Qualifications

Demonstrate your knowledge to begin working on the tasks.

Show that you are ready to brainstorm!

Qualifications

What is a variable ?

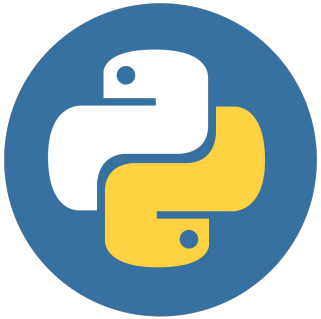
What is it for?

Qualifications

A variable

is a data element that has a name.

Variables are used to store the information that a program works with.



Qualifications

What is a function ?

What are functions for?

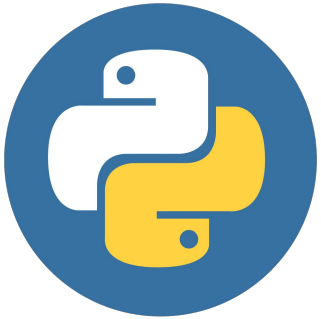
Qualifications

A function

is <u>an action with data</u> :

an algorithm made up of already known commands that has a name.

A function gives the programmer a command that executes an entire algorithm. This command can be used multiple times in the program.



Qualifications

Which commands match the descriptions?

Move the turtle 100 pixels **forward**	?
Turn the turtle right 100 degrees	?
Turn the turtle left 100 degrees	?
Change the thickness of the turtle's **pen** to 5 pixels	?
Change the color of the turtle's **pen** to orange	?

Qualifications

Which commands match the descriptions?

Move the turtle 100 pixels **forward**	`forward(100)`
Turn the turtle right 100 degrees	`right(100)`
Turn the turtle left 100 degrees	`left(100)`
Change the thickness of the turtle's **pen** to 5 pixels	`pensize(5)`
Change the color of the turtle's **pen** to orange	`color("orange")`

Qualifications

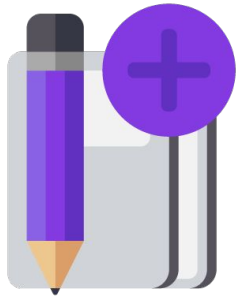
Qualifications confirmed!

Great, you are ready to brainstorm and work on your tasks!

Qualifications

Brainstorming:

Working with objects

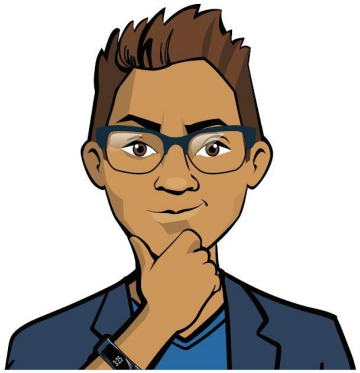


Working with objects

Before starting the Turtle Races, let's learn how to work with multiple objects in the same program.

But what is an object?

What examples of real world objects do you know?

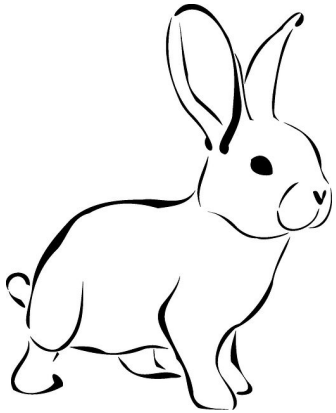


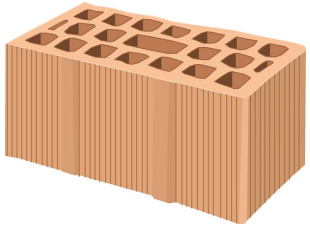
Brainstorming

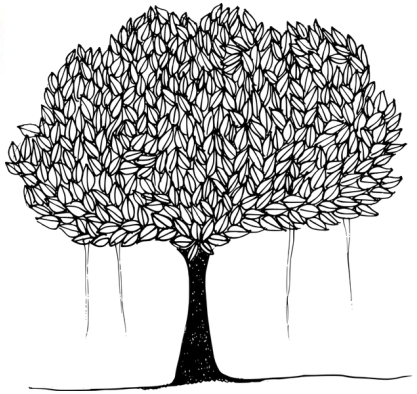
An object

is a set of data and actions that is convenient to perceive as a whole.

Real world objects:







Brainstorming

Each of these objects stores the necessary information about itself and knows how to perform some actions.

Brainstorming

In the English language: Rabbit, run!

In the programming language: Rabbit.run()

Object properties and methods

Every object has properties and is controlled by methods.

Properties	*Methods*
rabbit.speed = 50	rabbit.run()
turtle.speed = 1	turtle.walk()
fish.speed = 30	fish.swim()

Brainstorming

Object properties and methods

Every object has properties and is controlled by methods.

Properties	*Methods*
rabbit.speed = 50	rabbit.run()
turtle.speed = 1	turtle.walk()
fish.speed = 30	fish.swim()

Variable placed inside the object.

Function placed inside the object.

Brainstorming

Accessing properties and methods

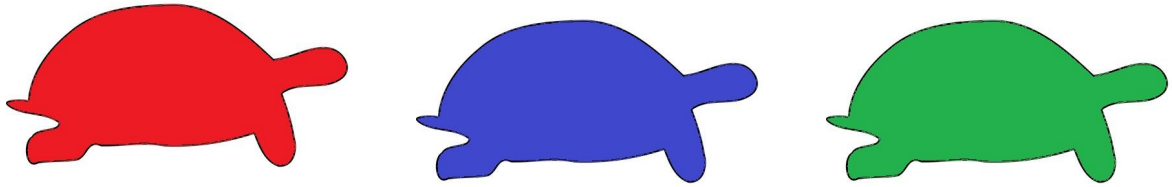
The dot is the choice between what the object remembers and is able to do.

```
object.method()
object.property
```

Brainstorming

Object-Oriented Programming

is an approach based on creating objects and controlling them.



Brainstorming

Let’s look at the task

The task. Program the animation as shown in the picture. Three lines (green, red, and blue) are drawn in different directions at the same time.

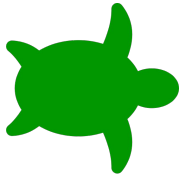
What kind of objects do we need?

Brainstorming

Let’s look at the task

The task. Program the animation as shown in the picture. Three lines (green, red, and blue) are drawn in different directions at the same time.

One line is drawn by one **object — a turtle**. There are **3 turtles** in total.



We set a **color** and **starting position** for each turtle.

We draw the pattern with three turtles at the same time.

Brainstorming

Let’s look at the task

The task. Program the animation as shown in the picture. Three lines (green, red, and blue) are drawn in different directions at the same time.

```
from turtle import *

t1 = Turtle()
t1.color('red')

t2 = Turtle()
t2.color('green')
t2.left(120)

t3 = Turtle()
t3.color('blue')
t3.left(240)
```

Created by three turtles.

Brainstorming

Let’s look at the task

The task. Program the animation as shown in the picture. Three lines (green, red, and blue) are drawn in different directions at the same time.

```
for i in range(60):
    t1.forward(5)
    t1.left(5)
    t2.forward(5)
    t2.left(5)
    t3.forward(5)
    t3.left(5)
```

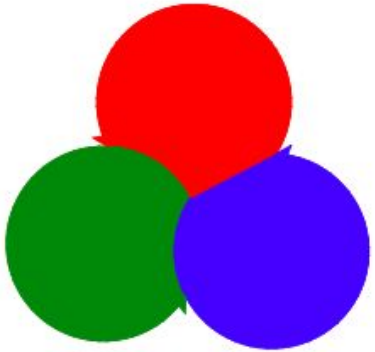
Drawing the pattern.

Brainstorming

Let’s look at the task

The task. Program the animation as shown in the picture. Three lines (green, red, and blue) are drawn in different directions at the same time.

There is an option to color in the different sectors. How do we do that?



Brainstorming

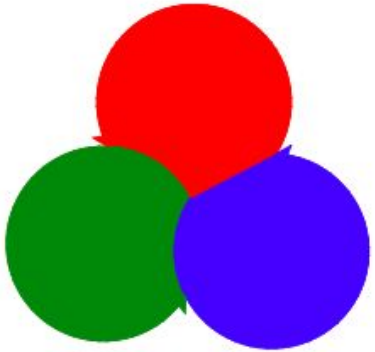
Let’s look at the task

The task. Program the animation as shown in the picture. Three lines (green, red, and blue) are drawn in different directions at the same time.

```
t1.begin_fill()
t2.begin_fill()
t3.begin_fill()

#drawing the pattern

t1.end_fill()
t2.end_fill()
t3.end_fill()
```



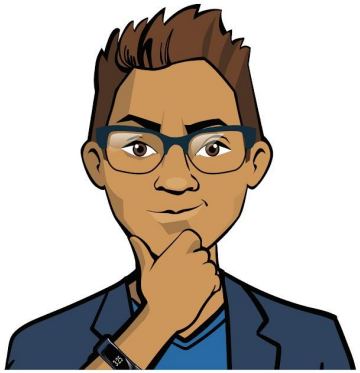
Brainstorming

Before we continue:

1. **Describe in your own words**: what is an object? Object property? Object method?

2. Give examples of real-life objects.
   What type of Python object do you already know?

3. How do we **access the property** of an object? Or the method?



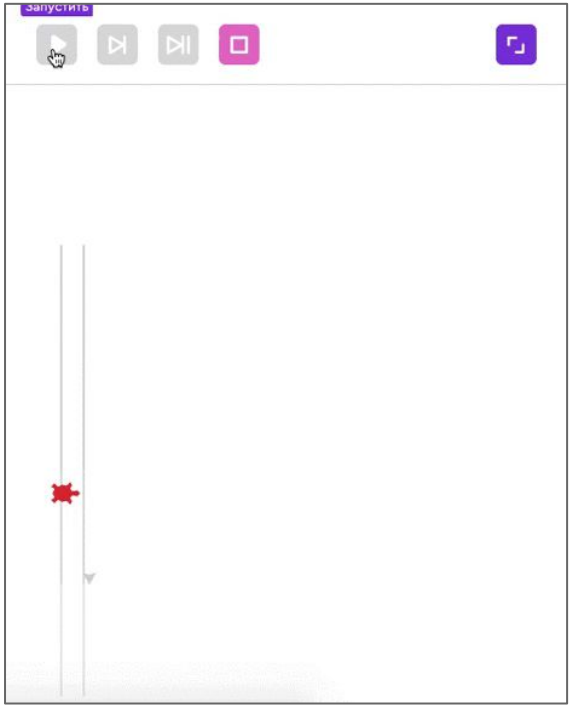
Brainstorming

Terms of Reference

The Turtle Races game. When the game starts, the course markings are drawn and two turtles appear. The speed of each participant is set randomly. The goal of the game is to guess who will finish first.

Project stages:

- course markings;
- race;
- victory dance.

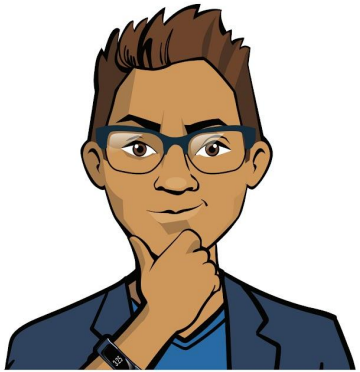


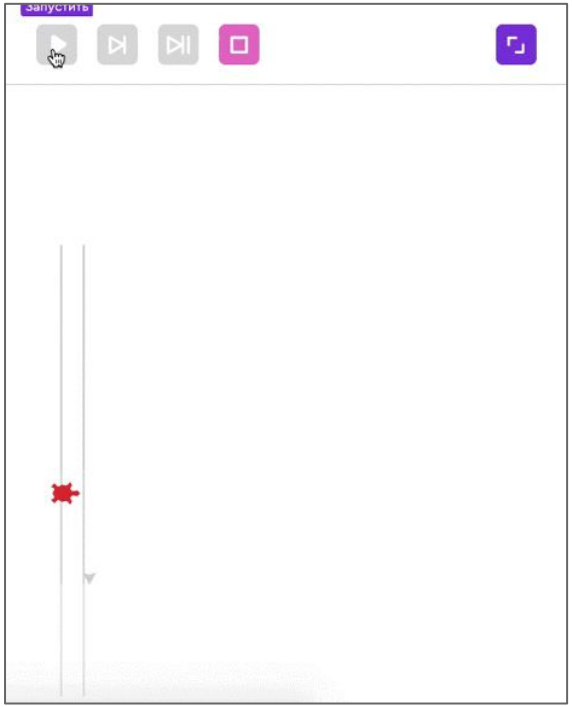
Brainstorming

Terms of Reference

The Turtle Races game. When the game starts, the course markings are drawn and two turtles appear. The speed of each participant is set randomly. The goal of the game is to guess who will finish first.

Let's look at a simplified prototype without markings.





Brainstorming

Turtle: new commands

Command	*Purpose*
`shape('turtle')`	Changes the turtle's "costume" (object shape). Possible costumes: "arrow", "turtle", "circle", "square", "triangle", "classic".
`xcor()`	The turtle's current X-coordinate (horizontal).

The commands are now applied to the objects:

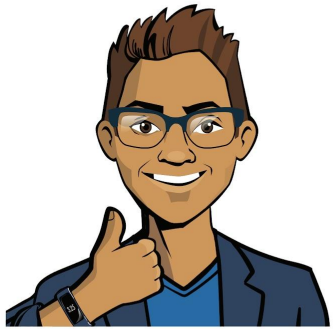
`shape('turtle')` → `t1.shape('turtle')`

Brainstorming

New command from the Python standard library

Command	*Purpose*
`result = max(number1, number2, ...)`	Determines the greatest of the given numbers and returns it.

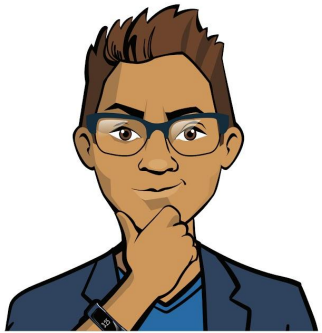
This command will come in handy when identifying the winner!

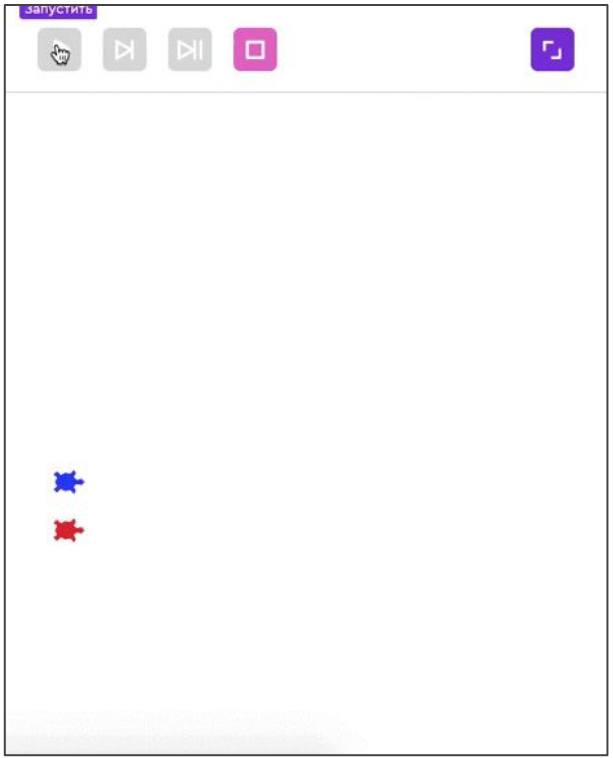


Brainstorming

Turtle Races

How do we program races the?

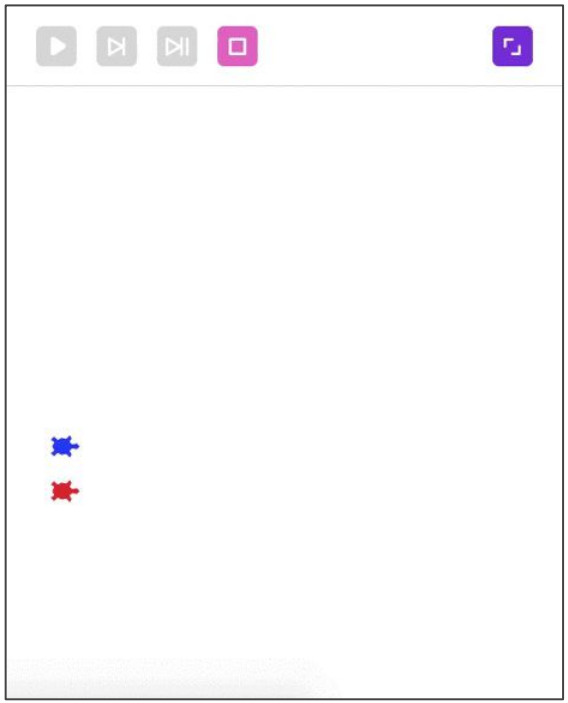




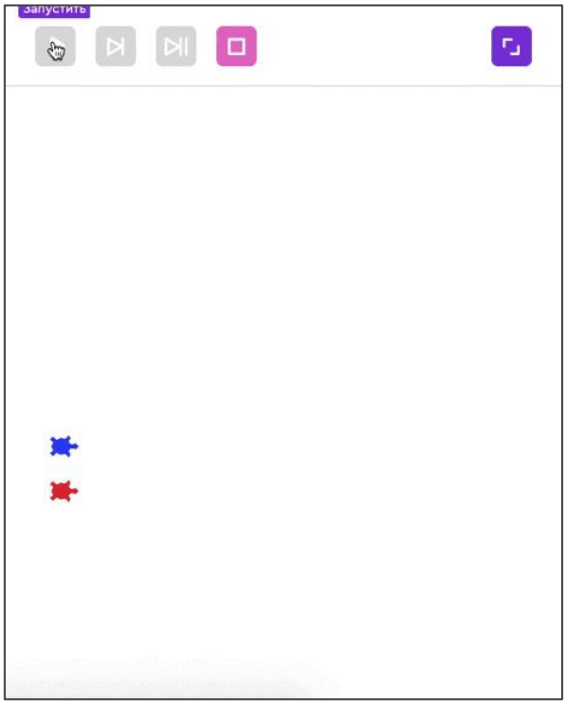
Brainstorming

Turtle Races

There won’t be any intrigue in the game if you set a random but constant speed for each turtle. The speed of the object should change.

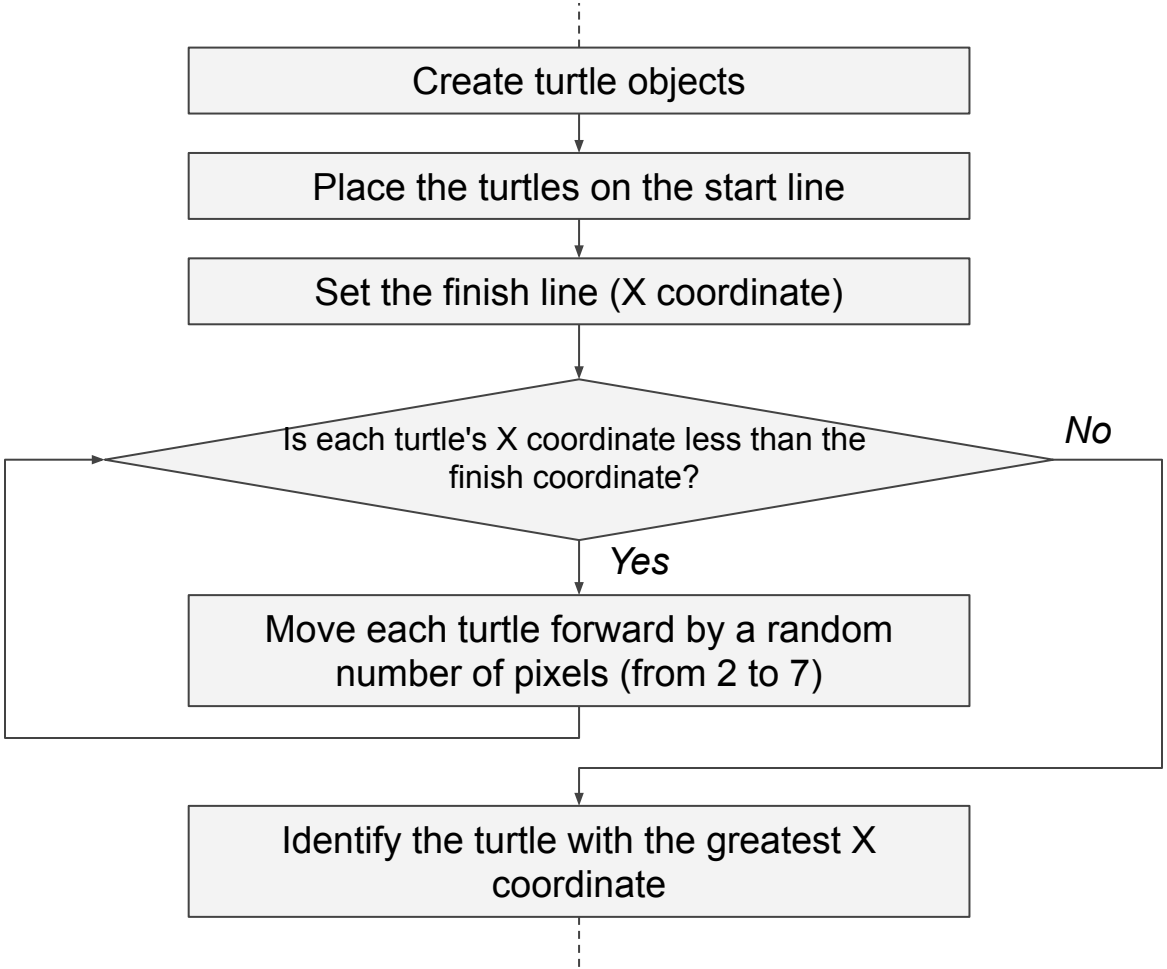


The speed is constant



The speed fluctuates

Brainstorming



Brainstorming

Turtle Races — the race

```
finish = 200
def startRace(t, x, y, color):
    t.color(color)
    t.shape('turtle')
    t.penup()
    t.goto(x, y)

t1 = Turtle()
t2 = Turtle()

startRace(t1, -200, -20, 'red')
startRace(t2, -200, 20, 'blue')

while t1.xcor() < finish and t2.xcor() < finish:
    t1.forward(randint(2,7))
    t2.forward(randint(2,7))

max_x = max(t1.xcor(), t2.xcor())
```

Get the turtles to the start.

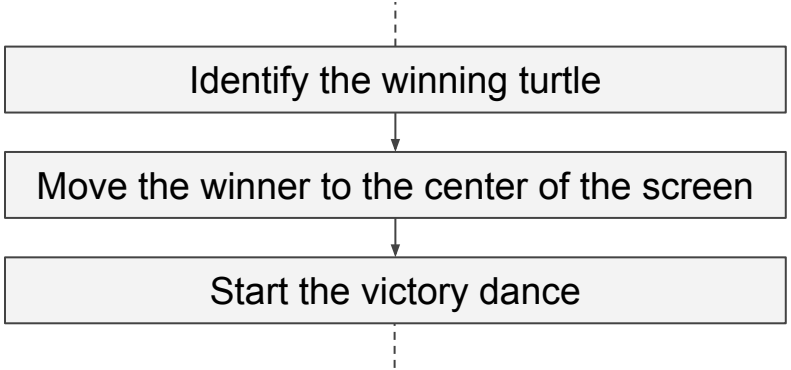
Each turtle speeds up and slows down randomly.

Brainstorming

Turtle Races — victory dance

You can use any victory dance for the prototype.

The more original the idea is, the more interesting it will be!



Brainstorming

Turtle Races — victory dance

Dance suggested by the developer Cole:

```
def dance(t):
    t.speed(15)
    t.left(randint(0, 90))
    j = 0
    while j < 8:
        t.penup()
        t.goto(0, 0)
        t.pendown()
        i = 1
        while i < 32:
            t.forward(i)
            t.left(i/2+5)
            i += 1
        j += 1
    t.penup()
    t.goto(0, 0)
```

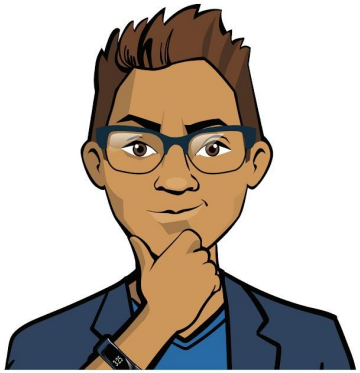
It is easy to program the dance in a separate function.

Just use the winning turtle as the ***argument.***

Brainstorming

Before we continue:

1. Which methods can be used to change the shape of the turtle? To find out the turtle's current X coordinate?
2. Which function in the Python standard library returns the greatest number of those given?
3. How do we add the rendering of markings to the program? What object will depict it?



Brainstorming