

Демо

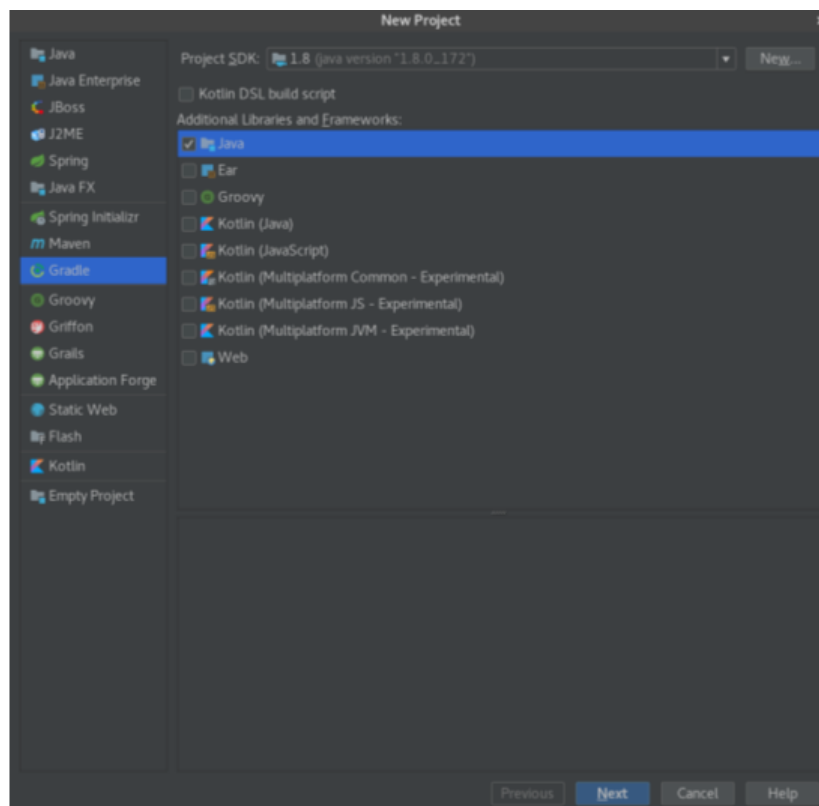
если будет время рассказать, как установить IDEA и Java для неё (см. «Настройка ПО.docx»).

## 1. Создание Gradle – проекта в IDE IntelliJ IDEA

Во многие среды разработки встроена интеграция с Gradle. Достаточно создать проект с использованием Gradle.



Создание проекта в IntelliJ IDEA



Создание проекта с использованием Gradle

## 2. Подключение тестов в Gradle – проект.

Для того, чтобы включить тесты в ваш проект, необходимо в файле

```
plugins {  
    id 'java'  
}  
  
group 'radik.labs'  
version '1.0-SNAPSHOT'  
  
repositories {  
    mavenCentral()  
}  
  
buildDir = "target"  
  
dependencies {  
    testCompile 'org.testng:testng:6.14.3'  
}
```

Важным тут является секция dependencies – мы указываем там название тестового пакета и его версию. Берётся она из репозитория mavenCentral() (в секции repositories).

## 3. Пример написания тестов

### Код, который нужно протестировать:

```
package variant_1;  
  
public class Queue<T> {  
    private Element<T> head = null;  
  
    private Element<T> tail = null;  
  
    public Queue() {  
    }  
  
    /**  
     * Добавляет элемент в очередь  
     */  
    void offer(T value) {  
        if (head == null) {  
            head = new Element<>(value);  
            head.next = null;  
            tail = head;  
        }  
    }  
}
```

```

    } else {
        // добавляем элементы в tail - конец очереди
        Element<T> newNode = new Element<>(value);
        newNode.next = tail;
        tail = newNode;
    }
}

/**
 * Удаляет элемент из очереди
 */
T poll() {
    if (head != null) {
        Element<T> temp = tail;
        while (temp.next != head) {
            temp = temp.next;
        }
        Element<T> proxy = head;
        head = temp;
        return proxy.value;
    } else {
        throw new NullPointerException("You're trying to " +
            "get element from empty queue!");
    }
}

/**
 * Проверка на пустоту
 */
boolean isEmpty() {
    return head == null;
}

private class Element<T> {
    Element(T value) {
        this.value = value;
    }

    Element<T> next = null;
    T value;
}
}

```

### Тесты:

```
package variant_1;
```

```
import org.testng.annotations.Test;
```

```
import static org.testng.Assert.*;
```

```
public class QueueTest {
```

```
    @Test
```

```
    void test_offer() {
```

```
        Queue<Integer> queue = new Queue<>();
```

```
        queue.offer(4);
```

```
        queue.offer(55);
```

```
        assertFalse(queue.isEmpty());
```

```
    }
```

```
    @Test
```

```
    void test_poll() {
```

```
        Queue<Integer> queue = new Queue<>();
```

```
        queue.offer(55);
```

```
        queue.poll();
```

```
        assertTrue(queue.isEmpty());
```

```
    }
```

```
    @Test(expectedExceptions = NullPointerException.class)
```

```
    void test_pollWithException() {
```

```
        Queue<Integer> queue = new Queue<>();
```

```
        queue.poll();
```

```
    }
```

```
    @Test
```

```
    void test_isEmpty() {
```

```
        Queue<Double> queue = new Queue<>();
```

```
        assertTrue(queue.isEmpty());
```

```
    }
```

```
    @Test
```

```
    void test_fullLogicTest() {
```

```
        Queue<Integer> queue = new Queue<>();
```

```
        queue.offer(6);
```

```
        assertFalse(queue.isEmpty());
```

```
        queue.offer(12);
```

```
        queue.offer(65);
```

```
        assertEquals(6, queue.poll().intValue());
```

```
    assertEquals(12, queue.poll().intValue());
    assertEquals(65, queue.poll().intValue());
    assertTrue(queue.isEmpty());
}
}
```