

# Task 1.

+ 2 point

## Group Discussion



**1st group:** What is the most important element in video games?

**2nd group:** What games do you know using 2d graphics?

**3rd group:** What libraries of modules in Python do you know?

**Lesson topic:**  
**Drawing shapes. Pygame.draw**  
**Module**

## **Lesson objectives:**

**learn other pygame modules;  
draw different shapes.**



# KWL chart

Name \_\_\_\_\_

Date \_\_\_\_\_

Topic: \_\_\_\_\_

## Know

Before you read, write what you think you know about the topic.

## wonder

Before or during your research, record questions about the topic.

## Learned

After you finish reading, write what you learned about the topic.

## Task 2.

+ 3 point

# Work in Groups

### 1st group

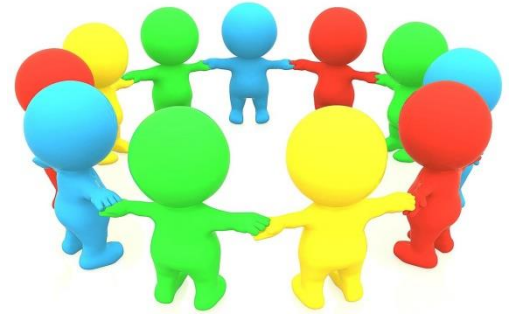
PyGame Drawing Basics

### 2nd Group

Creating a rectangle and a circle.

### 3rd Group

Creating a polygon and a line.



## Pygame.draw Module

`pygame.draw.rect`

`pygame.draw.polygon`

`pygame.draw.circle`

`pygame.draw.ellipse`

`pygame.draw.line`

`pygame.draw.lines`

`pygame.draw.arc`

draws a rectangular shape on the Surface

draws a shape with any number of sides

draws a circle around a point

draws a round shape inside a rectangle

draws a straight line segment

draws multiple contiguous line segments

draws a partial section of an ellipse

The Draw class has 9 methods for drawing geometric shapes, which have the following general parameters:

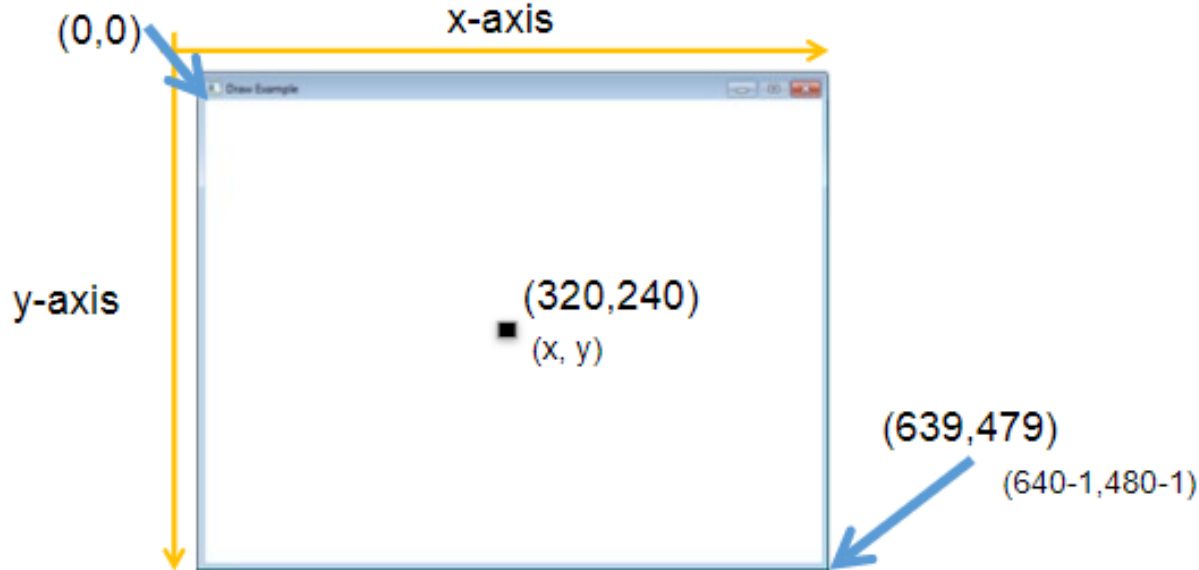
**Surface** is the drawing surface, in our case it is the screen object.

**Color** is the color of the shape, it is an RGB tuple, for example, RED = (255, 0, 0).

**Rect** - a rectangular area in which the shape will be drawn. Specified by the Rect(x, y, w, h) tuple, x, y are the coordinates of the upper left corner, w, h are the width, height, width is the line thickness, if width=0, then a filled figure is drawn.

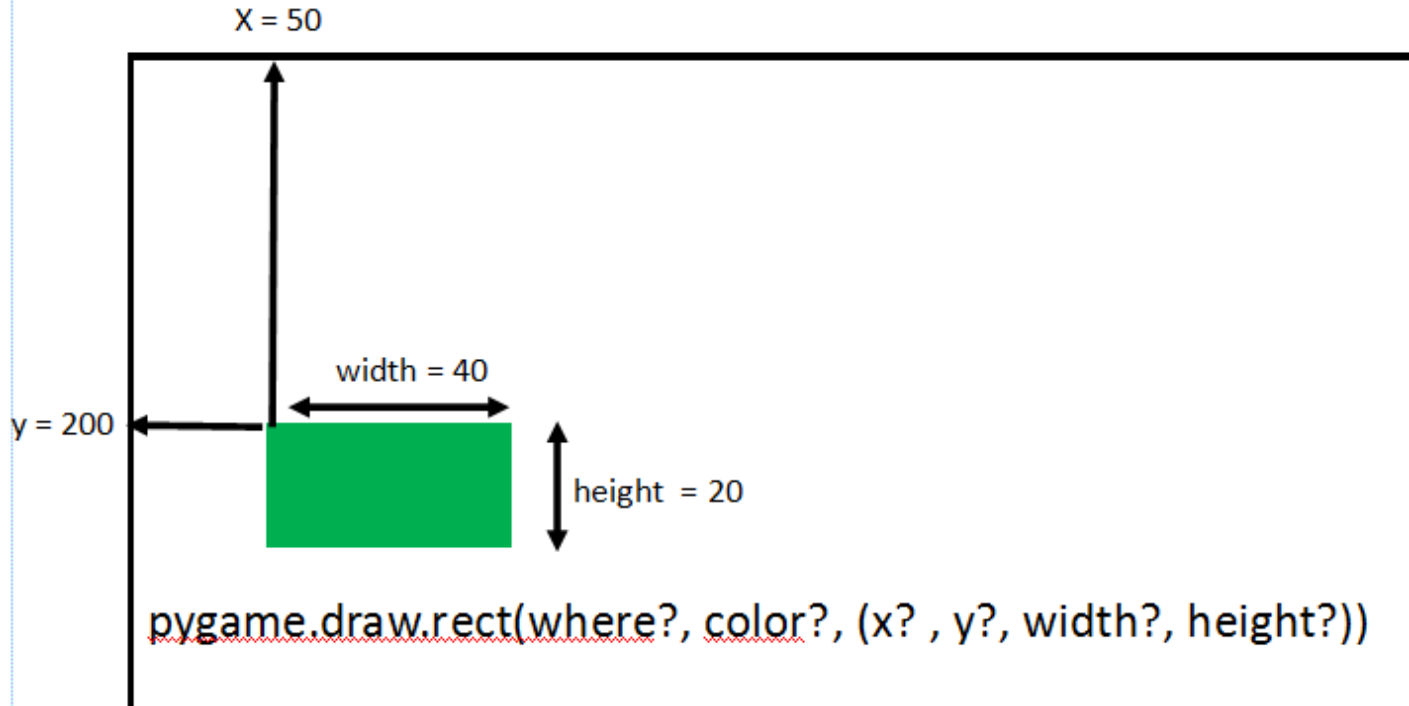
# Screen Coordinates

The screen has pixel coordinates starting from (0, 0) in the top-left, and moving across and down.



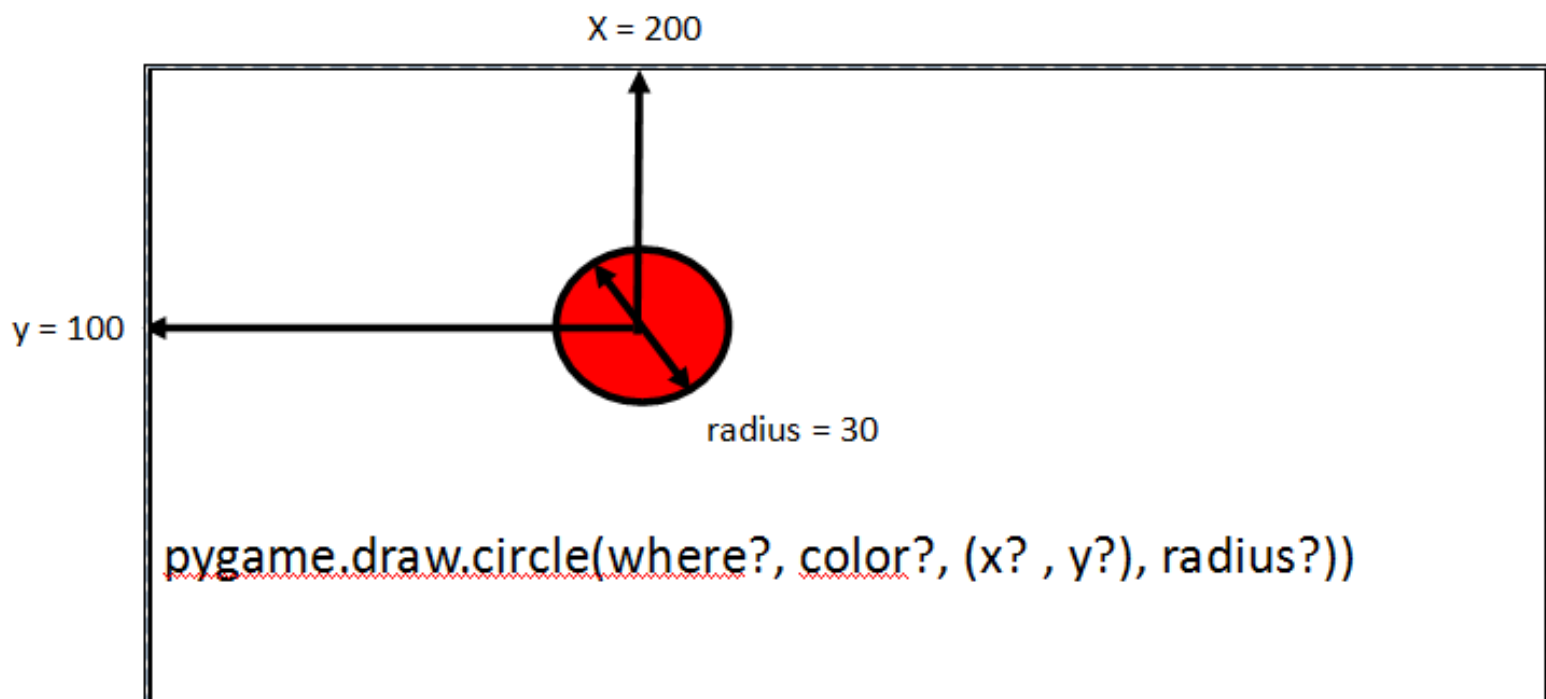


```
pygame.draw.rect(screen, green, (50, 200, 40, 20))
```



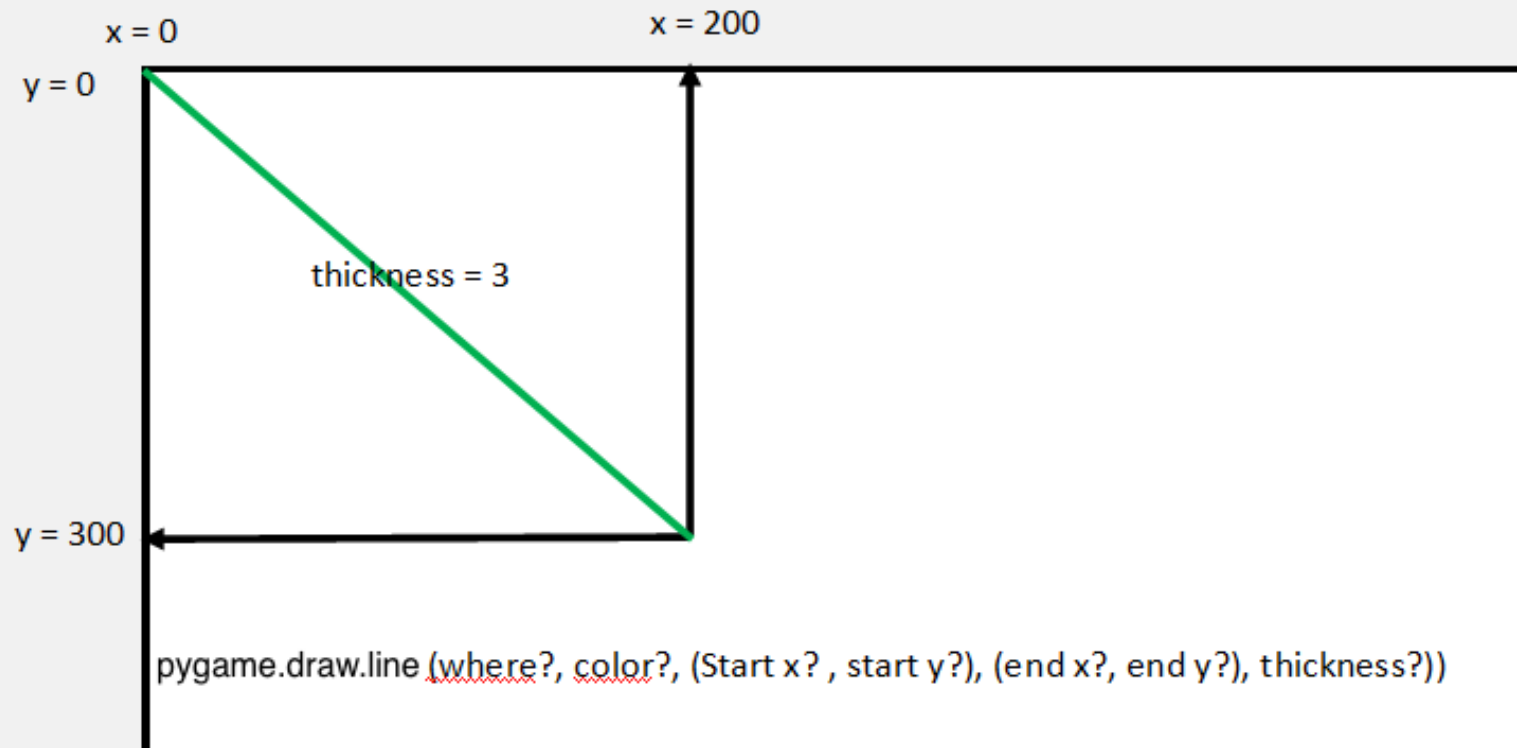
[https://youtu.be/53ucZ6\\_7\\_t4](https://youtu.be/53ucZ6_7_t4)  
(1.13-2.25)

```
pygame.draw.circle(screen, red, (200, 100), 30)
```



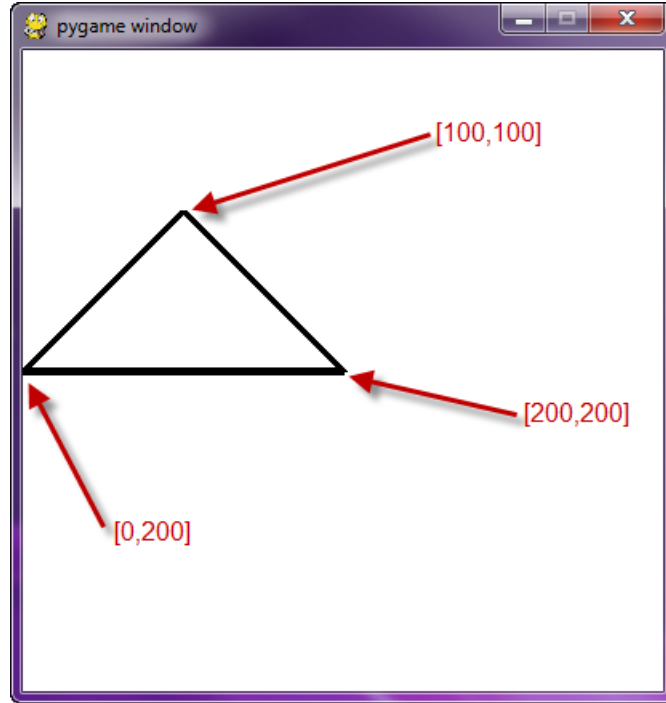
```
pygame.draw.circle(where?, color?, (x? , y?), radius?)
```

```
pygame.draw.line (screen, green, (0, 0), (200,300), 3)
```



```
pygame.draw.line (where?, color?, (Start x?, start y?), (end x?, end y?), thickness?)
```

```
# This draws a triangle using the polygon command  
pygame.draw.polygon(screen, BLACK, [[100,100], [0,200], [200,200]], 5)
```



## Task 3.

# Practical work

+ 3 point

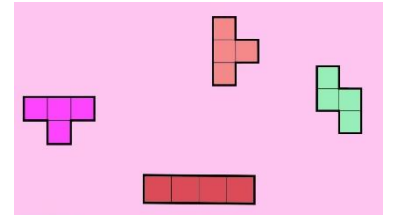
### 1st group

Draw a tree from Minecraft.



### 2nd group

Draw the shapes from Tetris.



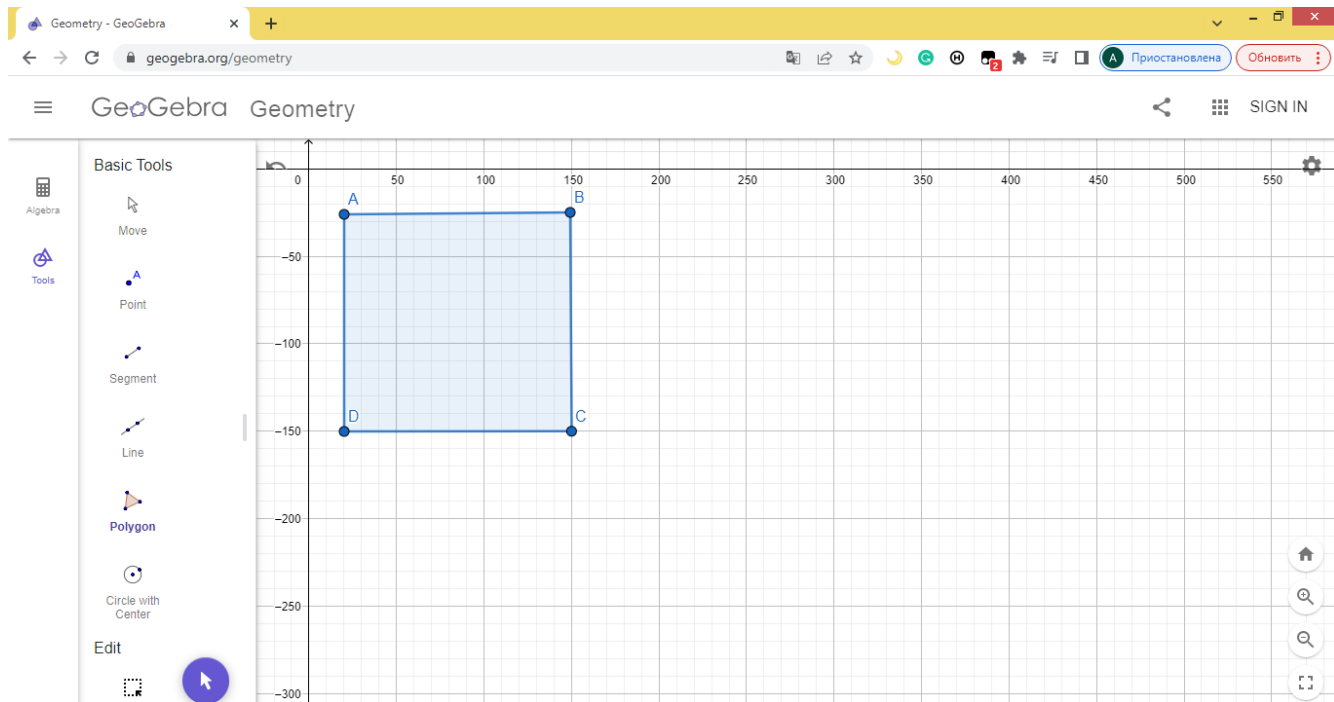
### 3rd group

Draw background elements from Super Mario.



If you need to determine the coordinates of a figure, you can use the geogebra math program:

<https://www.geogebra.org/geometry>



The image shows a screenshot of the GeoGebra Geometry web application. The browser address bar displays "geogebra.org/geometry". The application interface includes a left sidebar with "Basic Tools" such as Move, Point, Segment, Line, Polygon, and Circle with Center. The main workspace features a coordinate grid with x and y axes ranging from 0 to 550 and -300 to -150, respectively. A square is plotted with vertices labeled A, B, C, and D. The vertices are located at approximately (20, -20), (150, -20), (150, -150), and (20, -150). The square is filled with a light blue color and has a dark blue border. The top right corner of the application shows a "SIGN IN" button and a settings icon.

# Task 4.



+ 5 point

# PIN for task: 9RVPD4

The screenshot shows the Wizer.me editor interface. At the top, there is a navigation bar with options: Dashboard, Create, Preview (selected), Assign, Assess, Assess 2.0, and Insights. A 'WORKSHEET' button is visible on the right. Below the navigation bar, the main content area displays a worksheet preview. The worksheet title is 'Drawing shapes. Pygame.draw Module'. The background of the worksheet is decorated with colorful confetti. A central text box contains the title in green. Below the title, there is a matching exercise with the instruction 'Match the module name with its function'. The exercise consists of three rows, each with a module name on the left and a function description on the right, separated by a vertical line. The module names are 'pygame.draw.rect', 'pygame.draw.circle', and 'pygame.draw.line'. The function descriptions are 'draws a circle around a point', 'draws a straight line segment', and 'draws a shape with any number of sides'. Each module name and function description is enclosed in a rounded rectangle with an orange circle next to it, indicating a matching point.

Review your differentiated worksheet

**RULES PREVIEW**

See how they reflect on this worksheet

Select rules to preview

Select rules to view Differentiated versions of this worksheet.

A preview of the learners who receive Differentiated Instruction will appear here.

## Drawing shapes. Pygame.draw Module

Match the module name with its function

pygame.draw.rect	draws a circle around a point
pygame.draw.circle	draws a straight line segment
pygame.draw.line	draws a shape with any number of sides