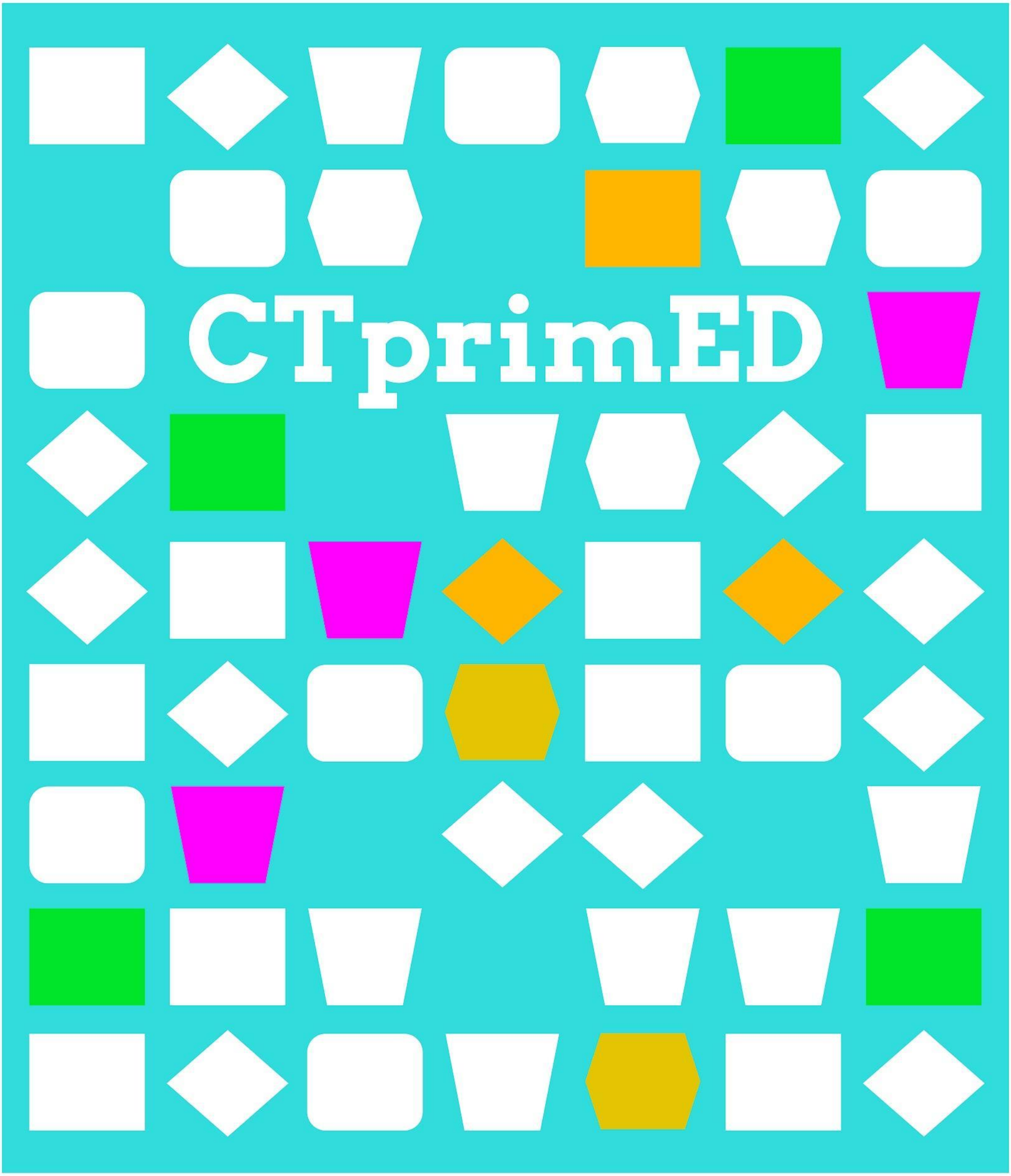


CTprimED

CT foundations - Introduction to Abstraction

Co-funded by the
Erasmus+ Programme
of the European Union

Summary

This activity explores the concept of Abstraction, which is one of the four foundations of Computational Thinking. Simply put, it is the process of taking away characteristics from something in order to reduce it to a set of essential characteristics.

Target group: All students in primary education can do this activity.

Duration: 30-50 minutes

Learning goals: The objective is to learn, in a practical way, the concept of Abstraction from Computer Science. It begins with an activity far removed from the computing field and then relates it to problem solving, computer programming or data structures.

Online/offline: offline

Computational Thinking:

- General skills: teamwork, creativity
- CT foundations: Abstraction
- CT concepts**:** none

Materials:

- Pencils and paper
- Cards with images or concepts (e.g. “Time's Up! Kids” game cards, or printable cards from appendix)

Preparation

Divide the class so that in each group there are 4 to 8 students around a table.
Each group needs somewhere they can draw (i.e. paper, blackboard, etc.) and pencils.
When applicable; print the cards from appendix 1.

General introduction to the four CT foundation lessons

Ask the students:

- What do you know about how computers and telephones work?
- Can they think for themselves? (Why or why not?)
- Who controls what a computer does?

Explain to the students that they are going to work on Computational Thinking. Simply put, this involves learning how to get a computer to solve a problem for you. It is not merely programming, but also, for example, learning how to break down a problem into pieces, or recognizing patterns so you can better solve a problem. There are four main foundations of CT:

- Decomposition→ dividing a problem into small pieces.
- Pattern recognition → looking for similarities or patterns within those small pieces that can help you solve the problem.
- Abstraction → distinguishing between the main and secondary issues. What is really important to solve the problem?
- Algorithms → coming up with step-by-step instructions to solve the problem.

In this lesson you will be introduced to pattern recognition.

Guess the image on the card

Introduction (5 min):
The group has to guess the image on the card that one of the members takes from the deck. In each round a different member of the group must take the card and draw so that the rest of the group can guess the image. Explain to the students that they can only use the following five shapes: triangle, square, rectangle, circle, and an ellipse.

Description of the lesson (10-20 min):
Explain the activity to the students. Either use the cards from appendix 1 or use other cards.

One student from the group takes a card from the deck and, without the others seeing it, tries to draw an Abstraction of that same image using only five of the following shapes: triangle, square, rectangle, circle and an ellipse). The remainder of the group must guess what the image depicts. Show the students the example below. The student sees the card featuring a cat and then draws the next image using two triangles, one circle and two rectangles.

The activity ends when everyone in the group has drawn at least one card.

Extra activity: To make it more difficult, you can limit the number of shapes the student can draw, by, for example, only allowing them to use three of the five shapes (it is possible to repeat a shape).

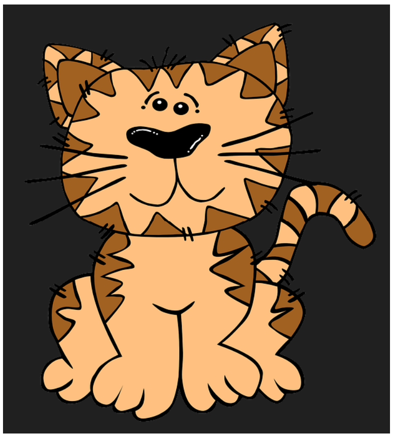


Fig. The card

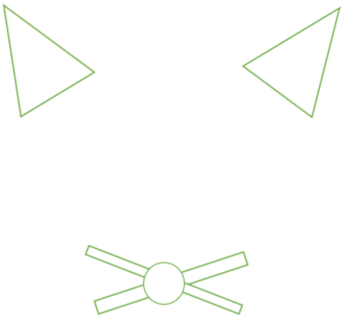


Fig. Image drawn by the member who took the card.

Conclusion (15-25 min)
Reflect with the students upon the concept of Abstraction.

Explain to the students that the limitations involved in using basic shapes to represent the images on the cards required them to extract the main and general characteristics of the image and leave out superfluous elements or details. This Abstraction Competence occurs when solving problems using Computational Thinking, since it is necessary to identify which variables or data are fundamental and which are either accessory or specific to a particular case. Similarly, the physical limitations of information systems also sometimes requires Abstraction Capacity to store a reduced set of data that faithfully represents the reality of what is to be digitized.

When designing computer programs, abstraction is essential to designing algorithms that solve problems in general and not, for example, only for a limited and known set of data.

EXTRA ACTIVITY: ABSTRACTING CHARACTERISTICS FROM A DETAILED DESCRIPTION

In the event that students find it difficult to understand Abstraction, this activity can help them to learn the basics of programming.

CHALLENGE 1

Ask the students to list the general characteristics that define a student. Give the students a few minutes to think about this individually, then discuss and agree on it in small groups, and finally share it in the large group (no discussion or agreement required).

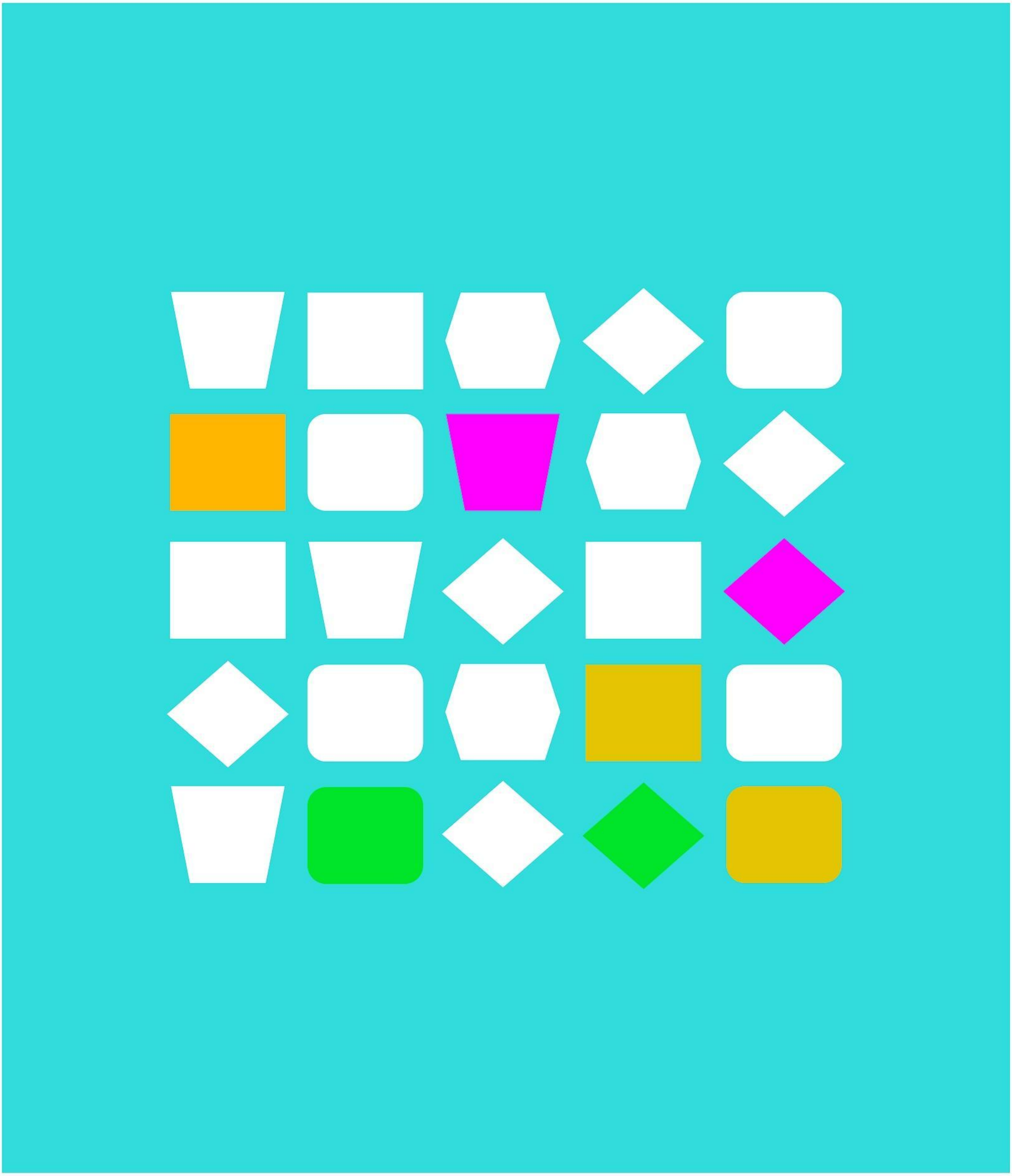
Possible characteristics listed by students:

1. Name
2. Surname
3. Date of birth
4. Course
5. Number of sisters or brothers
6. Favourite colour
7. ....

Reflect upon which characteristics are absolutely necessary to define a student. Can some be left out? This depends on the use of the data. For example, administrative staff at school do not need to know 5 and 6, but may need to know other characteristics (i.e. related to father, mother or legal tutor). The same occurs when we define data-structures in computer science in that the data needed depends on the context and scope of the problem.

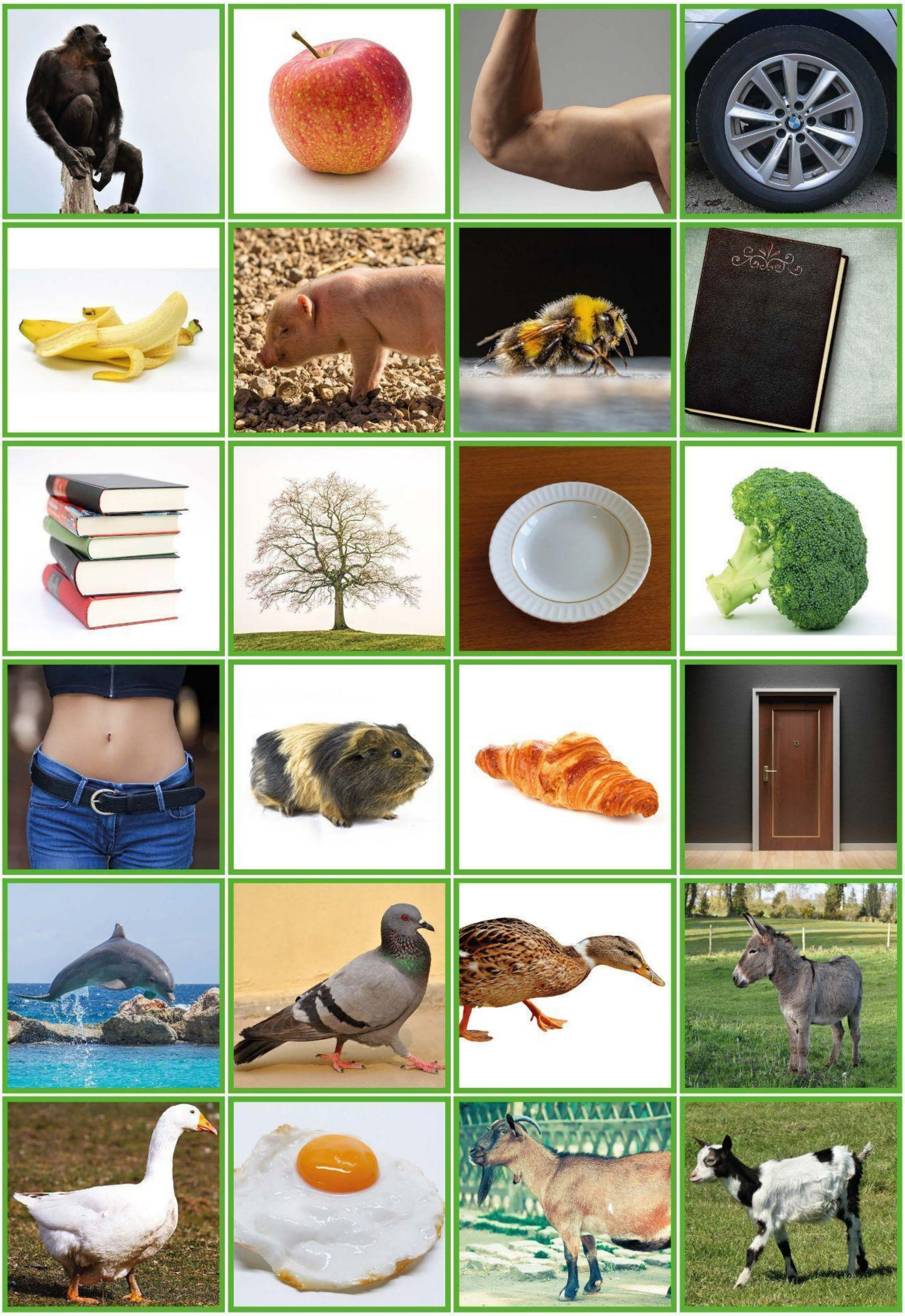
CHALLENGE 2

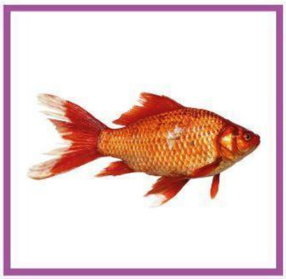
Ask the students to describe how they order objects in a room (e.g., their bedroom, the classroom, etc.). They should agree on the steps to follow to do so and realize that they leave out many details when doing so (e.g. it does not matter if the toy is a dog or a cat; its size matters more, as does the material it is made of, the weight, etc.). They abstract from the concrete characteristics to focus on those that are important for sorting.

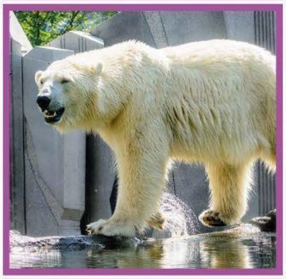


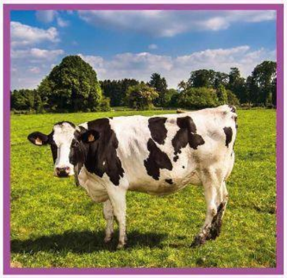
Appendix

Appendix 1: Printable cards

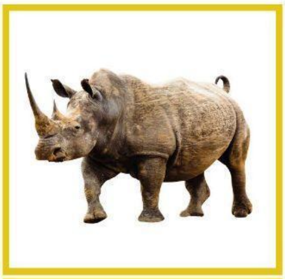
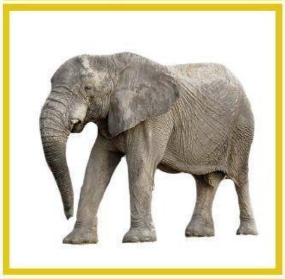


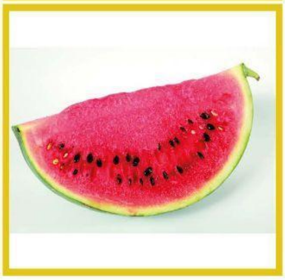






Münchner Neueste Nachrichten



Colophon

© CTPrimED

This publication is a product of CTPrimED (2021-1-NL01-KA210-SCH-000031319), funded with support from the Erasmus+ Programme of the European Union. This publication reflects the views of the authors, and the Commission cannot be held responsible for any use that may be made of the information contained therein.

Project Coordinator

NEMO Science Museum, The Netherlands

Partners

Universidad de la Iglesia de Deusto Entidad Religiosa, Spain
Stichting Children's Science Museum Curacao, Curacao

Co-funded by the Erasmus+ Programme of the European Union