

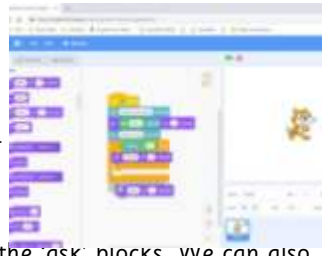
Computing

Selection and repetition in algorithms

Overview

Creating Quizzes in Scratch

Programming means giving a computer a set of instructions to follow so it can carry out specific tasks. Scratch is a fun and interactive tool that allows us to create our own quizzes, games, animations, and stories using code. When making a quiz in Scratch, we can add questions by using the 'ask' blocks. We can also use selection and conditional statements to make our quizzes respond differently depending on the user's answers. To make our projects work smoothly, we write algorithms – step-by-step instructions – that control what happens and when, such as moving characters, playing sounds, or showing messages.



Scratch basics

Scratch is divided into three main sections



On the left side, you'll find the Blocks Palette.

This is where all the coding blocks are stored. These blocks are used to give instructions to the characters and control what happens in the project.

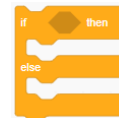
In the centre, there's the Coding Area. This is where we drag and drop the blocks to put them together.

On the right side, you'll see the Stage and the Sprite. The stage is where we can watch our program come to life, and the sprite is the main character or object that performs the actions based on our code.

Conditions in Scratch

In Scratch, we can use condition blocks to make our programs respond to different inputs or events.

The 'if-then' block, found in the dark orange Control section, lets us set up a condition. We place other blocks inside it to define what should happen if that condition is true. To create a condition, we often use Sensing blocks (light blue), which act as triggers — for example, detecting when a certain key is pressed. You can choose a different key or action by clicking the small arrow in the sensing block and selecting from the list. Then, we add action blocks such as motion or sound to control what the sprite does when the trigger is activated.



If we want the program to do one thing when the condition is met and something else when it isn't, we use the 'if-then-else' block. This allows us to add two different sets of instructions: one set goes under 'then' (for when the condition is true), and the other goes under 'else' (for when the condition is false — like when another key is pressed).

The 'forever' block is used to make a set of commands repeat without stopping. This is useful when we want the program to keep checking for conditions or performing actions continuously.



Asking Questions

In Scratch, we can ask the user questions by using the 'ask' block, which allows us to collect input during the program.

If we're expecting a particular answer, such as "yes" or "no", we can check this by combining the 'answer' block from the Sensing section with the '=' block found in the Operators category. To do this, drag the 'answer' block into the first space of the operator, then type the correct response (e.g. "yes") into the second space. This helps the program know what to check for.

Once the answer is evaluated, we can use the 'say' block from the Looks section to display a message on screen. This is a great way to give feedback, such as letting the user know if their answer was correct or incorrect.

Key Questions

What is a sequence error?

A command is in the wrong place

What is a keying error?

A command has been typed incorrectly

What is a logic error?

There is a mistake in your plan

Which block creates a condition for the program to act upon?

If-then blocks

What is a count-controlled loop?

A command that repeats a set number of times.

Key Vocabulary

Selection, Condition, True, False

Count-Controlled Loop, Outcomes, Conditional Statement