



SCRATCH BRASIL

Tutorial de Lógica Básica no Scratch

Material reorganizado e adaptado por:

Suélien Rodolfo Martinelli

Disponibilizado em:

www.scratchbrasil.net.br

Material original criado por:

Vitor de Carvalho Melo Lopes e Jucimar de Souza Lima Junior

Disponibilizado em:

www.originaldesigner.com.br



Criar - Jogar - Discutir - Compartilhar





Lógica Básica em Scratch

Neste tutorial vamos ver alguns recursos de lógica, importantes para serem usados na criação de jogos e animações no Scratch.

Mas o que é lógica? Lógica é a técnica de encadear pensamentos para atingir determinado objetivo, de forma rápida e eficiente.

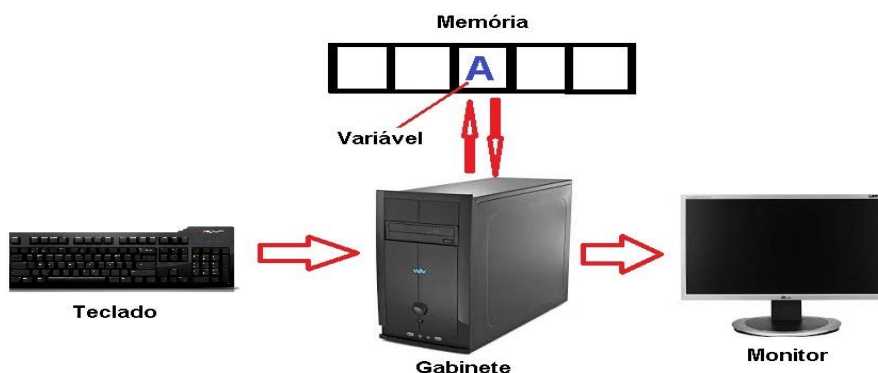
A partir disso, para cada recurso de lógica aprendido, serão feitos vários exemplos de projetos no Scratch, para fixar melhor este conteúdo.

Além disso, será visto sobre todas as categoriais e qual a função de cada uma delas.

Variáveis

Já foi visto que temos uma categoria no Scratch chamada Variáveis. Mas você sabe o que significa variável? Variável é um local reservado na memória do computador, que pode armazenar qualquer valor do conjunto ou conjunto de valores.

Para entender melhor isso, veja a imagem abaixo. Ela mostra que ao você digitar no teclado a letra “A”, a CPU manda isso para a memória. Na memória, é onde fica esse espaço reservado, que armazena um valor. Sendo que em seguida, esse valor é mostrado no monitor.



O exemplo é só para ilustrar o que significa variável, e como ela vai funcionar no Scratch. Vale lembrar que embora uma variável possa assumir diferentes valores, ela só pode armazenar um valor a cada instante.

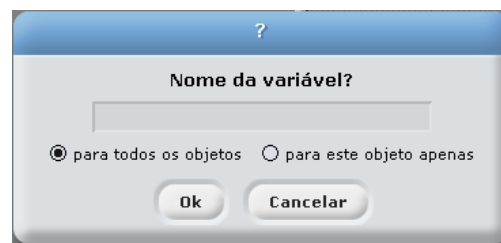
No Scratch, para se criar variáveis, acesse a categoria Variáveis, e clique no botão “criar uma variável” ou “criar uma lista”. Veja na imagem ao lado.





Depois de clicado em um dos botões, vai ser aberto uma tela que vai pedir para você dar um nome a sua variável, como mostrado ao lado.

Além disso, você deverá selecionar se ela vai ser uma variável apenas de um único componente do jogo, ou responsável por todos os objetos do jogo.



Operadores Relacionais

Assim como na matemática, haverá a necessidade no Scratch de comparar valores. Operadores relacionais servem para comparar qualquer tipo de valor. No Scratch, os operadores relacionais são apenas 3, sendo acessados na categoria Operadores:



Esses operadores mostrados são na ordem, os de menor (<), igual (=) e maior (>). É possível fazer com eles comparações com variáveis ou números, como mostrados abaixo:



Operadores Lógicos

Além dos operadores relacionais, existem os operadores lógicos, no Scratch. Os operadores lógicos também servem para comparar qualquer tipo de valor. No Scratch, os operadores lógicos também são 3, sendo acessados através da categoria Operadores:



Para entendermos melhor como funciona cada um deles, vejamos esses exemplos:

Quando eu digo “Tenho um computador e um celular”, ao afirmar isso, a palavra e indica que eu possuo os dois objetos, no caso o computador e o celular.

Quando eu digo “Tenho um computador ou um celular”, ao afirmar isso, a palavra ou indica que eu possuo apenas um dos objetos, ou o computador, ou o celular.

E quando eu digo “Não tenho um celular”, ao afirmar isso, a palavra não comprova que eu não possuo um celular.





Assim como no caso dos operadores relacionais, os operadores lógicos mostrados são o (e), (ou) e (não). É possível fazer com eles, comparações com variáveis ou números, como mostrados abaixo:



Uso da Condição "SE"

Para se criar vários jogos e animações, é importante usar condições. Para entendermos o uso da condição Se, faremos uma bola bater em um objeto e quando ela bater, voltar. Para isso, vamos seguir o exemplo abaixo:

O objetivo será fazer a bola cair, bater na cabeça do gato e voltar para cima. Quando ela bater na borda superior, voltar e bater novamente no gato.



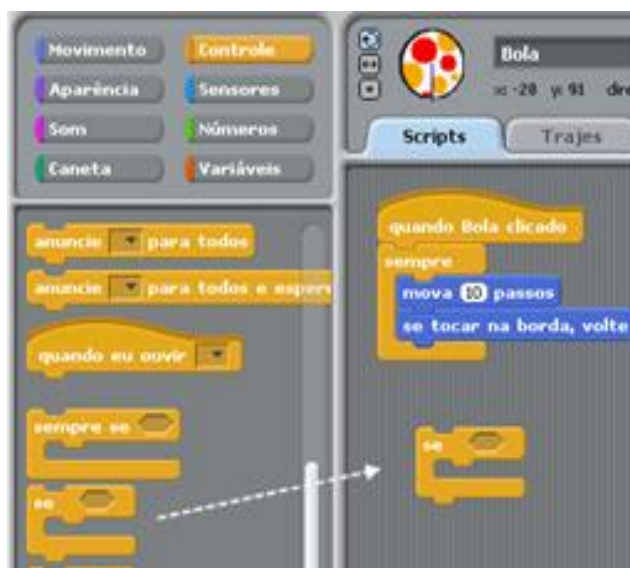
Inicialmente, mude o nome do *sprite* para "bola", e depois sua direção para 180 (para ela ir para baixo).



Os comandos da bola são iniciados quando for clicado sobre a bola. Ela sempre irá se mover, e se tocar na borda (qualquer borda do palco) ela vai para a direção contrária.

Puxe o bloco se na categoria controle e coloque na área de edição de scripts.

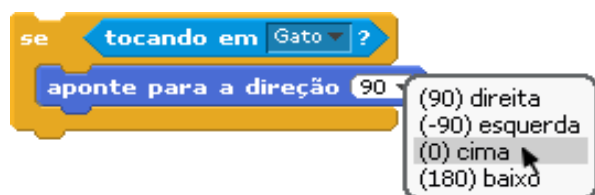
Dentro do se coloque o sensor tocando em que fica na categoria Sensores. Escolha no menu o nome do objeto que será tocado (no caso, o gato).





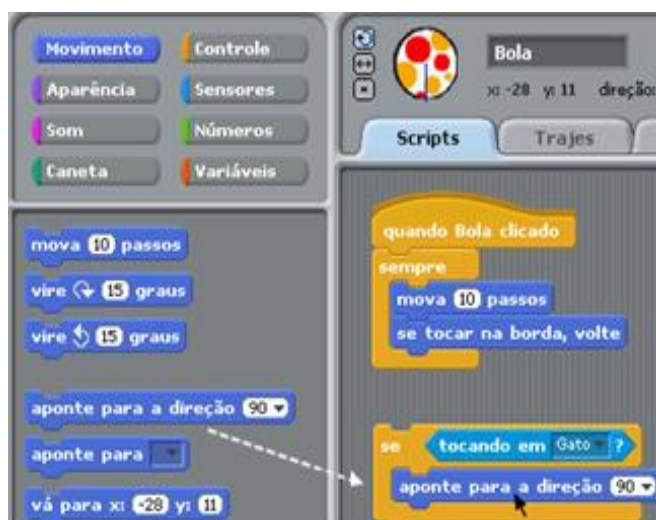
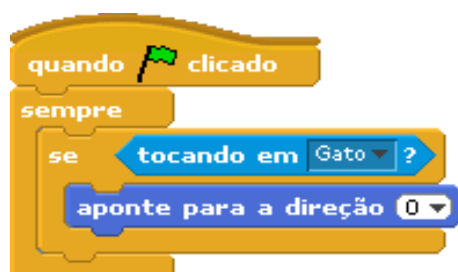
Dentro do bloco se, coloque o que acontece quando a bola tocar o gato, ou seja, acrescente o bloco aponte para a direção. Isso significa que quando a bola tocar no gato, ela irá mudar sua direção para aquela determinada no bloco de comando.



Escolha a direção (0) cima, pois após bater no gato a bola deve subir.



Acrescente o bloco sempre, para que o teste seja feito o tempo todo, e coloque a condição "Se" dentro do sempre.

Para finalizar, acrescente o controle para o início da execução do script. Neste caso foi usado o quando bandeira verde clicado.



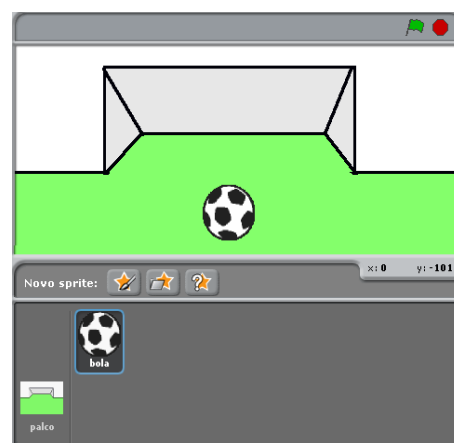
Para testar esse projeto, clique sobre a , e caso queira pará-lo, clique sobre o .



Uso da Condição "SE/SENÃO"

Agora vamos usar o teste completo: "Se/Senão". Para entendermos essa condição, vamos fazer uma bola ir para a direção do gol, e se bater nele, dizer "Gol!". Caso contrário, a bola apenas continuará movendo-se na tela.

Inicialmente faça o desenho do gol na área de palco, e adicione o sprite de uma bola para ficar semelhante ao mostrado.

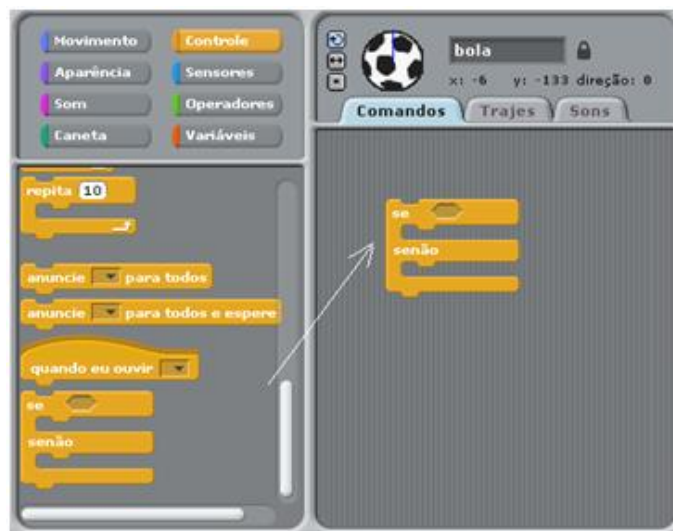




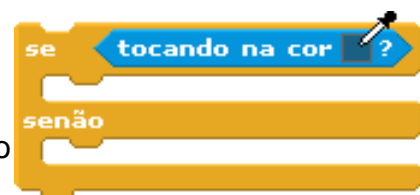
Primeiro mude a direção da bola para 0, para ela ir na direção do gol. Para isso, arraste para cima a linha em azul no objeto bola, como indica a imagem ao lado, com os itens circulados em amarelo:



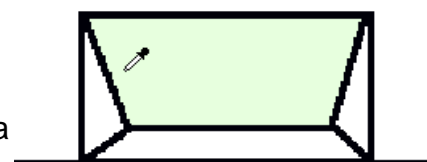
Acrescente na área de edição o bloco Se / Senão que fica na categoria Controle, como mostrado ao lado. Depois coloque no espaço do se o sensor tocando na cor, como indicado abaixo.



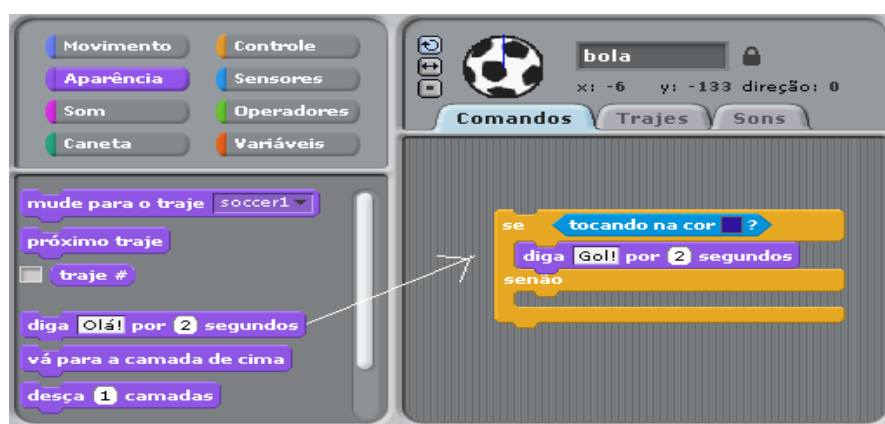
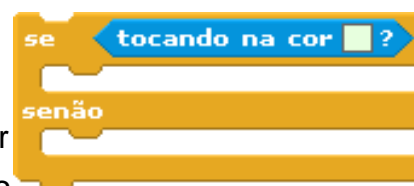
Em seguida, neste último bloco que você arrastou, clique no quadrado colorido. O ponteiro do mouse vai se transformar em um conta gotas, onde você deverá clicar em algum lugar para selecionar uma cor.



Com isso, você deverá clicar no fundo do gol, que possui uma cor cinza claro, deixando o quadradinho deste bloco também na cor cinza claro, como representado nos processos ao lado.



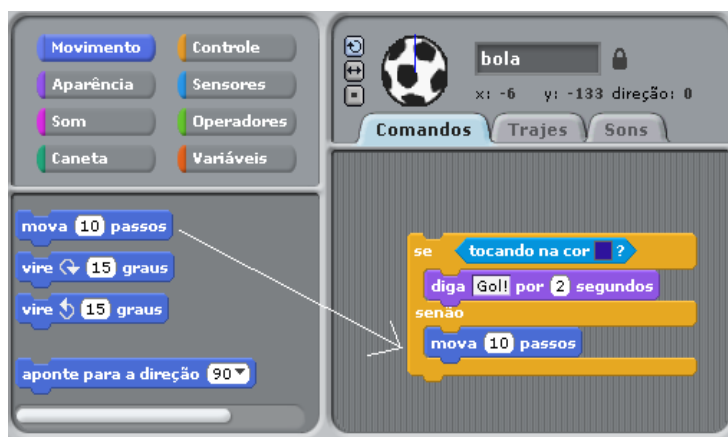
Ainda dentro do bloco se, coloque a ação que deve ocorrer quando a bola tocar o gol, ou seja, arraste o bloco diga Go! por 2 segundos, como mostrado abaixo:



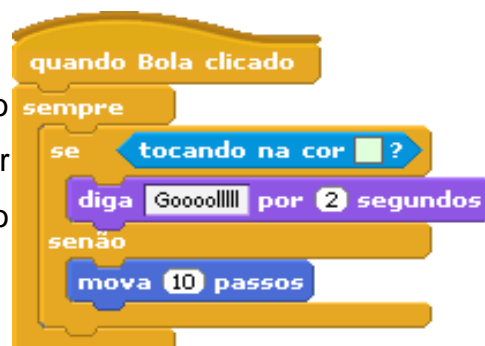


Em seguida, você deve colocar dentro do senão o comando mova, para que a bola continue andando caso ela não toque no gol, como mostrado ao lado.

Coloque os comandos Se/Senão dentro do bloco sempre, para que este teste seja feito o tempo todo.



Por fim, coloque no começo destes comandos, o bloco quando bola clicada. Isso fará com que a bola ande quando for clicada com o mouse. Seus comandos ficarão semelhantes ao mostrado ao lado.

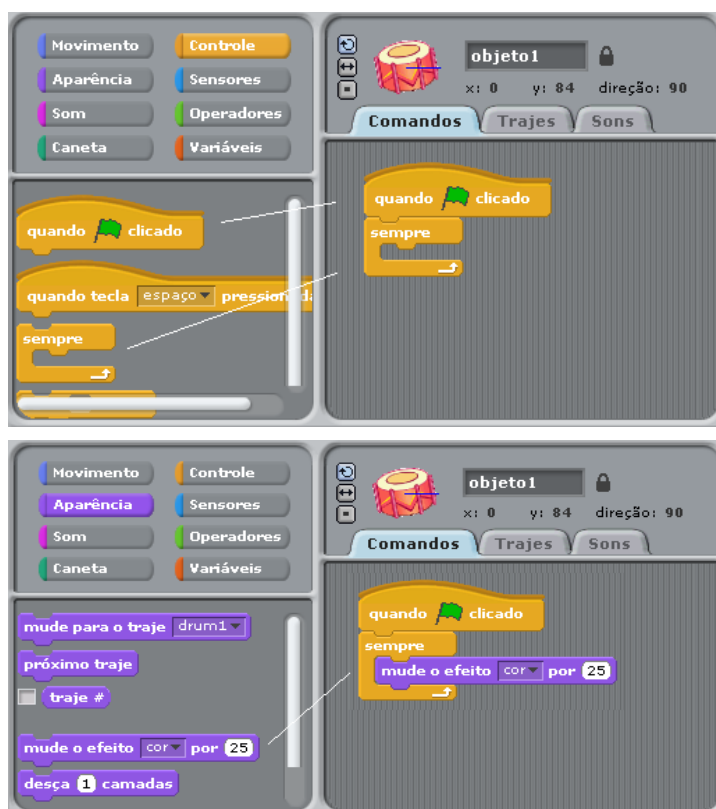


Uso da Repetição "SEMPRE"

Como já foi visto antes, é possível criar no Scratch uma ou mais ações que funcionem de maneira contínua, se repetindo por tempo indeterminado. Para isso se usa o comando sempre, da categoria Controle. Qualquer outro bloco que for encaixado dentro do comando sempre será executado até o programa ser parado. Para isso, vamos fazer o exemplo abaixo:

O objetivo será fazer um tambor tocar um som, com o intervalo de tempo de 1 segundo de diferença, além do tambor trocar constantemente de cor, assim como todo esse processo deve ocorrer infinitamente.

Inicialmente, vamos arrastar o bloco quando bandeira verde clicado e o sempre, unindo-os, ambos da categoria Controle, como mostrado na imagem abaixo.

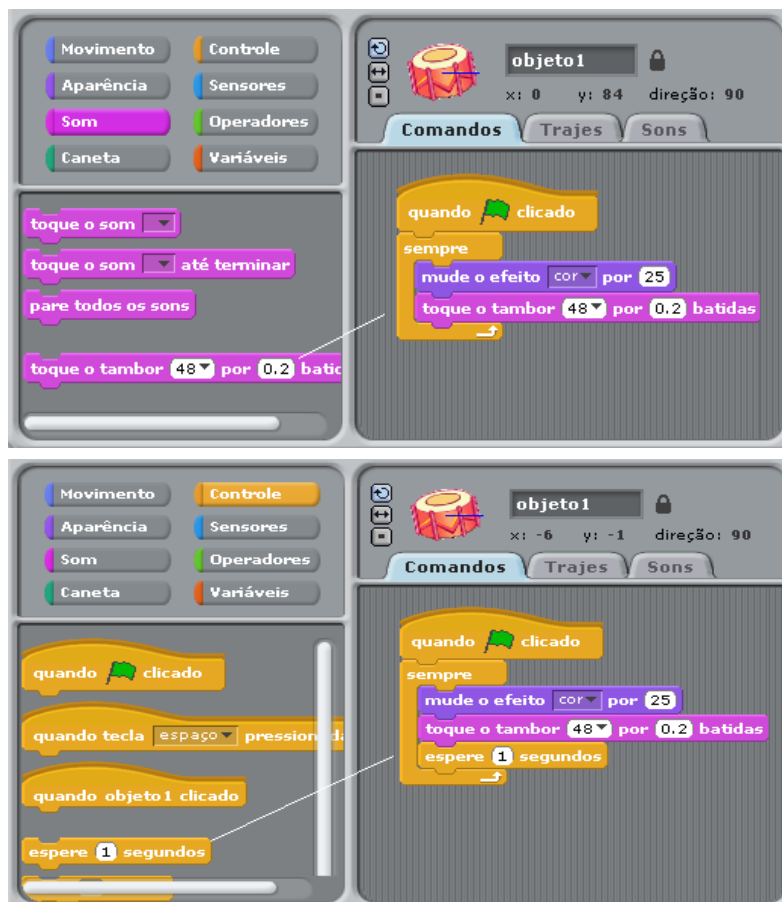






Em seguida, arraste para dentro do bloco sempre, o comando mude o efeito cor por 25, que esta na categoria Aparência, como mostra a imagem ao lado.

Depois, arraste para dentro do bloco sempre, o comando toque o tambor 48 por 0.2 batidas, que esta na categoria Som. Coloque-o abaixo do último bloco encaixado, como exibe a imagem ao lado.

Por fim, arraste para dentro do bloco sempre, o comando espere 1 segundo, que esta na categoria Controle. Coloque-o abaixo do último bloco encaixado, como exibe a figura ao lado.



Para testar esse projeto, clique sobre a , e ela vai executar os blocos de comando, sem parar, sempre mudando a cor do tambor e tocando um som. Caso queira parar o projeto, clique sobre o .



Uso da Repetição "SEMPRE SE"

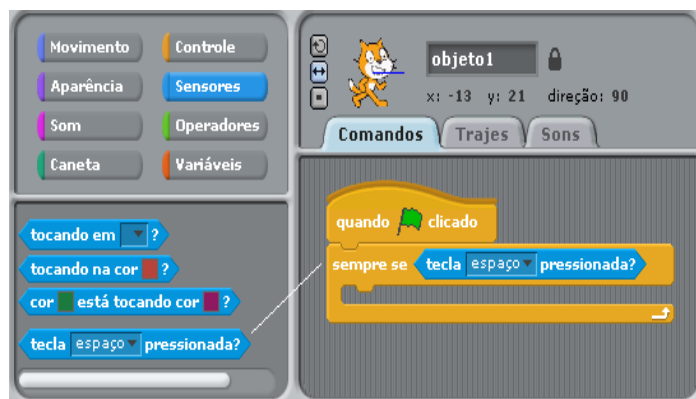
Depois de visto as estruturas sempre e se, existe também um bloco no Scratch que é a junção de ambos. O comando sempre se é a junção destes dois blocos, e tem por função verificar constantemente, se uma condição é verdadeira ou não.

Para isso, vamos criar um pequeno conjunto de blocos com o seguinte objetivo: Após habilitar a bandeira verde, faça com que seja verificado se a tecla "espaço" foi pressionada, e sempre que ela for apertada, fará o gato mover 10 passos. Além disso, caso ele encoste na borda, deverá voltar.





De início, arraste os blocos quando bandeira verde clicado e sempre se, ambos da categoria Controle, e encaixe-os como mostrado ao lado.



Depois, arraste o bloco tecla espaço pressionada?, da categoria Sensores, para frente da repetição do sempre se, como mostra a imagem ao lado.



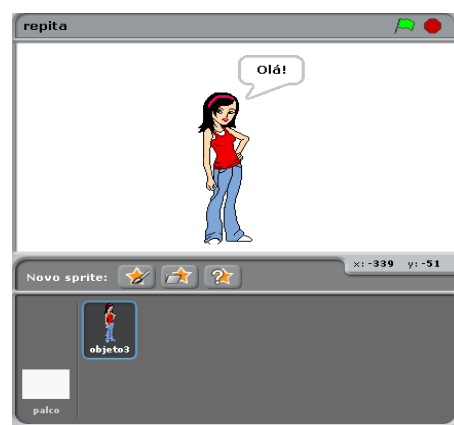
Em seguida, encaixe os blocos mova 10 passos e se tocar na borda, volte, que estão na categoria Movimento, e coloque-os dentro do sempre se, como indica a figura ao lado.

Para testar esse projeto, clique sobre a  e depois pressione a tecla “espaço” do teclado. Perceba que os comandos sempre serão executados, se esta tecla for apertada.

Uso da Repetição "REPITA"

Às vezes há a necessidade de no Scratch você fazer com que um processo dure um certo número de vezes. A estrutura de repetição repita faz com que as ações, dentro deste bloco, sejam realizadas uma determinada quantidade de vezes.

Para entendermos esse comando, vamos criar um projeto simples, onde nosso objetivo será fazer uma menina falar “Olá!” por 10 vezes, por 1 segundo para cada fala. Porém, depois desse processo, a menina deverá desaparecer da tela.



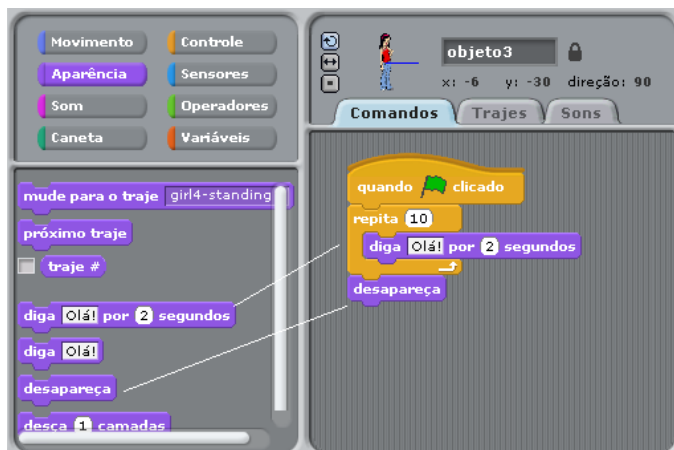
De início, vamos arrastar os blocos quando bandeira verde clicado e repita, ambos da categoria Controle, e vamos encaixá-los como mostra a imagem ao lado.



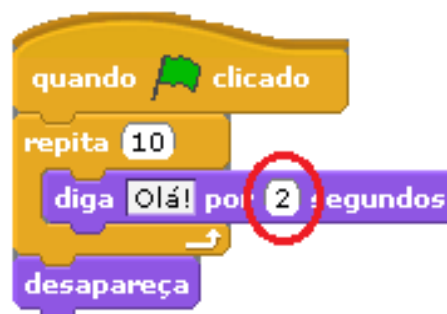


Em seguida, arraste os blocos diga Olá! por 2 segundos e desapareça, ambos da categoria Aparência.

O comando que mostrará a fala da menina deverá estar dentro do bloco repita, e o desapareça depois do comando de repetição, como indica a figura ao lado.



Por fim, no bloco diga Olá! por 2 segundos, altere o valor 2, para 1. Para isso, no local circulado na imagem ao lado, clique sobre o número 2. Ele ficará selecionado, e com isso você pode digitar o número 1.



Esse mesmo processo ocorre se você quiser alterar o valor do comando repita, ou mesmo o "Olá!" por outra palavra ou frase.

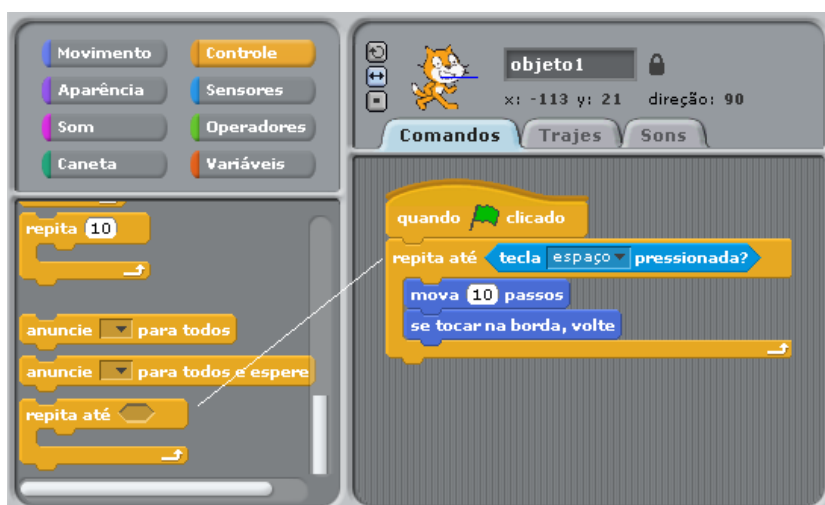
Para testar esse projeto, clique sobre a , e caso queira pará-lo, clique sobre o .




Uso da Repetição "REPITA ATÉ"

Como visto na estrutura repita, às vezes há a necessidade de no Scratch você fazer com que um processo dure determinado número de vezes. Mas também pode ser necessário que algo seja realizado continuamente, até uma ação acontecer. Para isso, usa-se a estrutura de repetição repita até, que faz com que as ações dentro deste bloco, sejam interrompidas quando alguma ação for feita.

Para melhor entendimento, vamos usar o mesmo exemplo apresentado na explicação do bloco sempre se. Mas agora, vamos eliminar o comando sempre se, e colocar em seu lugar o bloco repita até. Veja na imagem ao lado, o exemplo já pronto, com a troca do bloco, pelo repita até.





Para testar esse projeto, clique sobre a  e depois pressione a tecla “espaço” do teclado. Observe que desta vez, com o repita até, o programa já começa com o gato se movimentando de um lado para outro, e só para quando a tecla “espaço” for pressionada. Porém, anteriormente, com o sempre se, o programa fazia o gato se movimentar sempre que a tecla “espaço” fosse pressionada.



Uso de Mensagens

Às vezes, ao criarmos nossos jogos no Scratch, haverá a necessidade que sair dos blocos de comando de um *sprite*, e acessar comandos de outros personagens em nosso jogo. Para fazer esse tipo de ação, usamos o recurso de “Mensagens” no Scratch.

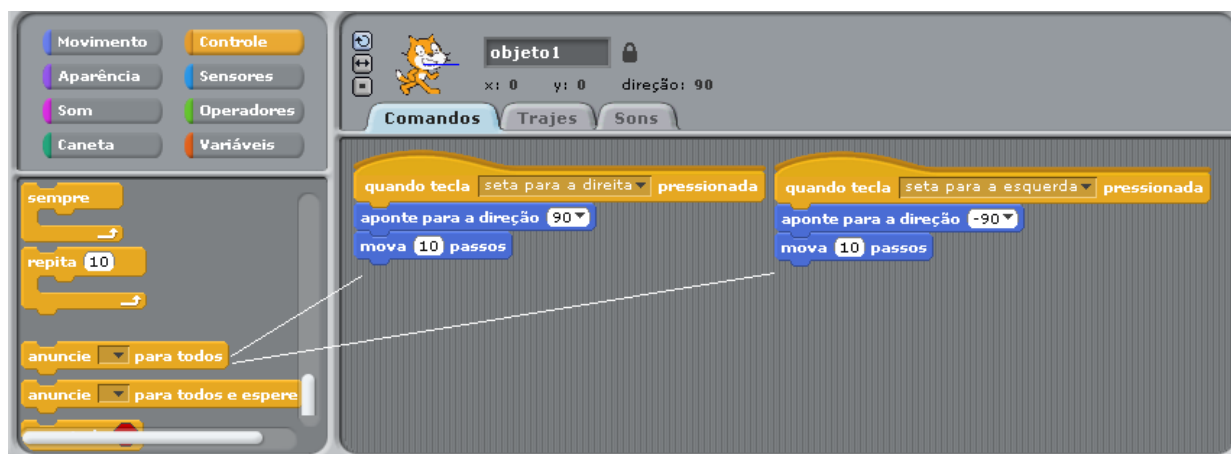
Para entendermos como funciona, vamos fazer o seguinte exemplo: Ao apertar a seta para a direita, nosso gato vai andar para essa direção, e será exibida na tela “Você está andando para a direita”. Caso seja pressionada a tecla para a esquerda, o gato anda para essa direção, e aciona a frase “Você está andando para a esquerda”.



No nosso gato Scratch, monte esses dois conjuntos de blocos, como configurados abaixo, que vai fazer nosso gato andar para a direita, quando a seta direita for pressionada, e andar para a esquerda, quando a seta esquerda for pressionada.

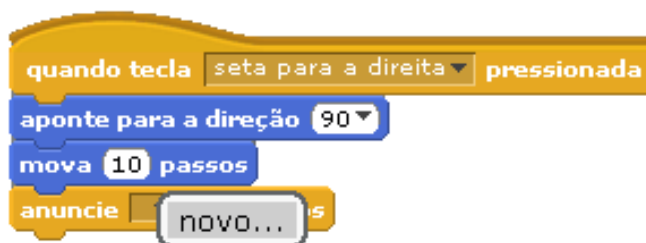


Depois de feitos esses conjuntos de blocos no gato, arraste o bloco anuncie para todos, que esta na categoria Controle, embaixo de cada conjunto de blocos, como mostra abaixo:

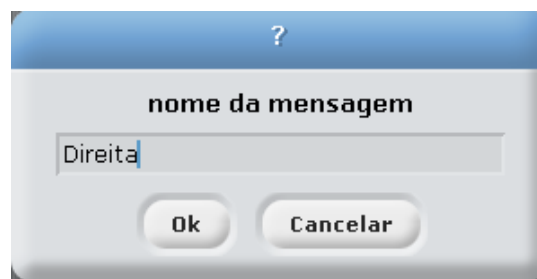




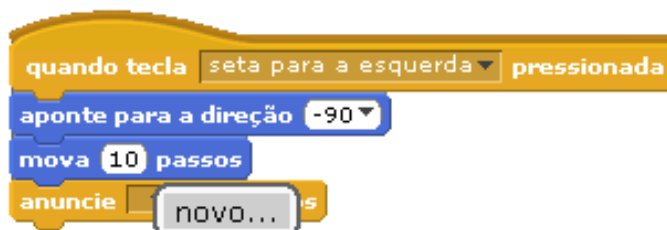
Após encaixado este bloco, no primeiro anuncie para todos, clique sobre a setinha que existe neste bloco, e sem seguida, na opção “novo”, como mostra ao lado.



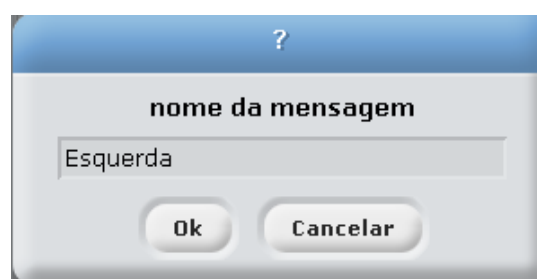
Depois de clicado em novo, será mostrada uma tela, em que é pedido para você dar um nome a esta mensagem. Para esta primeira mensagem, vamos chamá-la de “Direita”. Após digitado “Direita”, basta dar ok.



Agora, no segundo bloco de comandos do gato, também encaixamos um anuncie para todos. Clique também sobre a setinha que existe neste bloco, e sem seguida, na opção “novo”, como mostra ao lado.



Depois de clicado em novo, será mostrada uma tela, em que é pedido para você dar um nome a esta mensagem. Para esta segunda mensagem, vamos chamá-la de “Esquerda”. Após digitado “Esquerda”, basta dar ok.



Depois, crie um *sprite* que contenha apenas texto. Sendo o traje 1 deste *sprite* com a frase “Você está andando para a direita”, e o traje 2 do mesmo objeto, a frase “Você está andando para a esquerda”.



Vamos montar agora os blocos de comando do objeto “frase”. O bloco anuncie para todos, que colocamos no gato, é o que vai acionar nosso objeto “frase”.

Primeiro, vamos arrastar dois blocos do comando quando eu ouvir, que estão na categoria Controle. Em seguida, arraste dois blocos do comando mude para o traje, que estão na categoria Aparência. Nossos blocos deverão ficar como mostrados abaixo:





Para configurar os blocos abaixo, clique na setinha de cada comando, deixando a mensagem “Direita” junto ao “traje1”, e a mensagem “Esquerda”, junto ao “traje2”. Agora, basta testar pressionando as setas direita ou esquerda para ver o gatinho andar, e a frase ser mostrada na tela.



Outros comandos: categoria Controle

Outros comandos importantes desta categoria, são os espere até, pare comando e pare tudo, ainda não mostrados.



O espere até, faz com que seus blocos pausem sua execução, até ocorrer algo no jogo. Depois que isso ocorre, a execução dos comandos volta a acontecer.

O pare comando vai parar todo o conjunto de blocos em que este comando foi inserido.

E por fim, o pare tudo, vai fazer parar todos os blocos, de todos os objetos em nosso jogo. Este bloco é muito usado quando se deseja parar um jogo, depois que o jogador perde o jogo.



Outros comandos: categoria Movimento

Existem outros comandos da categoria Movimento, que são importantes de serem mostrados, envolvendo movimentos de rotação e movimentação em posições de x e y, sendo esses blocos o de vire graus à direita, vire graus à esquerda, vá para x y, vá para e deslize em segundos para x y:



Como funciona cada bloco:

vire graus à direita → esse bloco faz com que seu objeto vire para a direita o número de graus indicados;

vire graus à esquerda → esse comando faz com que seu objeto vire para a esquerda o número de graus indicados;

vá para x y → neste bloco, você deve indicar um número para x e outro para y, e depois destas posições dadas, o seu objeto vai para as indicações dadas;

vá para → neste comando, você deve indicar para onde seu objeto deve ir (como por exemplo, o ponteiro do mouse);





deslize em segundos para x y → neste bloco, você deve indicar um número para x, outro para y, e outro para o tempo em segundo depois destas posições dadas, o seu objeto desliza neste tempo mencionado, indo para as indicações dadas;

Já os comandos a seguir, ainda da categoria Movimento, dizem respeito especificamente às posições x e y de um objeto, além de sua direção, sendo eles os blocos posição x, posição y, direção, mude x por, mude x para, mude y por e mude y para:



Como funciona cada bloco:

posição x → este bloco captura a atual posição x de um objeto;

posição y → este comando captura a atual posição y de um objeto;

posição x → este bloco captura a atual direção de um objeto;

mude x por → neste comando, você deve indicar um valor para x, sendo que este número vai sempre incrementar o último valor da posição x;

mude x para → neste comando, qualquer valor que x possua, será substituído pelo número indicado no bloco para a posição x;

mude y por → neste bloco, você deve indicar um valor para y, sendo que este número vai sempre incrementar o último valor da posição y;

mude y para → neste bloco, qualquer valor que y possua, será substituído pelo número indicado no comando para a posição y;



Outros comandos: categoria Operadores

Há outros comandos importantes desta categoria, que ainda não vimos, e falaremos brevemente sobre cada um deles. De início, temos em Operadores, os 4 sinais das operações básicas na matemática, sendo eles adição, subtração, multiplicação e divisão:



Caso seja necessário realizar alguma conta em seu jogo ou animação criada, você utilizar destes operadores.

Ainda nesta mesma categoria, se caso for necessário realizar outros tipos de cálculos em seu jogo ou animação, temos os blocos resto da divisão por, arredonde, raiz quadrada de 10, e sorteie número entre 1 e 10:





Como funciona cada bloco:

resto da divisão por → ao realizar uma divisão, você pode precisar saber dela apenas o valor de resto, e esse bloco te ajuda exatamente nisso, capturando apenas o valor de resto de uma divisão;

arredonde → quando há a necessidade de se trabalhar com números inteiros (que não possuam vírgula), você pode indicar um número decimal ou variável numérica neste bloco, que ele vai te retornar apenas o valor inteiro deste número, de forma arredondada;

raiz quadrada de 10 → esse bloco serve para você realizar qualquer tipo de cálculo que sai do básico, das 4 operações. Na setinha onde está escrito “raiz quadrada”, ao clicar nela, você tem várias opções, como potência, seno, cosseno, tangente, entre outras. E onde está o número 10, você pode indicar qualquer número ou variável numérica;

sorteie número entre 1 e 10 → este bloco serve para sortear um número, a partir de uma faixa de valores. No exemplo acima, o primeiro número é 1 e o segundo 10, então o bloco vai sortear qualquer valor dentro de 1 e 10. Lembrando que os números 1 e 10 podem ser substituídos por qualquer outro número ou variável numérica.

Para finalizar esta categoria, temos alguns blocos que trabalham com textos e números, sendo eles o junte olá mundo, letra 1 de mundo e letras em mundo:



Como funciona cada bloco:

junte olá mundo → esse bloco junta duas mensagens de texto ou variáveis textuais, formando palavras separadas como uma só frase;

letra 1 de mundo → esse bloco mostra a letra, a partir do número e palavra ou variável textual indicada neste comando. Neste exemplo, ele mostra o número 1 e a palavra “mundo”, consequentemente, a letra de retorno será “m”, pois ela é a letra 1 desta palavra. Outro exemplo, seria indicar como número o valor 4 e a palavra “sorvete”, sendo a letra de retorno o “v”, que é a letra 4 desta palavra.

letras em mundo → esse comando retorna o número total de letras que possui uma palavra ou variável textual. Neste exemplo, ele retornaria o valor 5, pois a palavra “mundo” tem um total de 5 letras. Outro exemplo, baseado na palavra “sorvete”, ele retornaria o número 7, pois essa palavra tem um total de 7 letras.





Outros comandos: categoria Sensores

Há outros comandos importantes da categoria Sensores, que ainda não vimos, e falaremos brevemente sobre cada um deles. Para começar, temos em Sensores os comandos tocando em ?, tocando na cor ? e cor está tocando cor ?:



Como funciona cada bloco:

tocando em ? → esse comando vai fazer um comparativo, verificando se um *sprite* tocou em outro personagem;

tocando na cor ? → esse comando vai realizar uma comparação, verificando se um *sprite* tocou em uma cor específica (como mostrado no exemplo, a cor roxa);

cor está tocando cor ? → esse comando vai realizar uma comparação, verificando se uma cor tocou em uma outra cor (como mostrado no exemplo, se a cor lilás tocou na cor amarela).

Além desses blocos, temos dois comandos que manipulam perguntas e respostas dentro do Scratch, sendo eles o pergunte e espere e resposta:



Como funciona cada bloco:

pergunte e espere → nesse bloco, onde há a mensagem “Qual o seu nome?”, você pode digitar qualquer pergunta, que será mostrada ao usuário, assim como vai exibir uma caixa de texto, esperando com que o usuário digite uma resposta;

resposta → já este bloco, vai servir para capturar a resposta digitada, a partir da pergunta feita no bloco anterior, e realizar qualquer tipo de manipulação com esta resposta.

Existem blocos também que verificam e/ou manipulam as posições x e y de um objeto inserido no Scratch, assim como aqueles que realizam comparações de distância ou de toque em teclas. Esses blocos são mouse x, mouse y, posição x de objeto1, distância até, mouse pressionado? e tecla pressionada?:



Como funciona cada bloco:

mouse x → este bloco captura a atual posição x do mouse;





mouse y → este comando captura a atual posição y do mouse;

posição x de objeto1 → este bloco captura tanto a atual posição de x, como a de y, em um determinado objeto (*sprite*) mencionado;

distância até → neste comando, você indica uma determinada distância entre objetos (*sprites*), ou entre um objeto e o ponteiro do mouse;

mouse pressionado? → este bloco verifica se o botão do mouse foi pressionado ou não;

tecla pressionada? → este comando verifica se uma determinada tecla do teclado foi pressionada ou não;

Além destes comandos, existem mais dois desta mesma categoria, que servem especificamente para se trabalhar com tempo em um jogo, sendo eles o zere temporizador e temporizador.



Como funciona cada bloco:

zere temporizador → esse comando zera o valor que estiver no temporizador;

temporizador → esse bloco, quando acionado, mostra na tela do jogo um cronômetro, que temporiza o tempo desde o início da execução de um *game*.



Outros comandos: categoria Caneta

A categoria Caneta, que ainda foi pouco utilizada, possui blocos responsáveis por fazer nosso personagem desenhar na tela. Alguns dos blocos mais simples desta categoria são o abaixe a caneta, levante a caneta, limpe e carimbe:



Como funciona cada bloco:

abaixe a caneta → este comando abaixa a caneta, fazendo um personagem desenhar na tela;

levante a caneta → este bloco levanta a caneta, fazendo com que um personagem não desenhe na tela;

limpe → este comando limpa da tela tudo o que foi desenhado anteriormente;

carimbe → este bloco carimba na tela determinado objeto.

Além desses comandos, existem outros que são importantes para alterar efeitos de tons e cor da caneta, sendo os blocos mude o tom da caneta por, mude o tom da caneta para, mude a cor da caneta para cor, mude a cor da caneta por e mude a cor da caneta para:





Como funciona cada bloco:

mude o tom da caneta por → neste bloco, você deve indicar um valor para o tom da caneta, sendo que este número vai sempre incrementar o último valor do tom da caneta;

mude o tom da caneta para → neste comando, qualquer valor já dado que representa o tom da caneta, será substituído pelo número indicado no bloco para alterar sua tonalidade;

mude a cor da caneta para cor → este comando muda a cor atual da caneta, para outra cor;

mude a cor da caneta por → neste bloco, você deve indicar um valor para a cor da caneta, sendo que este número vai sempre incrementar o último valor da cor da caneta;

mude a cor da caneta para → neste comando, qualquer valor já dado que representa a cor da caneta, será substituído pelo número indicado no bloco para alterar sua cor.

Por fim, além de mudar tons e cores da caneta, é possível alterar seu tamanho, pelos blocos mude o tamanho da caneta por e mude o tamanho da caneta para:



Como funciona cada bloco:

mude o tamanho da caneta por → neste bloco, você deve indicar um valor para o tamanho da caneta, sendo que este número vai sempre incrementar o último valor do tamanho da caneta;

mude o tamanho da caneta para → neste comando, qualquer valor já dado que representa o tamanho da caneta, será substituído pelo número indicado no bloco para alterar seu tamanho.





SCRATCH BRASIL

Você sabe o que é ?

O Scratch ajuda crianças e jovens a aprender a pensar de maneira criativa, refletir de forma sistemática, e trabalhar de colaborativamente, além de treinar habilidades essenciais para a vida no século 21. O software Scratch é um projeto do grupo Lifelong Kindergarten no Media Lab do MIT, e é fornecido gratuitamente.

Com isso, temos o Scratch Brasil, onde fornecemos material gratuito em língua portuguesa sobre a ferramenta, além de mostrar notícias, eventos, entre outras informações, de como professores e alunos podem usar a plataforma em sala de aula para a criação de jogos e animações de tema educativo.

Saibam mais sobre nosso projeto em:



www.scratchbrasil.net.br



facebook.com/scratchbr



[@scratchbrasil](https://twitter.com/scratchbrasil)



youtube.com/scratchbr



scratchbrasil.net.br@gmail.com



Criar - Jogar - Discutir - Compartilhar

