



Python. Модуль 2. Урок 2.

Методические указания

Тема: Виджеты в tkinter. Класс Button. Создание и использование кнопки. Метод bind и place. Понятие функции.

Цели урока:

- Обеспечить закрепление учащимися понятий: координаты, объект, метод, класс, окно, канва.
- Обеспечить усвоение учениками понятий: событие, размещение объекта, функция.
- Отработать навыки:
 - набора текста и его форматирования;
 - работы с консольным приложением в среде разработки;
 - решение задач с использованием методов bind и place;
 - решение задач с использованием функций;
 - решения задач с использованием методов create_rectangle(), create_oval(), create_polygon(), pack().
- Создать условия для развития:
 - логического и алгоритмического мышления;
 - элементов творческой деятельности;
 - памяти;
 - критического мышления, культуры групповой самоорганизации;
 - коммуникативной культуры.
- Организовать самостоятельную работу учащихся, направленную на развитие алгоритмического мышления, познавательных интересов и навыков работы на компьютере.

Задачи урока:

- повторить и закрепить понятия, пройденные на предыдущих уроках;
- изучить работу метода mainloop();
- изучить Button и создание кнопки;
- написать программу с использованием Button;
- изучить методы bind(), place();
- отработать навык составления алгоритма с использованием bind(), place();
- написать программу с использованием bind(), place().



Перед началом урока:

1. Открыть страницу с логинами и паролями для детей.
2. Открыть и запустить на проекторе презентацию (**на слайдах присутствует gif анимация. Для её отображения запустить презентацию**).
3. Запустить тестовую программу.
4. Проверить наличие раздатки.

План урока:

- [1. Анонс занятия \(5 мин.\).](#)
- [2. Повторение. Часть 1 \(5 мин.\).](#)
- [3. Создание окна заданного размера \(10 мин.\).](#)
- [4. Повторение. Часть 2. Работа за компьютером \(10 мин.\).](#)
- [5. Виджет - кнопка \(10 мин.\).](#)
- [6. Событие и действие \(20 мин.\).](#)
- [7. Самостоятельная работа \(15 мин.\).](#)
- [8. Рефлексия \(10 мин.\).](#)
- [9. Выдача задания на дом \(5 мин.\).](#)

1. Анонс занятия (5 мин.).

Слайд 2. Соберите детей в свободной от компьютеров зоне и обсудите, с какими понятиями им сегодня предстоит работать. С какими понятиями они уже знакомы, а что будет для них новым.

2. Повторение. Часть 1 (5 мин.).

Слайд 3. Вспомните с детьми, что они изучали на прошлых занятиях, с какими понятиями познакомились. Для чего использовали ту или иную команду.

Слайды 4-8. Вспомогательные для повторения прошлого материала.

Слайды 9-12. Вспомните с детьми, как создается и для чего нужен холст. Откуда начинается отсчет координат в tkinter.



3. Создание окна заданного размера (10 мин.).

Слайд 13. Обсудите с детьми, как происходит создание холста. Можно провести аналогию с натягиванием художников полотна для будущей картины, то есть оно занимает все пространство. Но кроме холста должны же быть и другие элементы, да и мы не рисуем такие большие картины. Окно - это основа для крепления всех элементов, как пробковая доска, на которую с помощью кнопок (метода `pack`), мы можем размещать элементы - виджеты.

Слайд 14. Метод `geometry` позволяет создать окно определенного размера. Обратите особое внимание на синтаксис команды. ОДИНАРНЫЕ КАВЫЧКИ и между шириной и длиной знак `x` - "икс". Не знак умножения! Попросите детей написать программу с созданием окна размером 800x800 пикселей.

Слайд 15. В свою очередь, метод `place` - место, позволяет указывает точку начала размещения объекта. Точкой начала считается левый верхний угол. На картинке показано, что объект может быть меньше окна и располагаться в его части. В скобках достаточно указать координаты `x` и `y`.

Слайды 16-17. Обсудите с детьми программу, что мы увидим после ее запуска? Обратите внимание на размеры прямоугольника. Он нам покажет границы холста. Начальная точка `10,10` у прямоугольника выбрана не случайно, при меньших значениях мы не видим левые границы, не уместается толщина линии, поэтому выполнен отступ в 10 пикселей.

4. Повторение. Часть 2. Работа за компьютером (10 мин.).

Слайды 18-22. Повторите с детьми заливку фигур и основные методы для создания фигур: прямоугольник, многоугольник, овал, линия. С помощью листа раздатки дети должны определить координаты фигур. Обратите внимание детей, что несмотря на изменение расположения холста, внутри холста координаты не изменились, то есть левый верхний угол холста - это точка `(0;0)`.

5. Виджет - кнопка (10 мин.).

Слайды 23-24. При использовании модуля `turtle` мы всегда видели, как создается фигура, сейчас же фигуры появляются при появлении окна, практически моментально. Но можем ли мы проконтролировать порядок их появления? Нет. Так как для пользователя это все выглядит



одномоментно, не важен даже порядок написания команд в коде. Хотели бы мы научиться это контролировать? Конечно!

Слайды 25-28. Мы постоянно сталкиваемся с различными приложениями: игры, навигаторы, карты, социальные сети и везде есть кнопки. Какой функционал закреплен за этими кнопками? Какие имена они имеют? За что отвечают? Какое назначение у них может быть в нашем приложении?

Слайды 29-31. Разберите с детьми порядок создание кнопки. К какому классу будет относиться объект, как задается имя виджета. Выполните задание по созданию кнопки Квадрат.

Слайды 32-33. Кнопка есть, но работает ли она? Нет. Это просто макет, который пока ничего не умеет делать.

6. Событие и действие (20 мин.).

Слайд 34. Основа любого приложения - это отслеживание действий пользователя и выдача определенной реакции. Событие - это то, что происходит после действия пользователя. Например, на двери стоит ведро с водой, что произойдет, если дверь толкнут? Ведро упадет - это событие. Толкание двери - это действие. Действие пользователя - это нажать кнопку, событие - открытие файла и т.п.

Слайд 35. Событие необходимо описать для компьютера. Для этого используют функции. Это правило, которому должна следовать программа после определенного действия пользователя. То есть ведро всегда будет падать, когда толкают дверь. Не может в третий раз упасть подушка или что-то другое. Действие пользователя, событие и функция четко связаны между собой.

Слайд 36. Разберите конструкцию функции в Python.

Слайды 37-39. Для того, чтобы связать между собой все понятия, используется метод `bind`. В нем мы указываем действие пользователя и имя функции, которую необходимо вызвать после этого.

Слайды 40-45. Разбираем вместе с детьми изменение в программе. Добавилась одна функция по созданию прямоугольника после нажатие кнопки Квадрат левой кнопкой мыши. Попросите детей изменить свою программу.

7. Самостоятельная работа (15 мин.).

Слайды 46-47. Выдайте детям задание по созданию еще двух кнопок для остальных фигур. Обсудите название функций и их содержание.

Слайды 48-49. Выдайте детям задание по расположению кнопок в один ряд.



8. Рефлексия (10 мин.).

Слайд 50. Повторите материал занятия. Не забудьте загрузить проекты детей на склад.

«Отлично! Вы все большие молодцы! Но не забывайте, что только при регулярной практике и работе дома вы сможете добиться поставленных целей. Всем спасибо, жду вас на следующем занятии!»

9. Выдача задания на дом (5 мин.)

Слайды 51-52. Задание с двумя уровнями сложности:

1. Базовое, определяет степень усвоения нового материала.
2. Расширенное, связанное с материалами прошлых занятий.