Scratch – Jigsaw method

Motion and Drawing

Worksheets provide guidance throughout the program creation.

Mind the following symbols that..

- structure your work progress and show subgoals.
- provide help, mark and explain challenging and important notes.
- include assignments and activities.

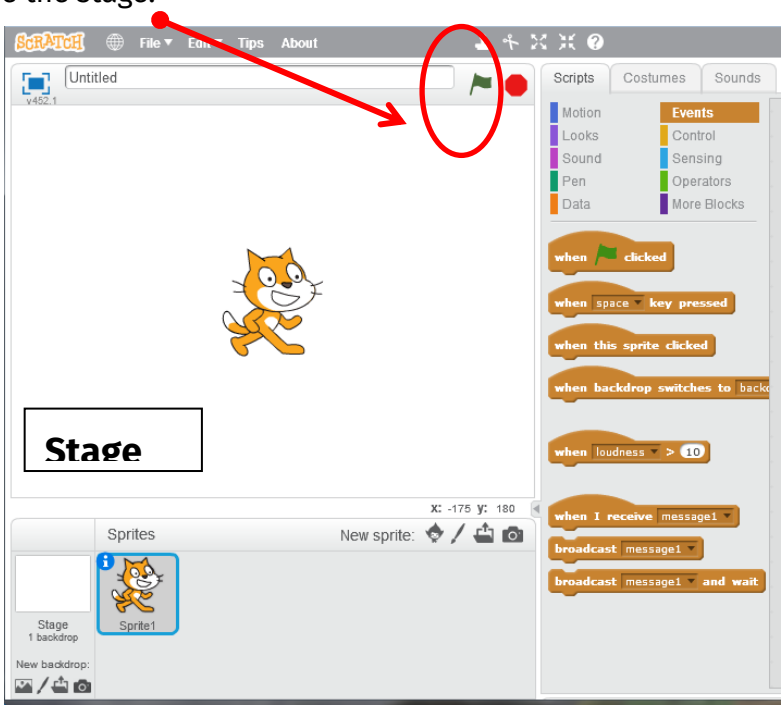
Jigsaw method: Motion and Drawing

SCRATCH shows you how to program figures, which are later on animated and able to act ( ).

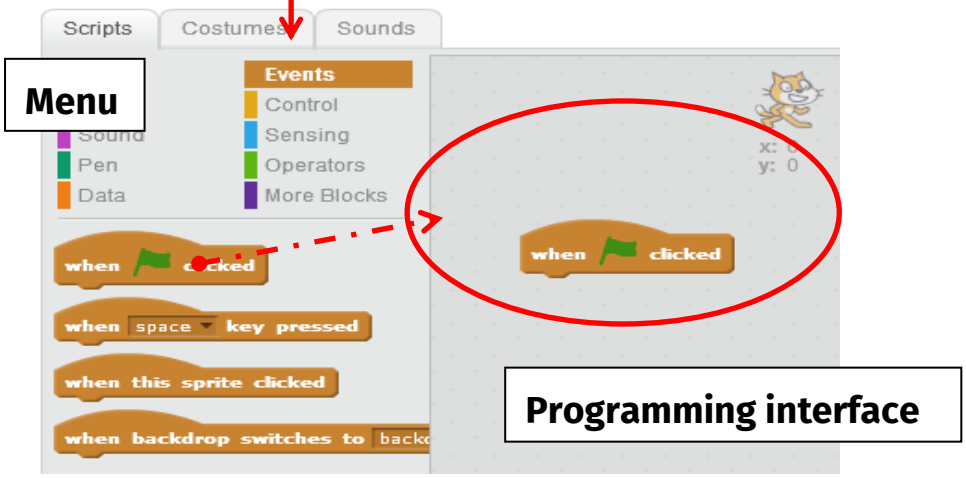
Curious? Let's get started!

Let the cat run:

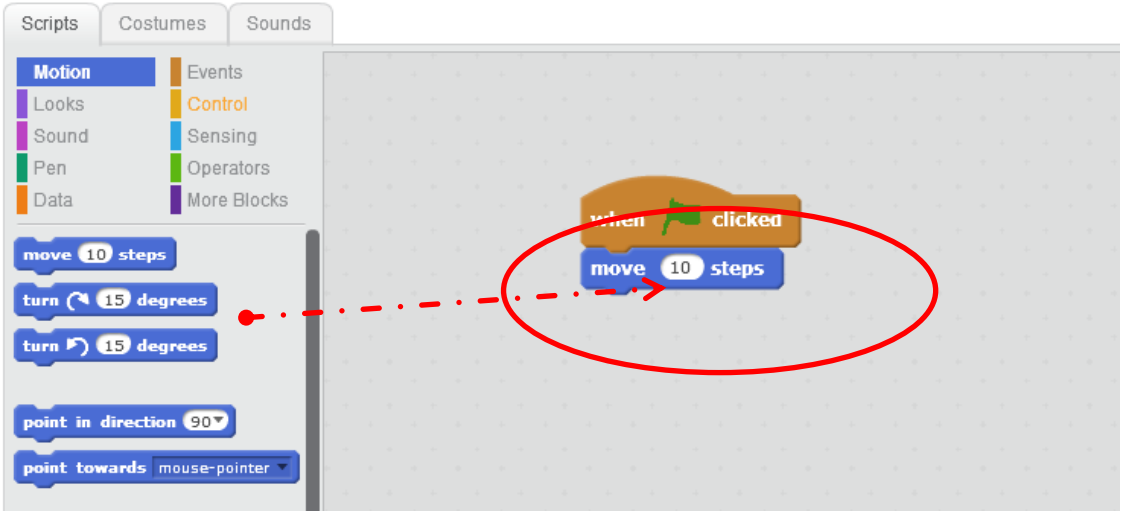
In our program, the cat is supposed to run whenever you click the little green flag above the stage.



In order to determine the **start conditions**, click the drop-down button “Scripts”, select **Events** and choose the “when-little-green-flag-is-clicked-on”- mask. You can simply drag the mask into the **programming interface**.



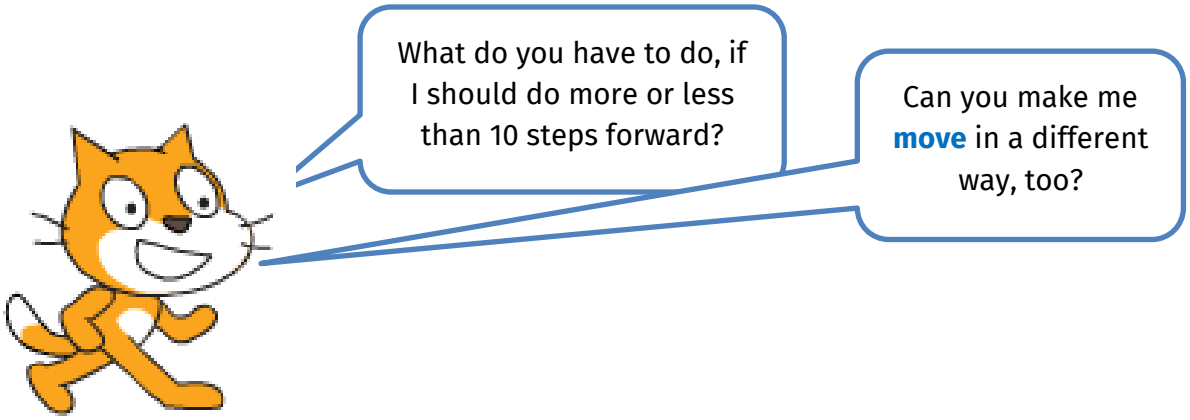
Now choose "Scripts" from the drop-down menu and select **Motion**. Choose the **function** "go-10-step" and drag it exactly under your starting condition (so that the puzzle pieces snap into each other).



Test your **program** by clicking on the *little green flag* above the stage.

You can always move the cat to any position within the stage after it has run.

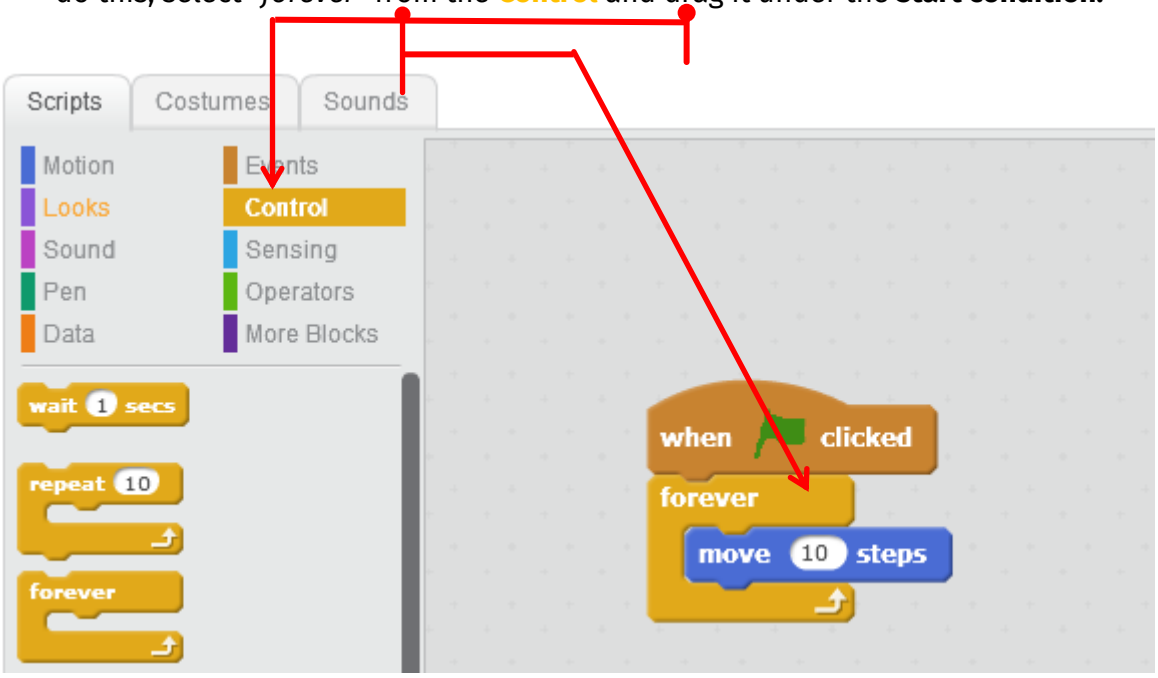
Motion and Drawing



- Write it down: 

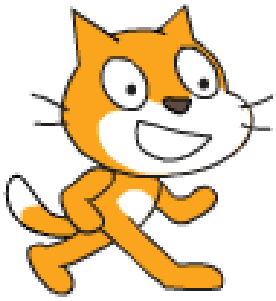
!
- Then, test your program. 

Now let the cat  run on the **stage** to the right edge. Use the function **loop**, so that your program does not have to run until the cat reaches the edge. In order to do this, select "*forever*" from the **Control** and drag it under the **start condition.**



Scratch – Jigsaw method

Motion and Drawing



How long will I do **10 steps** now?

Do I stop at the edge of the stage automatically?
What needs to be done for me to stop?

What do you think? List your considerations here:

If the cat now touches the edge of the **stage**, it should bounce off, change direction and continue.

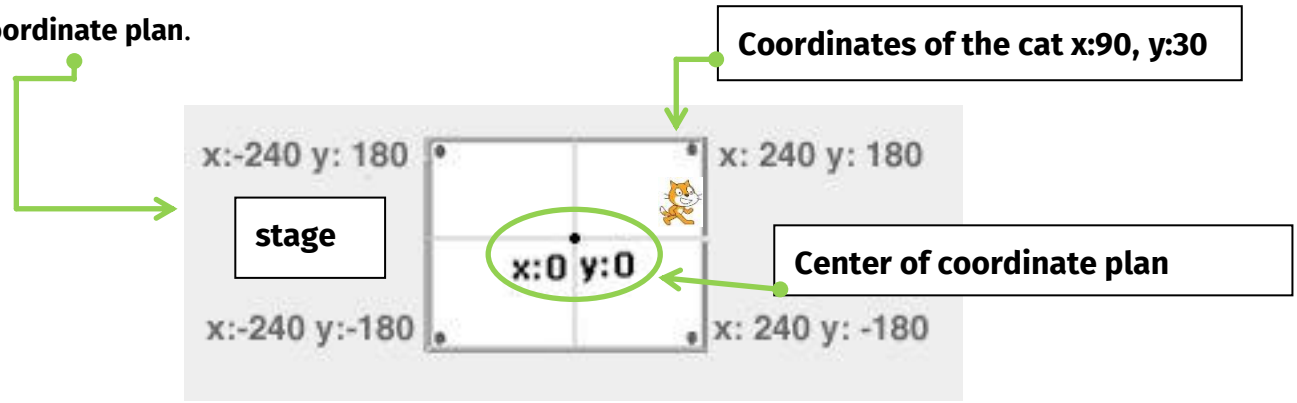
- Think about with what puzzle pieces from the **Motion** function you can let the cat changes its direction before it reaches the edge of the stage.
- Put your thoughts into practice and test your program.

The cat should now be moved with the **arrow keys**. Create a new program for this!
Attention: Now we have to change the **starting condition**!

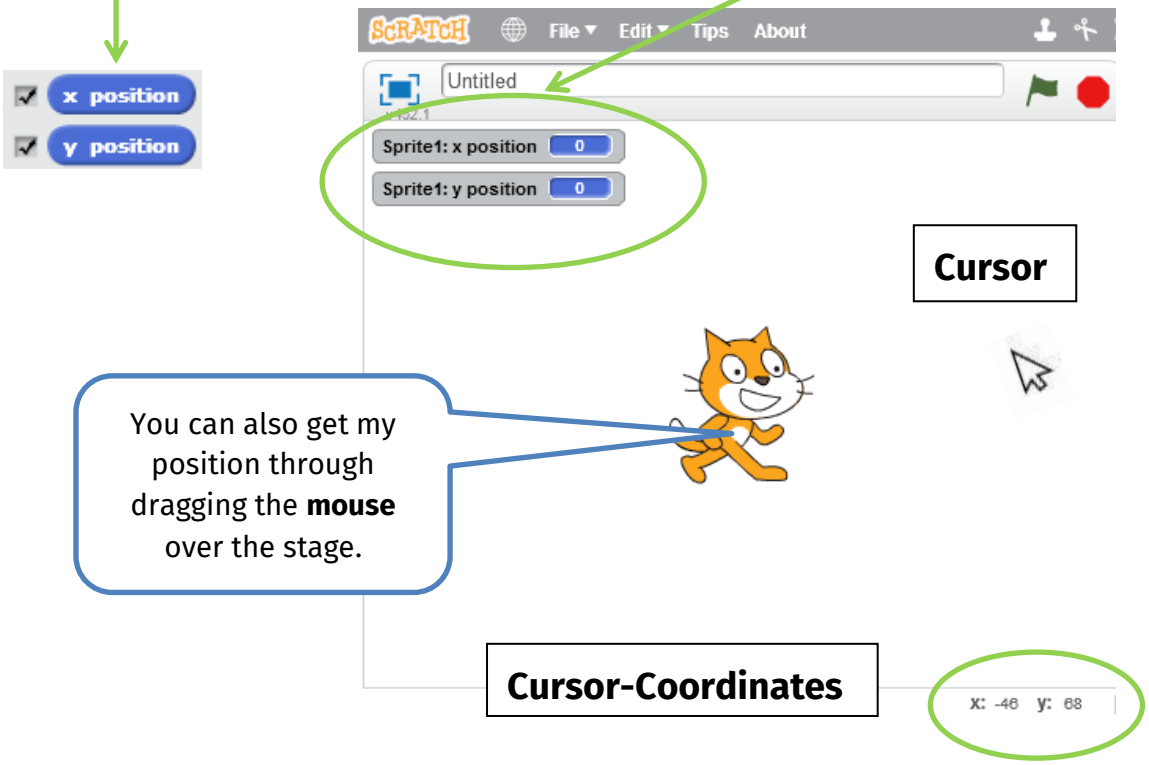
- Find the appropriate starting condition from the mask **Events**
- then the two jigsaw pieces of **Motion** that fit our goal

Motion and Drawing

The cat can also run to a certain spot on the stage you choose. It is oriented towards a **coordinate plan**.



You can display the coordinate positions from the **Motion**, to find out where your cat is on the stage right now. Just click on it.



Motion and Drawing

1. Let run to the **right** or to the **left.**

2. At what **coordinates** does it stop? Note:

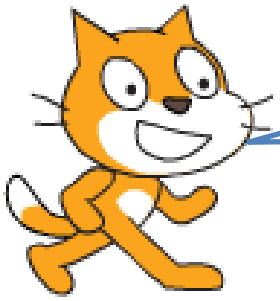
x:

y:

3. Select a random **x-** and any **y-coordinate** on the stage and let your cat go there. Use different commands from the **Motion** mask.

4. What does the term "**coordinate**" mean? Explain:

Please, keep in mind:
Everything you drag into the loop is performed permanently - without any break constantly repeated.
The loop can be used during its execution, by clicking on the point above your stage.



Well done! Let's take a short break and go to the final spurt!

Let the cat go up and down:

If touches the edge of the stage, it should reappear on the opposite side and continue. So do not collide with the edge anymore, or run over the edge.

The appropriate **statement** for this purpose can be found in the **Control:**

If the cat has arrived at the right edge,
then it should reappear at the left edge again continuing running.

„if-then“- **command**

Since we want to prevent the cat from running over the edge, we now **compare** its **current position** (

with the position from the right edge of the stage.

Do you still remember how the **x-coordinate's** count?

For this **comparison** the" *larger-than-operator*" forms the basic framework.

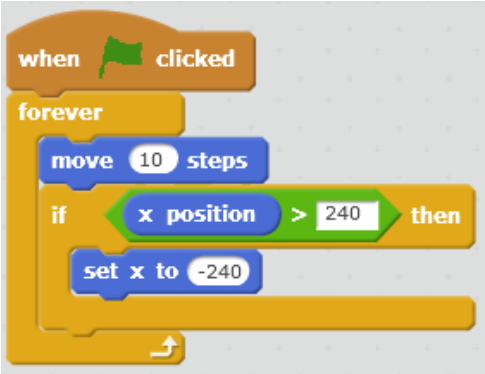
You will find this function under **Operators**. You are now supposed to draw the **x-position** from the **Motion** and write the **x-coordinate** of the right edge behind the”> -mark”.

From the **Motion** mask you need the following assignment: 

The **x-coordinate** lets the cat reappear at the left edge (**x: -240**).

- Let the now run towards the right edge of your stage and let it reappear at the left edge.
- In **addition** to the commands you are already familiar with, use commands from the areas **Control**, **Operators** and **Motion** mentioned above.

If your program looks like this, then you did it!



Now it's going to be creative:

In order to be able to track which way the  runs, you can have the trail tracked.

For this purpose, simply select "*switch-on*" from the **Pen** function and drag it into your program interface. You can determine the *size* of the pen and the pen *colour* yourself.

It is more complicated to have the cat paint a flower...

What ideas do you have? Try it out! 

Congratulations!
You have reached your goal!