

Школа:		
Дата:	ФИО учителя:	
Класс:	Участвовали:	Не участвовали:
Тема урока: Конструкторы классов, обработчики событий		
Цели обучения, которые достигаются на данном уроке	<ul style="list-style-type: none"> • Ознакомить с понятием обработчики событий; • Рассмотреть основные понятия конструкторов класса. • Понимать особенности языка программирования Java 	
Цели урока	<p>Все учащиеся смогут:</p> <ul style="list-style-type: none"> • ознакомиться с понятием обработчики событий; <p>Большинство учащихся смогут:</p> <ul style="list-style-type: none"> • описывать основные понятия конструкторов класса <p>Некоторые учащиеся смогут:</p> <ul style="list-style-type: none"> • смогут набрать готовую программу в среде программирования 	
Критерии оценивания	<ul style="list-style-type: none"> • может рассказать про ООП • может описать устройство объектов, их поведение и способ представления 	
Воспитание ценностей	Формировать умение работать в команде, эффективно распределять обязанности;	
Предварительные знания	Базовые знания Java	
Межпредметные связи	Информатика	
Запланированные этапы урока	Запланированная деятельность на уроке	Ресурсы
Начало урока _5_ мин	<p>Организационный момент. Создание благоприятной среды на уроке, мотивация учащихся на успех.</p> <p>Обработка событий в Swing Графический интерфейс пользователя включает в себя не только расположение в окне необходимых элементов управления, но и назначение реакции на пользовательские события. Большая часть действий в оконных программах выполняется в ответ на выбор пользователем команд меню, нажатие кнопок, а иногда даже просто в ответ на ввод нового символа в текстовое поле. Таким образом, при разработке программы необходимо: Выявить события, в ответ на которые потребуется реакция программы.</p>	Презентация
Середина урока 30 мин	<p>Написать код, реализующий эту реакцию (так называемый обработчик событий). Связать обработчик события с соответствующим событием. Первый пункт зависит от логики работы приложения и находится всецело на усмотрении программиста. Обработчик события — это, как правило, обычный метод и ничего особенного в его написании нет.</p> <p>Паттерн проектирования «наблюдатель» В разных языках программирования и визуальных библиотеках используются самые различные способы привязки процедуры (метода), которая должна вызываться в ответ на некоторое событие, к самому этому событию. В частности, в Delphi процедура просто должна иметь особое название, состоящее из</p>	

	<p>имени компонента и типа обрабатываемого события. В библиотеке Swing для привязки обработчика события к вызывающему его компоненту используется паттерн проектирования «наблюдатель».</p> <p>Паттерны проектирования — это стандартные приемы объектно-ориентированного программирования, позволяющие оптимальным образом справиться с нетривиальными, но часто возникающими в программировании ситуации. Паттерн проектирования описывает классы, которые необходимо ввести для разрешения такой ситуации и взаимодействие между классами. В частности, ранее мы рассматривали паттерн проектирования «итератор», позволяющий обойти все элементы коллекции с помощью двух простейших методов в порядке, скрытом в самом итераторе.</p>	
<p>Конец урока 5 мин</p>	<p>Вопросы:</p> <p>Рефлексия по содержанию по ключевым словам (глаголы) Закончи фразу: ООП позволяет...</p> <ul style="list-style-type: none"> ● уменьшить ● повысить ● обеспечить 	
<p>Дифференциация – каким образом Вы планируете оказать больше поддержки? Какие задачи Вы планируете поставить перед более способными учащимися?</p>	<p>Оценивание – как Вы планируете проверить уровень усвоения материала учащимися?</p>	<p>Охрана здоровья и соблюдение техники безопасности</p>
<p>1. Разделить детей на группы по 3 -4 человека 2. Поддержка друг друга учащимися, помощь учителя</p>	<p>1. Закрытые вопросы 2. Открытые вопросы</p>	<p>Физминутка</p>