Scratc\textsuperscript{2}h 2050 session Plans
Scratch 2050 Session Plans

1. Session plan Module 0: Setting Up a coding club
2. Session plan module 1-2 : Getting started with Scratch
3. Session plan module 3-4: Stories and animations
4. Session plan module 5 - Polygons & Flowers
5. Session plan module 6-7 - Games
Session plan Module 0: Setting up a coding club

**Objectives:**
- Get a common understanding of what a coding club is
- Get to know each other
- Reflect on aspirations and coding goals
- Align on practicals and set coding club rules

**Materials:**
- Powerpoint Module 0

**Duration:**
- +/- 1 hour

---

### Timing

<table>
<thead>
<tr>
<th>Engage Phase</th>
<th>15'</th>
</tr>
</thead>
</table>

### Teachers activities

**Introduction**
Ask the learners to introduce themselves. Ask them to share:
- Name and grade
- Their motivation
- What they expect from this coding club

**Why coding?**
Use the slide to illustrate why coding is important. Maybe use some elements that the learners highlighted in the introduction activity.

Explain what the coding club is and the principles. Use the slides in the PowerPoint of module 1.

**Coding club goals**
Ask the learners: What do you want to achieve after 12 weeks of coding?
- encourage the learners to write at least 3 things down
- It can help to give the instruction to complete this sentence: "After 12 weeks of coding I want to know/have/learned/understand/make..."
- Write the most common goals on a big poster/blackboard/wall

---

### Learners activities

- The learners take turns in introducing themselves
- The learners reflect on their aspirations and motivation.

---

### Competences

Communication and reflection skills.

---

### Suggestion

- take note of the expectations of the learners for later use.
Where and When
Discuss with the learners where and when the coding club will take place. Listen to the input of the learners and check if everyone agrees.

Representative
Discuss who will be the representative of the coding club. Explain that the tasks of the representative will be:
- facilitate the communication between club members and teachers.
- help in organising club activities
If there are many volunteers you can do a voting or rotate the function over different learners.

Club rules
In two groups, let the learners brainstorm for rules that will allow the club to function well. Give some examples such as being on time, being gentle with the computers,...

Suggestions:
- You can suggest to complete this sentence: “In order for our coding club to function well we all...”
- Write the most common rules on a big poster

Structure
End this session by explaining the structure of every coding club.
Now you are ready to start module 1-2!

Have fun!

- The learners agree on the time and dates of the coding club sessions.
- The learners agree on the chosen representative
- The learners discuss and brainstorm the coding club rules.
- The learners understand why club rules are important.
# Session plan module 1-2: Getting started with Scratch

## Objectives:
- Understand key principles of coding.
- Introduction to Scratch
- Develop the concept of computational creation through Scratch programming.
- Develop communication skills, critical thinking, problem solving, collaboration, and creativity skills.

## Materials:
- Computers and internet
- Powerpoint Module 1
- Scratch cards (digital or printed)

## Duration:
+/- 2 hours

## Timing

<table>
<thead>
<tr>
<th>Engagement Phase</th>
<th>Teachers activities</th>
<th>Learners activities</th>
<th>Competences</th>
</tr>
</thead>
<tbody>
<tr>
<td>20’</td>
<td>Ask the learners what you can use coding for. Ask for keywords. Write them down in a wordcloud. Place related keywords together.</td>
<td>The learners will think about what coding is.</td>
<td>Critical thinking, brainstorming and communication skills.</td>
</tr>
<tr>
<td></td>
<td><strong>Suggestions</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Use Mentimeter, a Word doc, a big paper or blackboard to make the wordcloud</td>
<td>- The learners have an understanding about what coding is</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- Show this aftermovie of the exposure day at Zorabots to link coding to the world of work.</td>
<td>- Learners think about what they can use coding for (make games, develop websites, calculations, make video`s, apps, robots...)</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Exploration Phase</th>
<th>Shoe Robot (Unplugged Activity)</th>
<th>Critical thinking, collaboration, computational, problem solving skills</th>
</tr>
</thead>
<tbody>
<tr>
<td>20’</td>
<td>Unplugged activities help students understand the basics of coding without touching a digital device themselves.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>In the “Shoe Robot” activity, students practice their skills to construct a code with a well-thought-out sequence. In this activity, as a facilitator, you are a robot and you are ‘programmed’ by the students, the programmers. The goal of the programmers is to make the robot successfully put on his shoe.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Put off your shoe. Then explain to the students that you are a robot and want to put on your shoe. However, you are not programmed to do that. The students are the programmers or engineers and have to make an instruction or code that you can follow.</td>
<td></td>
</tr>
</tbody>
</table>
Divide the learners in 2 groups. Give them a few minutes to discuss and write down what they think is the right instruction to make a robot put on his/her shoe. Then they can tell you the code and you, as a robot, carry out their commands.

**Warning**: a robot takes everything very literally. When the students instruct you to pick up the shoe, pick it up upside down. When they tell you to put your foot in it, do it in the wrong direction (or to the wrong foot). Make lots of “mistakes” so they understand how accurate programming is.

**Example**: [https://www.youtube.com/watch?v=2ZVbWu1i5pM](https://www.youtube.com/watch?v=2ZVbWu1i5pM)

Ask your students what the conclusion is of this exercise.

### Teachers activities

<table>
<thead>
<tr>
<th>Timing</th>
<th>Teachers activities</th>
<th>Learners activities</th>
<th>Competences</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Divide the learners in 2 groups. Give them a few minutes to discuss and write down what they think is the right instruction to make a robot put on his/her shoe. Then they can tell you the code and you, as a robot, carry out their commands.</td>
<td>The learners discuss and write down the instructions to make a robot put on a shoe. When they are ready one learner reads the instruction to the robot. the learners understand the key principles of coding learners make a conclusion of this exercise: Ask your students what the conclusion is of this exercise. It can be something like this: Computers are not clever, but they are very obedient. They will do exactly what you tell them to do. Coding is the art of instructing a computer what to do. Programming languages are used to communicate to a computer what you want it to do. Coding involves feeding the computer step-by-step commands.</td>
<td>Critical thinking and communication skills</td>
</tr>
<tr>
<td></td>
<td><strong>Warning</strong>: a robot takes everything very literally. When the students instruct you to pick up the shoe, pick it up upside down. When they tell you to put your foot in it, do it in the wrong direction (or to the wrong foot). Make lots of “mistakes” so they understand how accurate programming is.</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Example</strong>: <a href="https://www.youtube.com/watch?v=2ZVbWu1i5pM">https://www.youtube.com/watch?v=2ZVbWu1i5pM</a></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ask your students what the conclusion is of this exercise.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Demonstration of Scratch

Show your students what Scratch is before you dive into it. You can use the PowerPoint of module 1 and/or share your own screen to showcase some moves in Scratch. You can also show an introductory YouTube video. There are plenty you can use, for example [http://bit.ly/intro-scratch](http://bit.ly/intro-scratch).

Ask the learners to observe. Maybe you can ask some questions:
- What did you see? Does it look difficult?
- (in case you showed a project of yourself) how did it work? What did I do?
- do you see similarities with the shoe robot?

### Demonstration

**Phase**

20’
### Experimenting Phase

50'

Now your students are ready to start coding in the most fun way possible: by playing and experimenting.

Divide the learners in groups (2-3 learners). Hand out the set of **Scratch starter cards**, each containing a simple code such as moving with the keyboard arrows, moving with a mouse, changing the color of a character, and so on. Challenge your students to try out as many cards as possible. You can download the cards here: [https://resources.scratch.mit.edu/www/cards/en/scratch-cards-all.pdf](https://resources.scratch.mit.edu/www/cards/en/scratch-cards-all.pdf)

**How to facilitate this experimenting phase:**
- Walk around and help the students by asking questions to get them to the right answer.
- Let groups help each other. Eg: the ‘faster’ groups can assist the ‘slower’ ones.
- After some time, encourage the groups to experiment (outside the starter cards)
- See a great idea? Ask the creator to share with others.

### Reflection Phase

10'

Ask the learners what they have discovered today and what they would like to do in the next coding club.

**Guiding questions:**
- What do you like the best about the project you made?
- What did you learn?
- What was the hardest part?
- What would you like to improve still?
- What do you expect from this club?

---

### Teachers activities

- The learners play around and experiment with Scratch. They can do this individually or in groups.
- The learners use the Scratch starter cards to explore functionalities.

### Learners activities

- The learners share and reflect on the first coding club session.
- The learners express their aspirations for the next coding club session.

### Competences

- Collaboration, creativity
- Computational, problem solving, coding skills
- Reflection, communication skills

---

**Scratch lesson plan - First Coding Club Session**

---

**Timing**

<table>
<thead>
<tr>
<th>Experimenting Phase</th>
<th>Teachers activities</th>
<th>Learners activities</th>
<th>Competences</th>
</tr>
</thead>
<tbody>
<tr>
<td>50'</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

**Teachers activities**

- Now your students are ready to start coding in the most fun way possible: by playing and experimenting.

**Learners activities**

- The learners play around and experiment with Scratch. They can do this individually or in groups.
- The learners use the Scratch starter cards to explore functionalities.

---

**Competences**

- Collaboration, creativity
- Computational, problem solving, coding skills
- Reflection, communication skills
Session plan module 3 - 4 - Stories and animations

Objectives:
- Making animations and stories with Scratch
- Use computational concepts in storytelling
- Explore computational creation
- Develop communication skills, critical thinking, problem solving, collaboration, and creativity skills.

Timing

Engage Phase
20'

Teachers activities
- Ask the learners what their favorite animation movie is and open up a class discussion:
  - What makes your favorite movie a good movie?
  - What do your favorite movies have in common?
  - What characteristics make up an animation movie?
  - How are animation movies developed?

Competences
Critical thinking, brainstorming and communication skills.

Learners activities
- The learners understand the characteristics of an animated story (different characters, storyline...) and realize that coding is needed to make animated stories.
- Learners realize they can use Scratch for developing an animated story.

Materials:
- Computers
- Internet
- PowerPoint module 2

Duration:
+/- 2.5 hours

Demonstration Phase
20'

Show your students how they can use Scratch for developing an animated story. You can do that by using the PowerPoint of module 2 and by showing an example of an animated video (you can find them on the Explore page of Scratch).

Ask the learners to observe. Maybe you can ask some questions:
- What did you see? Does it look difficult?
- (In case you showed an example) how did it work?

Objectives:
- Making animations and stories with Scratch
- Use computational concepts in storytelling
- Explore computational creation
- Develop communication skills, critical thinking, problem solving, collaboration, and creativity skills.

Materials:
- Computers
- Internet
- PowerPoint module 2

Duration:
+/- 2.5 hours

Competences
Critical thinking, brainstorming and communication skills.

Learners activities
- The learners observe the demonstration and take notes of the following questions:
  - What did you see? Does it look difficult?
  - (In case you showed an example) how did it work?
  - What did I do?
Walk around and help the students by asking questions to get them to the right answer.
Let groups help each other. Eg: the ‘faster’ groups can assist the ‘slower’ ones.
See a great idea? Ask the creator to share with others

What was difficult about being able to use only 10 blocks?
What was easy about being able to use only 10 blocks?
How did it make you think of things differently?

**Story with 10 blocks**
Give students time to create a project with only these 10 Scratch blocks: go to, glide, say, show, hide, set size to, play sound until done, when this sprite clicked, wait, and repeat. Remind students to use each block at least once in their project and encourage them to experiment with different sprites, costumes, or backdrops.

**How to facilitate this experimenting phase:**
- Walk around and help the students by asking questions to get them to the right answer.
- Let groups help each other. Eg: the ‘faster’ groups can assist the ‘slower’ ones.
- See a great idea? Ask the creator to share with others

After some time wrap up the exercise. **Ask the learners:**
- What was difficult about being able to use only 10 blocks?
- What was easy about being able to use only 10 blocks?
- How did it make you think of things differently?

The conclusion could be that it is surprising how much one can do with just 10 blocks! Take this opportunity to encourage different ideas and celebrate creativity by inviting a few students to present their projects in front of the class.
Experimenting Phase

50'

Animation video with at least 10 sprites
Let the learners create an animated story with at least 10 sprites. Encourage the students to:
- brainstorm ideas with a neighbor
- sketch the story/animation outline/ideas on paper first
- experiment with different blocks and costumes
- try adding sound to their project.

When the learners are ready invite them to share their projects to a feedback group. A feedback group is a small group of learners who share ideas and projects-in-progress with one another in order to get feedback and suggestions for further development. Divide students in smaller groups of 3-4 people. In these feedback groups, ask the learners to take turns sharing their animated story project. Encourage students to record notes, feedback, and suggestions.

Reflection Phase

20'

Ask the learners what they have discovered today and what they would like to do in the next coding club.

Guiding questions:
- What do you like the best about the project you made?
- What did you learn?
- What was the hardest part?
- What would you like to improve still? Was the feedback group useful?
- Was it difficult or easy to give/receive feedback? Why (not)?
- What do you expect from this club?
Session plan module 5 - Polygons & Flowers

Objectives:
- Apply the computational concepts of shapes and repeat control.
- Know how to draw angles and use the pen function.
- Develop communication skills, critical thinking, problem solving, collaboration, and creativity skills.

Materials:
- Computers
- Internet
- PowerPoint module 5

Duration: +/- 2 hours

Timing

Engage Phase
30’

Teachers activities
Explain to the learners that today coding session is all about creating your own sprite by drawing. To be able to draw, we need to be able to recognize polygons. Ask the students what a polygon is.

Compile a definition together with the learners. It could look like this:

a polygon is a plane figure with at least three straight sides and angles.

Ask the learners to sketch some of the polygons they know.

Fill in the table in in the PowerPoint. Looking at the completed table, ask the students what strikes them. If the learners don’t point it out, explain that if they divide 360 by the number of sides in a polygon they would find the angle of the polygon. In this way they can draw a wide range of things with Scratch.

Learners activities
- The learners understand what polygons are.
- The learners understand that polygons are useful in drawing sprites in Scratch
- Learners realize they can calculate the angles of polygons.

Competences
Critical thinking, brainstorming and communication skills.

Engage Phase
30’

Teachers activities
Explain to the learners that today coding session is all about creating your own sprite by drawing. To be able to draw, we need to be able to recognize polygons. Ask the students what a polygon is.

Compile a definition together with the learners. It could look like this:

a polygon is a plane figure with at least three straight sides and angles.

Ask the learners to sketch some of the polygons they know.

Fill in the table in in the PowerPoint. Looking at the completed table, ask the students what strikes them. If the learners don’t point it out, explain that if they divide 360 by the number of sides in a polygon they would find the angle of the polygon. In this way they can draw a wide range of things with Scratch.

Learners activities
- The learners understand what polygons are.
- The learners understand that polygons are useful in drawing sprites in Scratch
- Learners realize they can calculate the angles of polygons.

Competences
Critical thinking, brainstorming and communication skills.
### Scratch lesson plan - Module 2

<table>
<thead>
<tr>
<th>Timing</th>
<th>Teachers activities</th>
<th>Learners activities</th>
<th>Competences</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Demonstration Phase</strong>&lt;br&gt;20’</td>
<td>Show your students how they can draw angles and use the pen function in Scratch. You can do this by using the PowerPoint of module 5 and by showing an example.&lt;br&gt;&lt;br&gt;Ask the learners to observe. Maybe you can ask some questions:&lt;br&gt;• What did you see? Does it look difficult?&lt;br&gt;• How did the game work?</td>
<td>The learners observe the demonstration and take notes of the following questions:&lt;br&gt;• What did you see? Does it look difficult?&lt;br&gt;• How did the game work? What did I do?</td>
<td>collaboration, creativity, computational, problem solving, coding skills, communication skills</td>
</tr>
<tr>
<td><strong>Experimenting Phase</strong>&lt;br&gt;30’</td>
<td><strong>Drawing Angles</strong>&lt;br&gt;Let the learners try to draw the polygons they listed in the engage phase. This requires them to know the degrees of each polygon. Remind them they can have a look at the table.&lt;br&gt;&lt;br&gt;After a while encourage them to draw more complex things such as a circle and a flower.</td>
<td>Learners design (individually or in pairs/groups) different polygons. They discover the possibilities of drawing in Scratch</td>
<td>collaboration, creativity, computational, problem solving, coding skills, communication skills, feedback skills</td>
</tr>
<tr>
<td><strong>Experimenting Phase</strong>&lt;br&gt;30’</td>
<td><strong>Design your own Sprite</strong>&lt;br&gt;Let the learners design their own sprite by using pen blocks.&lt;br&gt;&lt;br&gt;Encourage the learners to:&lt;br&gt;• Add a trigger (event)&lt;br&gt;• Add the “Erase all”&lt;br&gt;• Hide and show&lt;br&gt;• Change color</td>
<td>Learners design (individually or in pairs/groups) a sprite.&lt;br&gt;When done learners show their project in their feedback group. Let learners gather feedback by having their feedback group members respond to the following questions:&lt;br&gt;• What is something that doesn’t work or could be improved?&lt;br&gt;• What is something that is confusing or could be done differently?&lt;br&gt;• What is something that works well or you really like about the project?</td>
<td>collaboration, creativity, computational, problem solving, coding skills, communication skills, feedback skills</td>
</tr>
</tbody>
</table>
Reflection Phase
10'

The learners share and reflect on the coding club session.
The learners express their aspirations for the next coding club session.
Reflection, communication skills

Ask the learners what they have discovered today and what they would like to do in the next coding club.

Guiding questions:
• What was challenging about this coding club session?
• What are you proud of?
• What was the main feedback you got from your fellow coders?
• What would you like to improve still?

When the learners are ready invite them to share their projects to a ‘feedback group’. A feedback group is a small group of learners who share ideas and projects-in-progress with one another in order to get feedback and suggestions for further development. Divide students in smaller groups of 3 - 4 people. In these feedback groups, ask the learners to take turns sharing their games. Give the learners time to improve the games using the feedback.

Scratch lesson plan - Module 2
Session plan module 6-7 - Games

**Objectives:**
- Apply the computational concepts of conditionals, operators, and data.
- Apply the computational practices of iterating, testing, debugging, and reusing, abstracting, and modularizing by building game projects.
- Identify and use common game mechanics.
- Develop communication skills, critical thinking, problem solving, collaboration, and creativity skills.

**Timing**

<table>
<thead>
<tr>
<th>Engage Phase</th>
<th>Teachers activities</th>
</tr>
</thead>
</table>
| 20’          | Divide students into small groups of 2-3 people. In their small groups, ask students to generate a list of games that they enjoy playing. Give students a short time period (1-2 minutes) to write down as many games as they can. Then, have students narrow down their favorites from the brain dump list. After a few minutes, facilitate a class discussion about what characteristics make up a game and generate a class list of common game mechanics. Ask:  
  - What do the games have in common?  
  - What features of their design make them a game?  
Next, ask students to imagine their dream game and write a list of design elements for that game. Invite students to share their dream game lists in their small groups to get feedback and suggestions. |

**Materials:**
- Computers
- Internet
- PowerPoint module 6-7

**Duration:**
- +/- 2.5 hours

**Learners activities**
- The learners understand the characteristics of a game
- Learners realize they can use Scratch for developing a game.
- Learners make a list of their favorite design elements
- Learners share their dream game list in their groups to get feedback and suggestions.

**Competences**
- Critical thinking, brainstorming and communication skills.
Show your students how they can use Scratch for developing a game and what the design process is. You can do that by using the PowerPoint of module 6-7 and by showing an example of a game such as Maze, Pong, and Scrolling (you can find them in the Game studio).

Ask the learners to observe. Maybe you can ask some questions:
- What did you see? Does it look difficult?
- How did the game work?

**Demonstration Phase**
- **Timing**: 20’

**Experimenting Phase**
- **Timing**: 60’

**Design your Dream Game**
Let the learners create their dream game. As a first step let them first fill in the game design template here. You can suggest to the learners to have a look at some examples in the Game studio such as Maze, Pong, and Scrolling.

Encourage the learners to:
- Add multiple levels to their game! This can be done through the use of different backdrops and using broadcast blocks to trigger the next level.
- Add difficulty to the game by creating different levels, using a timer, or keeping score.
- Use the ‘make’ a variable block to keep score!
- Experiment with timer blocks to add new challenges.

Learners create (individually or in pairs/groups) a game. They can start building their own games or remix one of the starter projects in the Games Studio.

When done learners show their project in their feedback group. Let learners gather feedback by having their feedback group members respond to the following questions:
- What is something that doesn’t work or could be improved?
- What is something that is confusing or could be done differently?
- What is something that works well or you really like about the project?
Experimenting Phase

30'

Reflection Phase

20'

Game Fair
To celebrate and share final game creations, we recommend hosting a Game fair. Final game projects are placed in presentation mode; students walk around and play each other's games.

Half of the students stay in their seats with their projects open while the other half walks around trying out the games, asking questions, and giving feedback, then switch sides.

Ask the learners what they have discovered today and what they would like to do in the next coding club.

Learners showcase and test each other's game and give feedback.

Guiding questions:
- What was challenging about designing your game?
- What are you proud of?
- What was the main feedback you got from your fellow coders?
- What would you like to improve still?

- The learners share and reflect on the coding club session.
- The learners express their aspirations for the next coding club session.

Reflection, communication skills

collaboration, creativity computational, problem solving, coding skills, communication skills

Learners activities

Competences
This project was funded by Belgium through the Wehubit programme implemented by Enabel