено                       | Не зачтено                           |
|-------------------------------|--------------------------------------|
| Все пункты задания выполнены. | Хотя бы один из пунктов не выполнен. |

Обучающийся считается аттестованным:

**Критерии оценивания:** если обучающимся получены «зачёты» по итогам промежуточного контроля и итоговому тестированию.

### Оценивание:

«Зачтено»: выполнены тесты № 1, № 2, № 3, № 4 и № 5, Проекты № 1, № 2, № 3 и № 4, итоговое тестирование.

**«Не зачтено»:** не выполнен хотя бы один из пяти тестов или не сделан хотя бы один из четырёх проектов, или не выполнено итоговое тестирование.

## **Раздел 4. Организационно-педагогические условия реализации программы**

### **4.1. Учебно-методическое обеспечение и информационное обеспечение программы**

#### **Нормативно-правовые документы**

1. Российская Федерация. Закон. Об образовании в Российской Федерации : Федеральный закон № 273-ФЗ : [принят Государственной думой 21 декабря 2012 года : одобрен Советом Федерации 26 декабря 2012 года ] : [Электронный ресурс] / Минпросвещения России (официальный интернет-ресурс) : <http://pravo.gov.ru/proxy/ips/?docbody=&nd=102162745> (дата обращения 01.09.2023). – Текст : электронный.
2. Российская Федерация. Приказ. Об утверждении и введении в действие федерального государственного образовательного стандарта среднего общего образования : приказ Министерства образования и науки РФ от 6 октября 2009 г. № 413 : [приложения утверждены приказом Министерства образования и науки РФ от 17 мая 2012 г. № 413] : [Электронный ресурс] / Реестр примерных основных общеобразовательных программ : <https://fgosreestr.ru/uploads/files/cc49b8b607ab29a7ea856f3a8cfd17d9.pdf> (дата обращения 01.09.2023). – Текст : электронный.
3. Российская Федерация. Приказ. Об утверждении федерального государственного образовательного стандарта основного общего образования : приказ Министерства образования и науки РФ от 31 мая 2021г. № 287 : [утверждён приказом Министерства образования и науки РФ от 31 мая 2021 г. № 287] : [Электронный ресурс] / Реестр примерных основных общеобразовательных программ : <https://fgosreestr.ru/uploads/files/238eb2e61e443460b65a83a2242abd57.pdf> (дата обращения 01.09.2023). – Текст : электронный.

4. Российская Федерация. Приказ. О внесении изменений в федеральный государственный образовательный стандарт среднего общего образования, утверждённый приказом Министерства образования и науки Российской Федерации от 17 мая 20 12 г. № 413 : приказ Министерства просвещения Российской Федерации от 12 августа 2022 г. № 732 : [приложения утверждены приказом Министерства образования и науки РФ от 17 мая 2012 г. № 413] : [Электронный ресурс] / Реестр примерных основных общеобразовательных программ : <https://fgosreestr.ru/uploads/files/3ecd094e3813dce94559978a8a95fc4e.pdf> (дата обращения 01.09.2023). – Текст : электронный.

### Основная литература

1. Флэнаган, Дэвид. Ф73 JavaScript. Полное руководство: пер. с англ. – СПб. : ООО «Диалектика», 2021. – 720 с.: ил. – парал. тит. англ.

### Дополнительная литература

1. Минник Крис, Холланд, Ева. М62 JavaScript для чайников. : пер. с англ. – М. : ООО «И.Д. Вильямс», 2017 - 321с. : ил. – парал. тит. англ.
2. Фримен Э., Робсон Э. Изучаем программирование на JavaScript. – СПб. : Питер, 2015. – 640 с. : ил. – (Серия «Head First O’Reilly»)

## 4.2. Материально-технические условия реализации программы

| Наименование аудиторий                                     | Вид занятий                          | Наименование оборудования, программного обеспечения                           |
|------------------------------------------------------------|--------------------------------------|-------------------------------------------------------------------------------|
| Рабочие места, оборудованные для проведения онлайн-лекций. | Онлайн-лекция / практические занятия | Компьютеры (ноутбуки) с доступом в интернет, 1 компьютер на каждого слушателя |

### **4.3. Кадровые условия реализации программы**

Количество преподавателей, привлечённых для реализации дополнительной профессиональной программы повышения квалификации, – не менее одного человека: может быть штатным сотрудником или внешним совместителем, обладающим компетенциями в области программирования.