

# Python Start

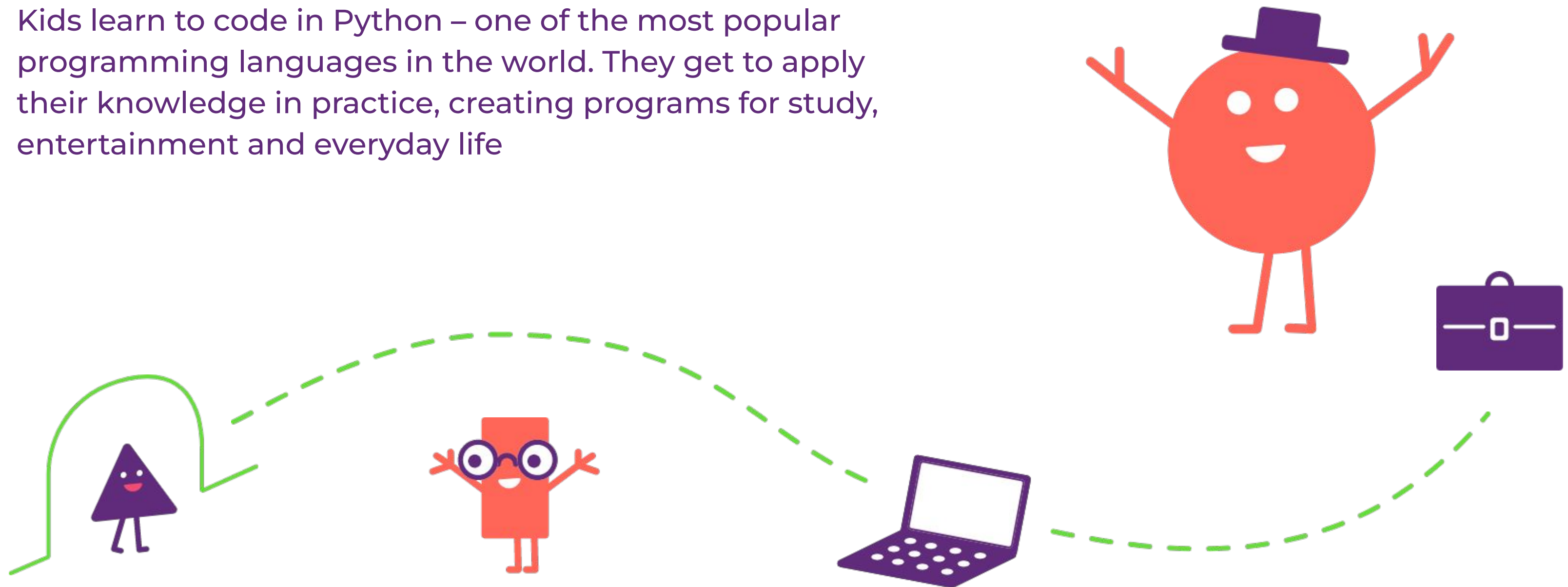
A course for kids aged 12-13

Fun programming  
in one of the world's  
most popular languages



# A solid start in programming

Kids learn to code in Python – one of the most popular programming languages in the world. They get to apply their knowledge in practice, creating programs for study, entertainment and everyday life



# Everyone will enjoy it!



## A deep dive into the process

The course uses a storyline about working in a real IT company, which helps us retain the attention of students who find concentrating difficult



## We nurture mathematical thinking

By studying the basic principles of programming, we deepen students's knowledge of math, even if they've never done well in the subject before



## We find their motivation

We don't do tests, instead we apply what we've learned in practice straight away, by creating projects and bringing our own little dreams to life

# In their first year of studies, **teens will learn to:**



- ◆ Understand the basics of algorithms and object-oriented programming
- ◆ Develop interactive graphical games for PCs using the PyGame library
- ◆ Work with graphics and use the Turtle library
- ◆ Solve real tasks using Python, and apply an iterative approach
- ◆ Apply the principles of project work when creating projects

# Course structure 1st year

## Module 1. Language basics

- Getting started with Python. Input-output functions.
- Variables. Numeric data types.
- Strings.
- Nested structures.

## Module 2. Control structures

- Logical data type. Conditional statement.
- Nested conditional statement and conditional statement for multiple branches.
- While loop. Counting loop.
- Nested algorithmic structures (loop within loop, conditional statement within loop, etc.)

## Module 3. Functions and modules

- Creating functions. Local and global scopes.
- Using one function within another.
- Modules. Using the built-in modules of the standard library.
- Creating modules. Connecting multiple modules.

## Module 4. Turtle module. Mathematics for the developer

- The Turtle graphical executor. The mathematics of the executor movement.
- Conditional statement. Drawing simple geometric shapes.
- Loop. Drawing polygons.
- Project lesson

## Module 5. Object-oriented programming

- Objects, their fields, and methods.
- Mouse and keyboard event handling.
- Classes. Class constructor
- Inheritance (from ready-made or own class).

## Module 6. Basics of game development using the PyGame

- Basics of game development. Sprite, game loop.
- Lists and methods of working with them. Iterating through the list in the for loop.
- Handling in-game events.
- Game physics.

## Module 7. A series of hackathons

- Workshop: Developer skills
- Hackathon. TestIT.
- Hackathon. Simple Paint.
- Grow Up Hackathon.

## Graduation

- Summing up and refreshing the learned topics while playing.

\* — lessons with a star are conducted for courses designed for 28, 32 and 36 classes;  
\*\* — lessons with two stars are conducted for courses designed for 32 and 36 classes.  
\*\*\* — lessons with three stars are conducted for courses designed for 36 classes.





# In their second year of studies, **teens** will **learn to:**

- ◆ Write and read code in Python and work with data structures
- ◆ Develop sophisticated games and apps for PCs using the PyGame and PyQt libraries
- ◆ Design interfaces
- ◆ Automate work with graphical files
- ◆ Work in a team and create projects from ideas before publicly presenting them



# Course structure 2nd year

## Module 1. Data structures

- Recap
- Data structures: lists and dictionaries
- Nested data structures
- Handling exceptions

## Module 2. Developing windowed applications

- Main widgets of a windowed application
- Designing an app interface
- Creating the professional computer app “Memory Card”

## Module 3. Working with files

- Recording and reading data from text files
- Recording and reading data from JSON files
- Creating the windowed application “Smart Cards” with record search by tags

## Module 4. Automatic image processing

- The PIL library (Python Imaging Library)
- Correcting the size, placement, brightness, color and contrast of images
- Mass processing of images

## Module 5. Advanced game development in PyGame\*

- Different types of sprites and their properties
- Using the physical properties of sprites
- Game event handling
- Designing a game: background and music

## Module 6. The developer’s portfolio\*

- Making creative programs and apps. Project presentation

\*Only in the full version of the course

# Why Python?

- It's one of the most popular and universal programming languages according to dozens of rating systems
- It's widely applied in various spheres of activity – from solving simple tasks to artificial intelligence and machine learning
- Knowledge of Python is in highly sought-after among job applicants to large IT companies, as well as other employers
- Programmers working in Python are some of the most highly paid specialists in the world





# Course storyline

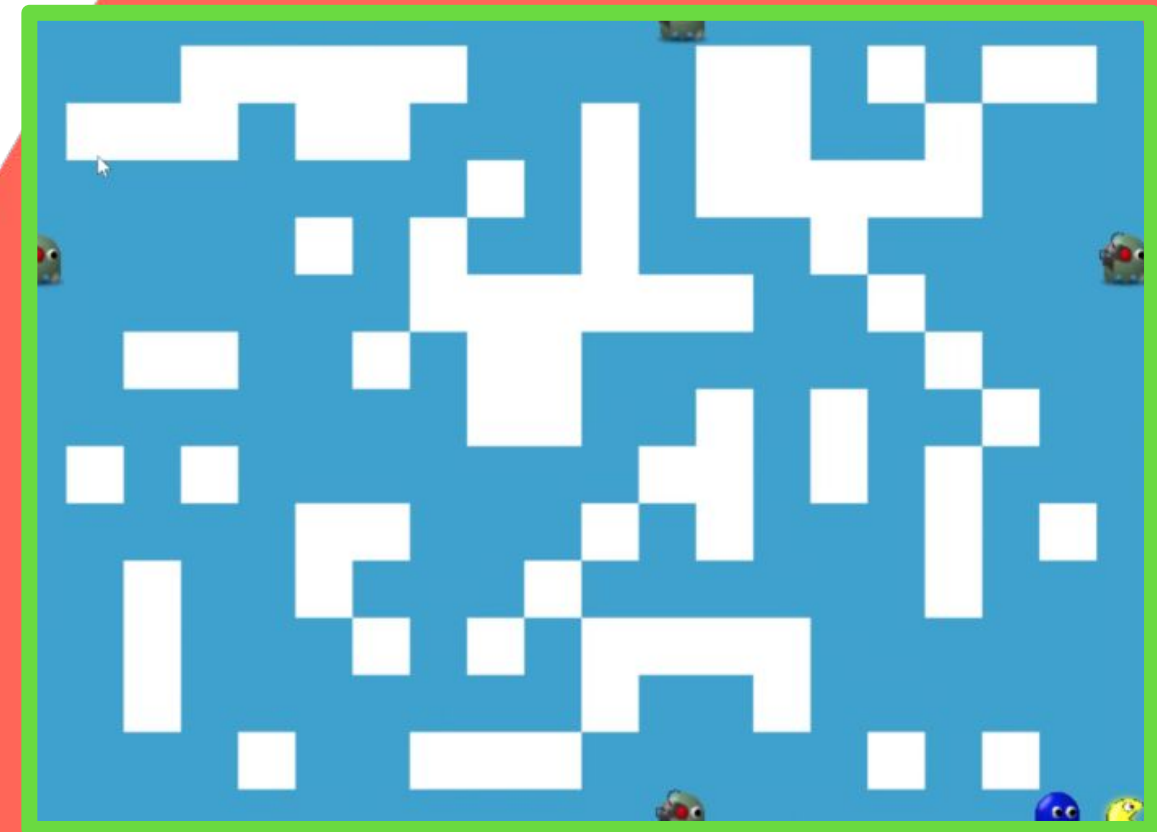
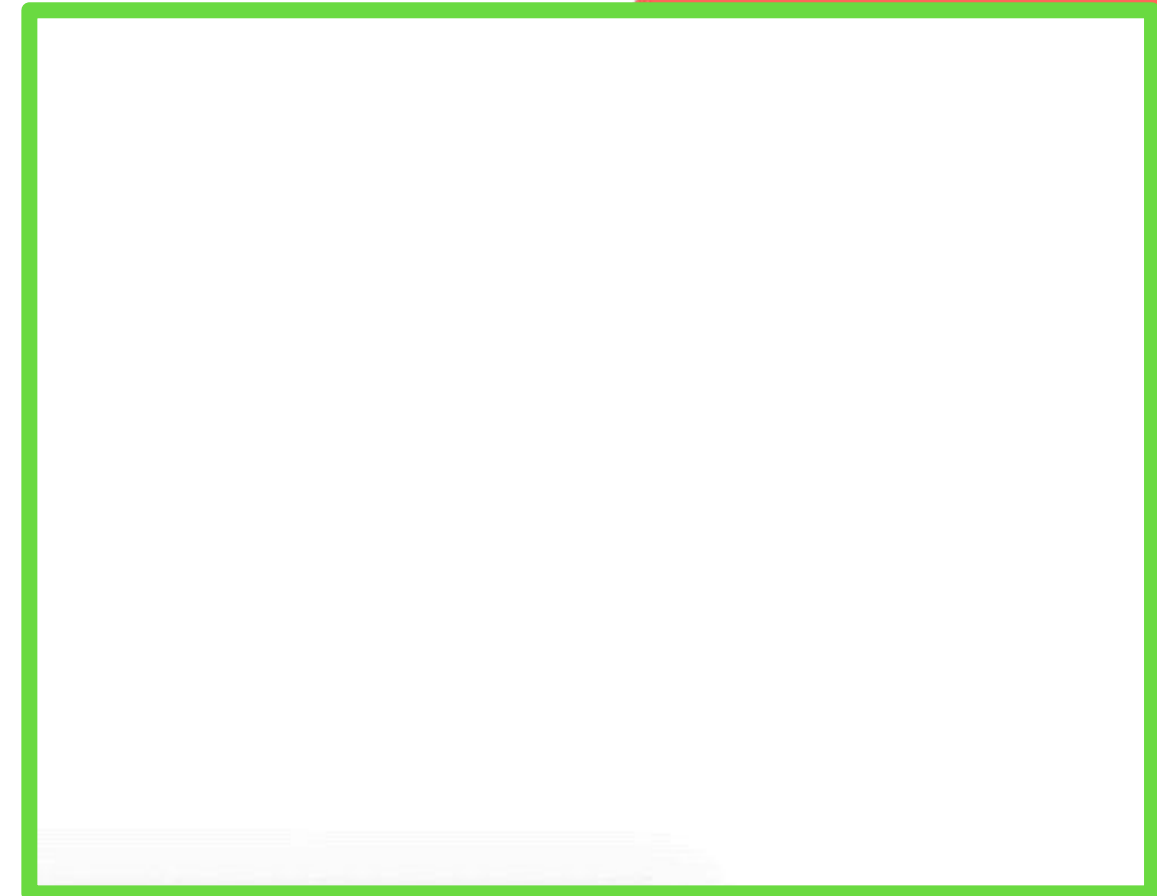
At the start of their studies, each student signs an improvised job contract making them a trainee at an IT company. They then move up the career ladder to the position of senior Python developer.

The storyline introduces teens to such concepts as **employers, projects, brainstorming, deadlines, time management, checklists** and **mind maps**, and ties together all the projects and tasks on the platform.



# A project-based approach

- ◆ Our kids create mini-projects right from their first lessons, applying the knowledge they've gained in practice
- ◆ Kids share their projects with their classmates directly in the platform, and learn to give and receive feedback
- ◆ At the end of each module, they present a full individual or group project



# What are our classes like?

- **Online** or **at the Algorithmics** school in your city
- In groups of **up to 10** online and up to **12** offline
- Classes last for **90 minutes** with a break in the middle
- **Once a week** ,  
2 academic years

The teacher explains the material in an interesting way and **gets the kids interested in the new topic**

Your kid won't ever fall behind in the program: **any classes they miss can be taken on the platform, 24/7**

You won't need to check any homework: at Algorithmics, **there are no obligatory homework tasks**

You'll be given **access to the platform** and will be able to follow your kid's progress

# How much does it cost?

starting at \_\_\_\_ per class

**4**  
**classes**  
Online

\_\_\_\_  
\_\_\_\_ per class

Offline

\_\_\_\_  
\_\_\_\_ per class

**18 classes**

Online

\_\_\_\_  
\_\_\_\_ per class

Offline

\_\_\_\_  
\_\_\_\_ per class

**5-10%**

**36**  
**classes**  
Online

\_\_\_\_  
\_\_\_\_ per class

Offline

\_\_\_\_  
\_\_\_\_ per class

**10-15%**



# Why do people choose Algorithmics?

- ◆ The curriculums for all our courses are developed by a team of professional **educators, pedagogues and psychologists**
- ◆ Algorithmics' **teachers** talk to the kids in understandable language, love their subject and know how to captivate children
- ◆ Our **IT learning platform** is 3 in 1: it's a smart task book, an environment for creating projects, and a community of shared interests



# Algorithmics

International School of Programming  
for children aged 6 to 18

😊 1 100 000 graduates













🚩 90+ countries

🏠 515 partners



# Courses for kids aged 6 to 18

Kids can start studying at Algorithmics at any age. At the end of the course, students can move straight on to the next one to continue studying in the new academic year

Course name:	Age:	6 — 7	8 — 9	10 — 11	12 — 13	14 — 15	16 — 18
Front-end Development							
Python Pro (2 years)							
<b>Python Start</b> (2 years)							
Game Development on Unity							
Building Websites							
Game Design							
Graphic Design							
Video Content Makers							
Visual Programming							
Summer Sessions							
Digital Literacy							
The Coding Knight							





Algorithmics

# Book a place in one of our groups

Please wait for our manager to call you and help you select a class start date that works for you

[\[link to the website\]](#)

