

INSTITUTO FEDERAL

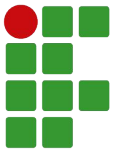
Sergipe

Campus Tobias Barreto

LabJogos

Oficina de Desenvolvimento de Jogos 2D em
RPG Maker VX Ace

Prof. Me. Christiano Lima Santos



Agenda do curso

— — —

Primeiro Dia

Introdução à Indústria de Jogos Independentes

Introdução ao RPG Maker

Criação de um projeto de jogo

Criação dos mapas

Segundo Dia

Criação do protagonista

Criação dos personagens e objetos interativos

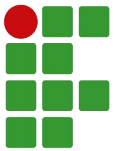
Criação dos eventos de diálogo

Terceiro Dia

Criação de outros eventos do jogo

Próximos passos

Primeiro Dia!



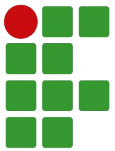
A Indústria de Jogos Eletrônicos

Mundial (INFORCHANNEL, 2021):

- ❑ Valor total da indústria de jogos eletrônicos excede US\$ 300 bilhões por ano;
- ❑ Estimam-se mais de 2,7 bilhões de jogadores;
- ❑ Jogos de tiro em primeira pessoa, RPG, arena de batalha multiplayer online e jogos no estilo Battle Royale são os mais populares.

Nacional (OLHAR DIGITAL, 2021):

- ❑ Mercado de jogos no Brasil deve atingir US\$ 2,3 bilhões em 2021;
- ❑ 47% das vendas são para dispositivos móveis, 29% para consoles e 24% para PC;
- ❑ Títulos do gênero battle royale (tipo 'Fortnite', 'PUBG' e 'Clash Royale') são os que geram maior receita.



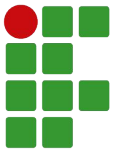
Alguns Indies de Sucesso

Stardew Valley:

- Iniciou como um jogo singleplayer para PC;
- Incorporou um modo multiplayer cooperativo;
- É vendido para PC, PS4, Xbox One, Nintendo Switch e Android;
- E em tabuleiro (físico)!
- 10 milhões de cópias vendidas - até 2020.



O quanto você acredita em sua ideia?



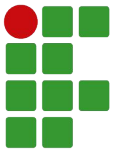
Alguns Indies de Sucesso

Forager:

- ❑ Iniciou como um protótipo singleplayer para uma game jam no itch.io ;
- ❑ Criado no Game Maker;
- ❑ Hoje, um jogo completo (multiplayer já em beta);
- ❑ Disponível para PC, PS4 e Nintendo Switch;
- ❑ Mais de 150.000 cópias vendidas (em abril e maio de 2019).

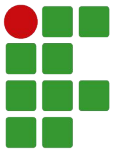


Se você acredita em sua ideia, você deve tentar e tentar!



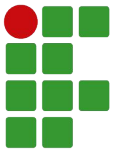
Oportunidades Para Quem Está Iniciando

- ❑ Inovação por meio do *gameplay* (chave do sucesso de muitos jogos indie);
- ❑ Possibilidade de financiamento coletivo (*crowdfunding*);
- ❑ Diversos modelos de receita (não foque nisso nos primeiros projetos);
- ❑ Aspecto multiplayer/social como pilar do *design*;
- ❑ Multiplataforma (*web, mobile e desktop*);
- ❑ Publicar seu próprio jogo x *publisher*.



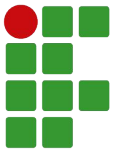
Ferramentas

<p>Quero trabalhar em um grande estúdio</p> <ul style="list-style-type: none">❑ IDE/Motor: Unity / Unreal Engine;❑ Programação: VSCode.	<p>Quero começar meu estúdio de jogos</p> <ul style="list-style-type: none">❑ IDE/Motor: Unity / GoDot / GameMaker;❑ Programação: VSCode.	<p>Quero contar uma história</p> <ul style="list-style-type: none">❑ IDE/Motor: RPG Maker / GameMaker;❑ Programação: VSCode;❑ Escrita do roteiro: Twine / WorkFlowy.
<ul style="list-style-type: none">❑ Gráficos: Gimp (2D) / Blender (3D);❑ Música: Bosca Ceoil ou Musescore (gratuitos);❑ E seu computador! <p>} Ou compre na web!</p>		



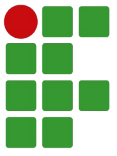
Por Que RPG Maker?

- ❑ Longa história - primeira publicação em 1997 (RPG Maker 1995);
- ❑ Bom histórico de versões (95, 2000, 2003, XP, VX, VX Ace, MV, MZ e versões para console);
- ❑ Ferramenta robusta e multiplataforma;
- ❑ Preço bastante acessível (R\$ 1,50 ~ R\$ 250,00).



O Que Podemos Fazer Com RPG Maker?

- ❑ Projetado para desenvolvimento de jogos 2D top-down de RPG baseados em turnos;
- ❑ Facilmente estendido para diversos outros tipos:
 - ❑ Jogos de action RPG (sistema de batalha ativo);
 - ❑ Jogos dirigidos à narrativa e exploração;
 - ❑ Jogos de terror.

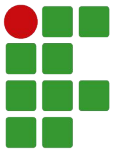


Alguns Jogos de RPG Maker

To The Moon

FreeBird Games

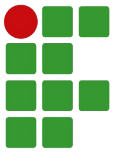




Alguns Jogos de RPG Maker

To The Moon
FreeBird Games



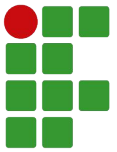


Alguns Jogos de RPG Maker

OneShot

Future Cat



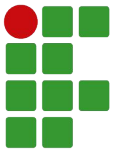


Alguns Jogos de RPG Maker

OneShot

Future Cat



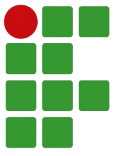


Alguns Jogos de RPG Maker

Rakuen

Laura Shighihara



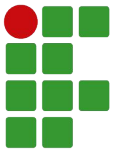


Alguns Jogos de RPG Maker

Rakuen

Laura Shighihara





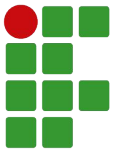
Alguns Jogos de RPG Maker

Yume Nikki

Kikiyama

(12 ou 14 anos)



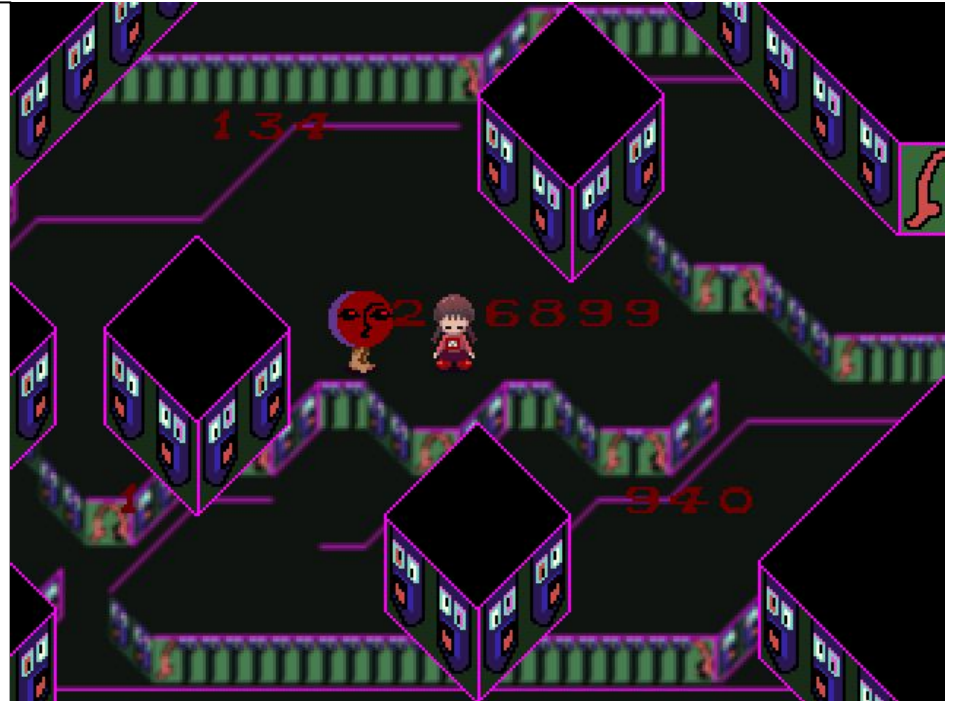


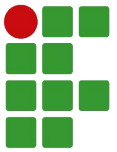
Alguns Jogos de RPG Maker

Yume Nikki

Kikiyama

(12 ou 14 anos)



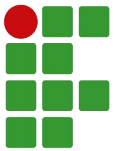


Alguns Jogos de RPG Maker

Corpse Party

Team GrisGris
(18 anos)





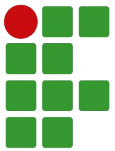
Alguns Jogos de RPG Maker

Corpse Party

Team GrisGris

(18 anos)



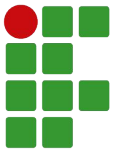


Alguns Jogos de RPG Maker

Wing of Misadventure

Allyson de Souza
Bacon
(MMORPG!)

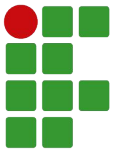




Alguns Jogos de RPG Maker

Wing of Misadventure
Allyson de Souza Bacon
(MMORPG!)



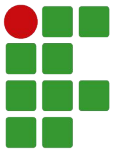


Nossos Jogos em RPG Maker

Defensores de Atlantis

<https://christianosantos.com/jogos/defensores-de-atlantis/>





Nossos Jogos em RPG Maker

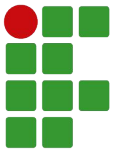
The Shadow Box

<https://theshadowbox.christianosantos.com>



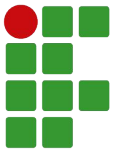
Era só um pesadelo... Aquele pesadelo...
De novo...





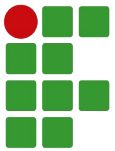
Conhecendo o RPG Maker VX Ace





Conhecendo o RPG Maker VX Ace

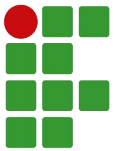
- ❑ Lançado em 2012, suporta todos os recursos presentes até hoje no RPG Maker:
 - ❑ Mapas criados a partir de *tiles* (pequenos gráficos que compõem cenários de jogos 2D);
 - ❑ Banco de dados (para criação de personagens, classes, itens, habilidades, eventos comuns etc.);
 - ❑ Personagens criados a partir de *sprites* animadas;
 - ❑ Batalhas baseadas em turnos;
 - ❑ Eventos (usados em personagens não-jogáveis, objetos no cenário, cutscenes, controle da história etc.);
 - ❑ Diálogos.



Conhecendo o RPG Maker VX Ace

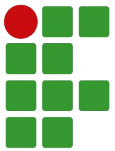
- ❑ Neste curso, estudaremos três componentes muito usados em jogo de RPG:
 - ❑ Mapas;
 - ❑ Personagens;
 - ❑ Eventos (personagens não-jogáveis, objetos no cenário, diálogos e *cutscenes*).

- ❑ E para facilitar nosso aprendizado, construiremos um jogo chamado “A Fuga”, no qual o jogador acorda em um quarto de uma casa e precisa encontrar um jeito de escapar dela.



Conhecendo o RPG Maker VX Ace

- ❑ Neste curso, usaremos a versão Lite (gratuita) do RPG Maker VX Ace.
- ❑ O RPG Maker VX Ace Lite pode ser baixado via Steam...
https://store.steampowered.com/app/224280/RPG_Maker_VX_Ace_Lite/

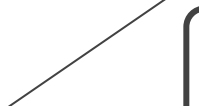


Tela do RPG Maker VX Ace

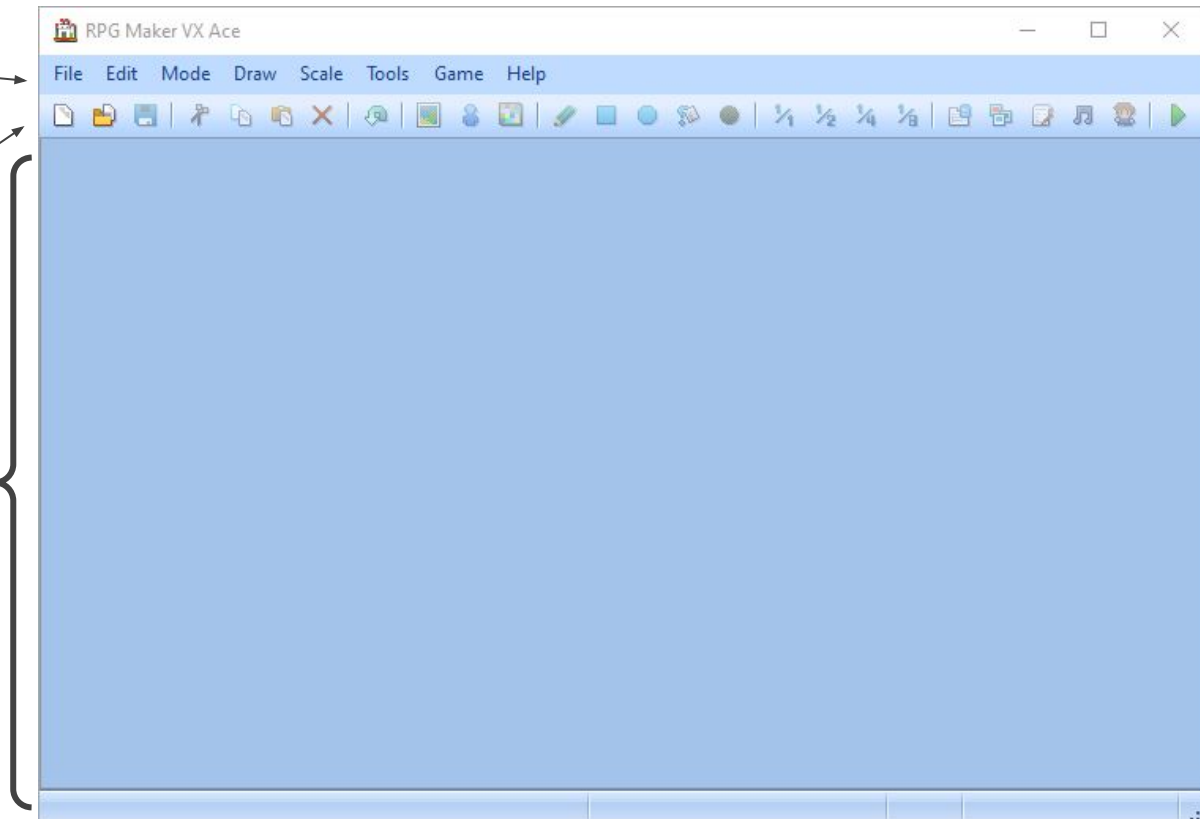
Menu

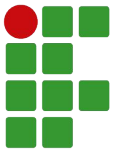


Barra de
Ferramentas



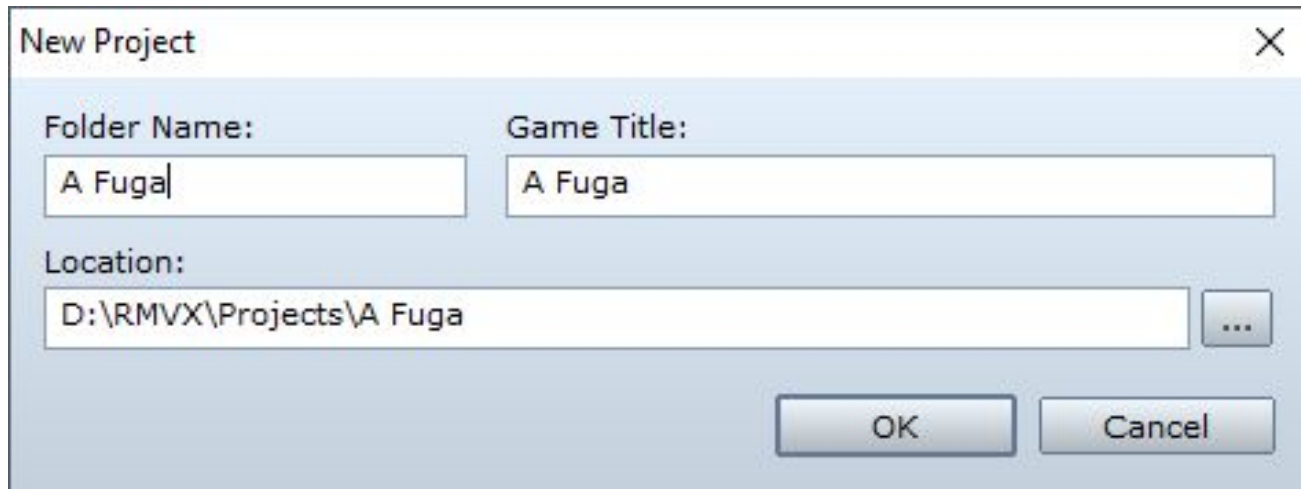
Área de
Trabalho

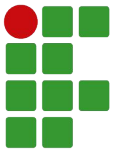




Criando Nosso Projeto

- ❑ Começamos criando o nosso projeto, por meio do menu **File** → **New Project...** ou pelo botão “New Project” na barra de ferramentas

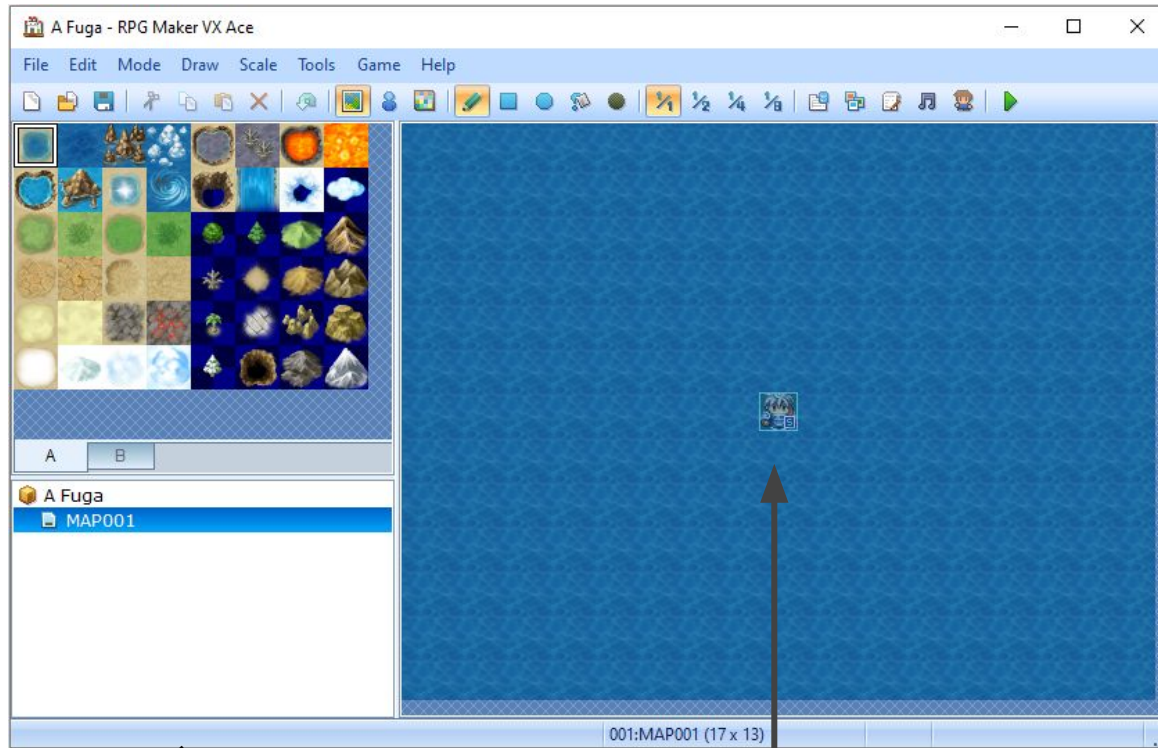




Tela do RPG Maker VX Ace (com Projeto Aberto)

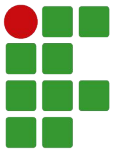
Tiles
(gráficos
para
desenhar
o mapa)

Lista de
mapas do
projeto
atual



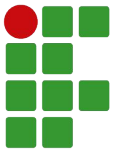
Área
mostrando o
mapa
selecionado

Ícone indicando onde o personagem do
jogador começará



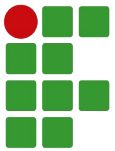
Nosso Primeiro Mapa

- ❑ Mapas compõem a estrutura do mundo e dos cenários em que um jogo ocorre;
- ❑ Neles podemos encontrar o cenário em si, objetos, personagens não-jogáveis (NPCs), marcar a posição inicial do jogador, barco ou avião (sim, é possível ter meios de transporte) bem como executar eventos com diálogos, cutscenes, batalhas etc.



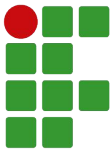
Propriedades de um Mapa

- ❑ **Name** - é o nome do mapa, como ele irá aparecer dentro de nosso projeto. Vamos criar um mapa chamado **Quarto 01**;
- ❑ **Display Name** - caso esteja preenchido, é como o nome do mapa aparecerá na tela quando o jogador entrar nesse local. Também vamos preencher com **Quarto 01**;
- ❑ **Tileset** - é o conjunto de *tiles* (gráficos) que será usado nesse mapa. Cada mapa pode usar um *tileset* diferente. Em nosso projeto, vamos utilizar o **Interior**.



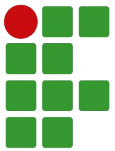
Propriedades de um Mapa

- ❑ **Width** e **Height** - são, respectivamente, a largura e a altura de nosso mapa em *tiles*. Para termos uma referência, a tela do RPG Maker VX Ace mede 17x13 *tiles*. Como um quarto não é muito grande, vamos manter esse tamanho;
- ❑ **Scroll Type** - usado caso se deseje que o mapa tenha “scrolling infinito”, isto é, quando o jogador sai pelo lado direito, ele retorna pelo lado esquerdo do mapa, por exemplo, ou ao sair pela parte de baixo do mapa, ele retorna pela parte de cima. Um quarto não precisa de tal recurso, então vamos deixar como **No Loop**.



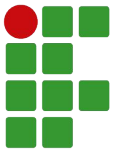
Propriedades de um Mapa

- ❑ **Specify Battleback** - usado para especificar as imagens que vão compor o fundo do cenário de batalha (*battle background*) dentro desse mapa. Como não haverá batalhas nesse mapa, vamos deixar desmarcado;
- ❑ **Encounters** - aqui podemos especificar quais batalhas aleatórias podem acontecer neste mapa. Aqui, vamos deixar esta lista vazia;
- ❑ **Steps average** - indica, em média, quantos passos o jogador andarás antes de ocorrer um encontro (batalha aleatória). Também vamos ignorar esse campo por agora.



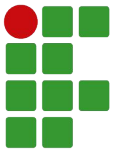
Propriedades de um Mapa

- ❑ **Auto-change BGM** - usado para escolher uma música (*background music*) a ser iniciada assim que o jogador entrar nesse mapa. Caso não selecionemos uma música, o jogo continuará executando a música selecionada atualmente. Marque essa opção e clique em “...” para escolher a música **Dungeon9** com pitch 50%;
- ❑ **Auto-change BGS** - usado para escolher um efeito sonoro de fundo (*background sound*) que será executado “em loop” enquanto o jogador estiver nessa tela. Na versão lite há somente dois sons de fundo, a chuva (Rain) e o vento (Wind). Como o quarto é um local fechado, vamos deixar desmarcada essa opção.



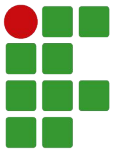
Propriedades de um Mapa

- Disable dashing** - se marcada, o jogador não pode usar a opção de correr nesse mapa. Vamos deixar desmarcada;
- Parallax background** - usado para especificar uma imagem de fundo que se moverá por trás do mapa, como um céu estrelado, planícies ou árvores de uma floresta ao longe. Por ora, também não vamos selecionar nada aqui.
- Confirme as alterações clicando em **OK** e vamos agora desenhar nosso mapa.



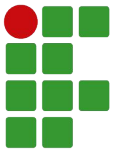
Desenhando o Primeiro Mapa

- ❑ Para desenhar o mapa, precisamos estar no modo de edição do mapa. Caso não esteja, pressione a tecla **F5**;
- ❑ A área com tiles apresenta três abas:
 - ❑ A - Gráficos geralmente usados em chão ou paredes;
 - ❑ B e C - Gráficos de objetos e coisas que podem ser colocadas no chão ou nas paredes.

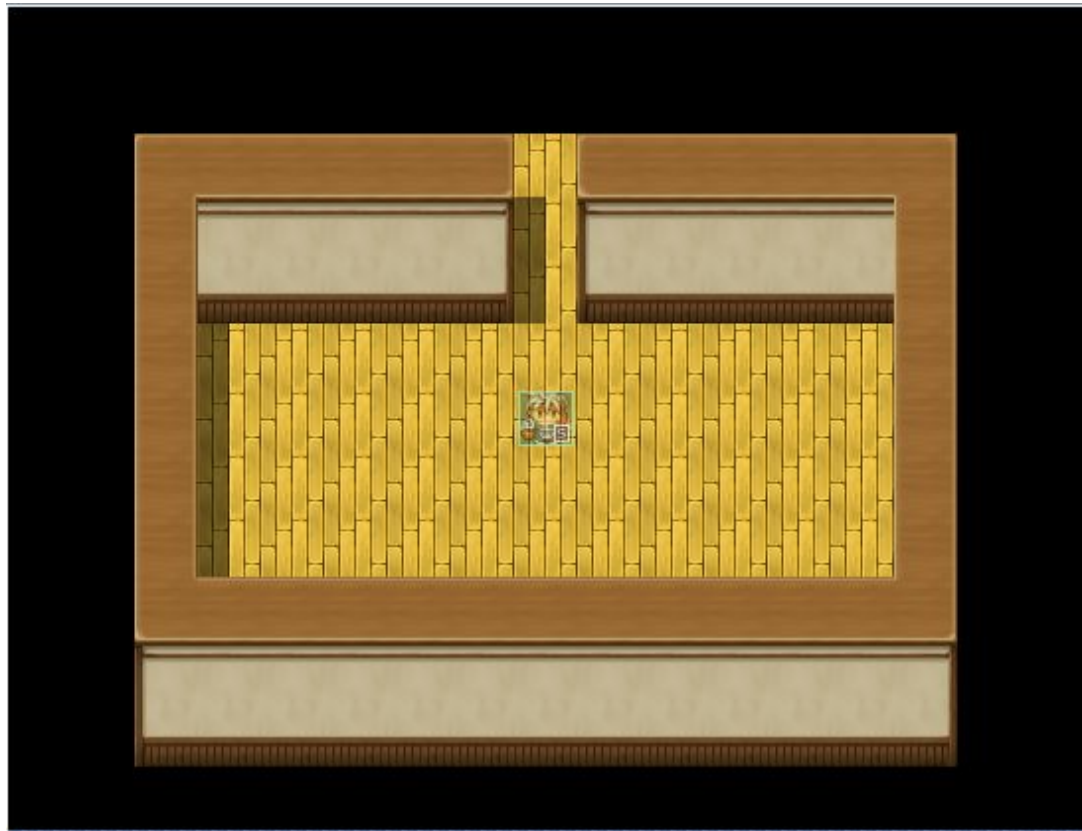
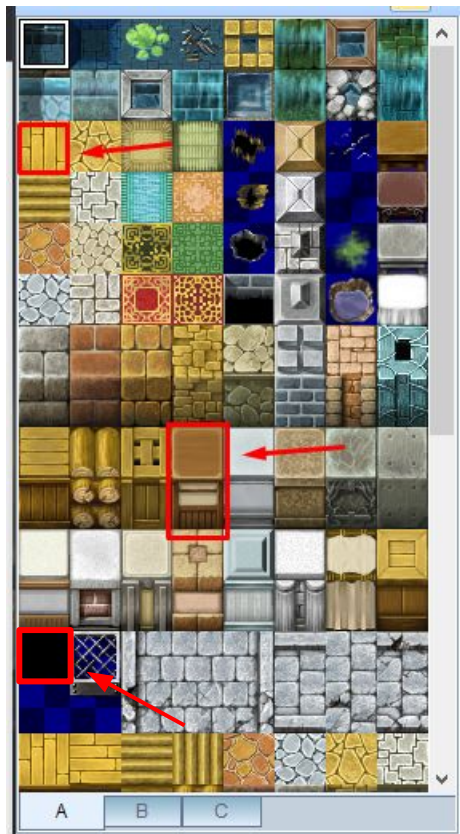


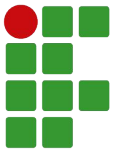
Conhecendo o RPG Maker VX Ace

- ❑ Com os gráficos disponíveis na aba A, vamos desenhar o chão e as paredes do primeiro quarto;
- ❑ Dicas:
 - ❑ Use o botão esquerdo do mouse para selecionar uma *tile* e colocá-la no cenário e o botão direito para copiar uma que já está no cenário;
 - ❑ Segure o botão do mouse e arraste tanto para selecionar mais de um *tile* (botão esquerdo) quanto para copiar mais de um (botão direito) ao mesmo tempo.



Conhecendo o RPG Maker VX Ace

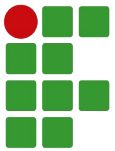




Conhecendo o RPG Maker VX Ace

- Agora, use os gráficos disponíveis na aba B e coloque uma cama, um armário e outros detalhes;
- Dicas:
 - Você pode selecionar mais de um *tile* para aplicar ao mesmo tempo;
 - Use o primeiro *tile* da paleta para apagar gráficos desta aba.

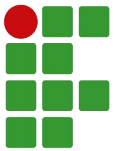




Conhecendo o RPG Maker VX Ace

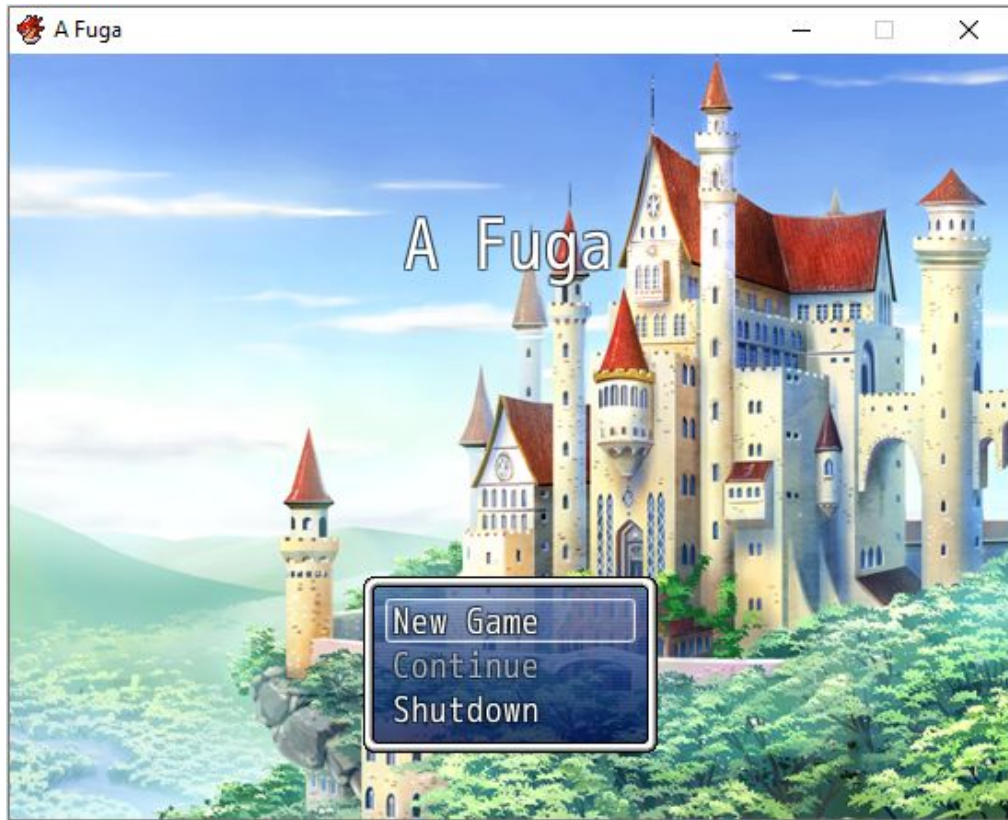
- ❑ Agora, vamos fazer o mesmo com a aba C, usando os gráficos disponíveis para detalhar mais o ambiente.

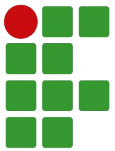




Conhecendo o RPG Maker VX Ace

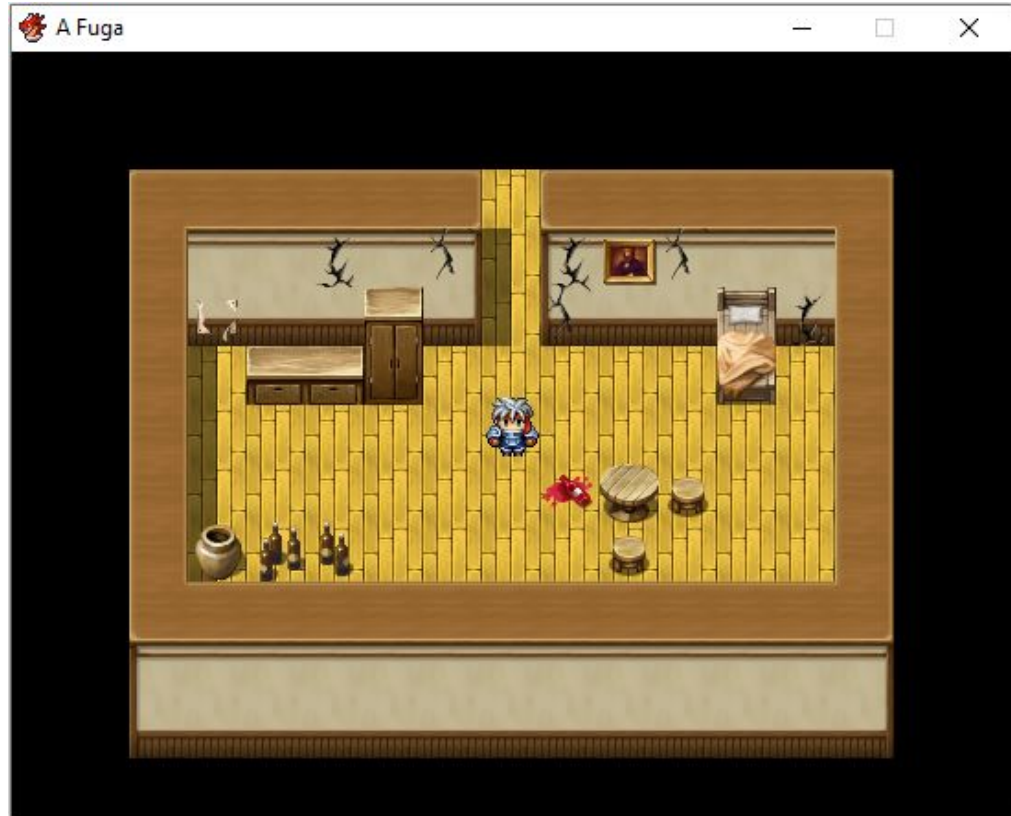
- ❑ Que tal testarmos como está o nosso jogo?
- ❑ Para isso, clique no botão verde **Playtest** na barra de ferramentas ou aperte **F12**;
- ❑ Dica: use a tecla **Z** ou **Enter** para iniciar um novo jogo.

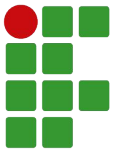




Conhecendo o RPG Maker VX Ace

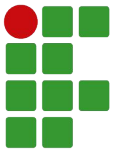
- ❑ Você pode mover o personagem com as setas direcionais;
- ❑ Após mover-se um pouco pelo mapa, pode fechar a janela do jogo.





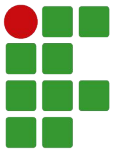
Conhecendo o RPG Maker VX Ace

- ❑ Vamos criar agora o próximo mapa: a sala da casa. Para isso, clique com o botão direito na área inferior à esquerda em que está o mapa e escolha **New Map...**;
- ❑ Name e Display Name: **Sala**;
- ❑ Tileset: **Interior**;
- ❑ Width x Height: **25** x **13**.



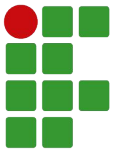
Conhecendo o RPG Maker VX Ace





Conhecendo o RPG Maker VX Ace

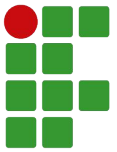
- ❑ Crie agora um evento para ligar um mapa ao outro;
- ❑ O RPG Maker oferece um mecanismo pronto para criar evento de “porta” que leva para outro mapa. Para isso:
 - ❑ Retorne ao mapa do **Quarto 01**;
 - ❑ Mude para o modo de edição de evento (aperte **F6** no teclado ou botão **Event** na barra de ferramentas);
 - ❑ Clique com o botão direito no local em que quer a porta, escolha a opção **Quick Event Creation** → **Door...**;
 - ❑ Escolha o gráfico da porta e em **Destination**, vamos escolher o mapa **Sala** no local em que queremos que o jogador apareça quando mudar de local.



Conhecendo o RPG Maker VX Ace

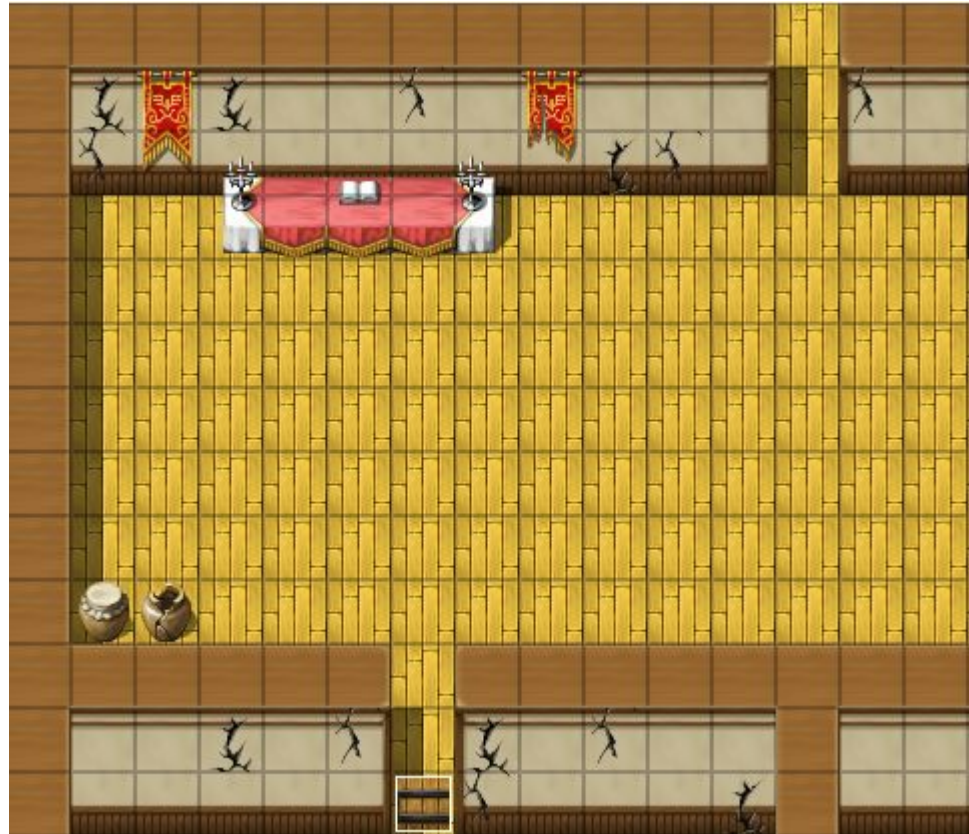
- ❑ Se feito corretamente, um evento com o desenho de uma porta deve ter aparecido em seu mapa. Algo assim:
- ❑ Teste o jogo e tente ir para a sala.

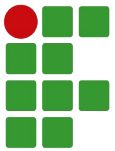




Conhecendo o RPG Maker VX Ace

- ❑ Agora, vá até o mapa **Sala** e siga o mesmo procedimento para criar uma porta que leva de volta para o **Quarto 01**.

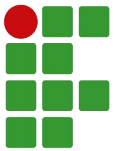




Conhecendo o RPG Maker VX Ace

❏ Dicas:

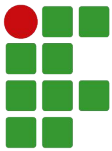
- ❏ Caso queira alterar o local em que o jogador começa dentro do jogo, vá até o mapa desejado e, no modo de edição de evento, clique com o botão direito no local inicial e escolha **Set as Starting Position → Player**;
- ❏ Você pode usar um processo similar para definir a posição inicial de um barco (**Set as Starting Position → Boat**), navio (**Set as Starting Position → Ship**) ou aeronave (**Set as Starting Position → Airship**).



Conhecendo o RPG Maker VX Ace

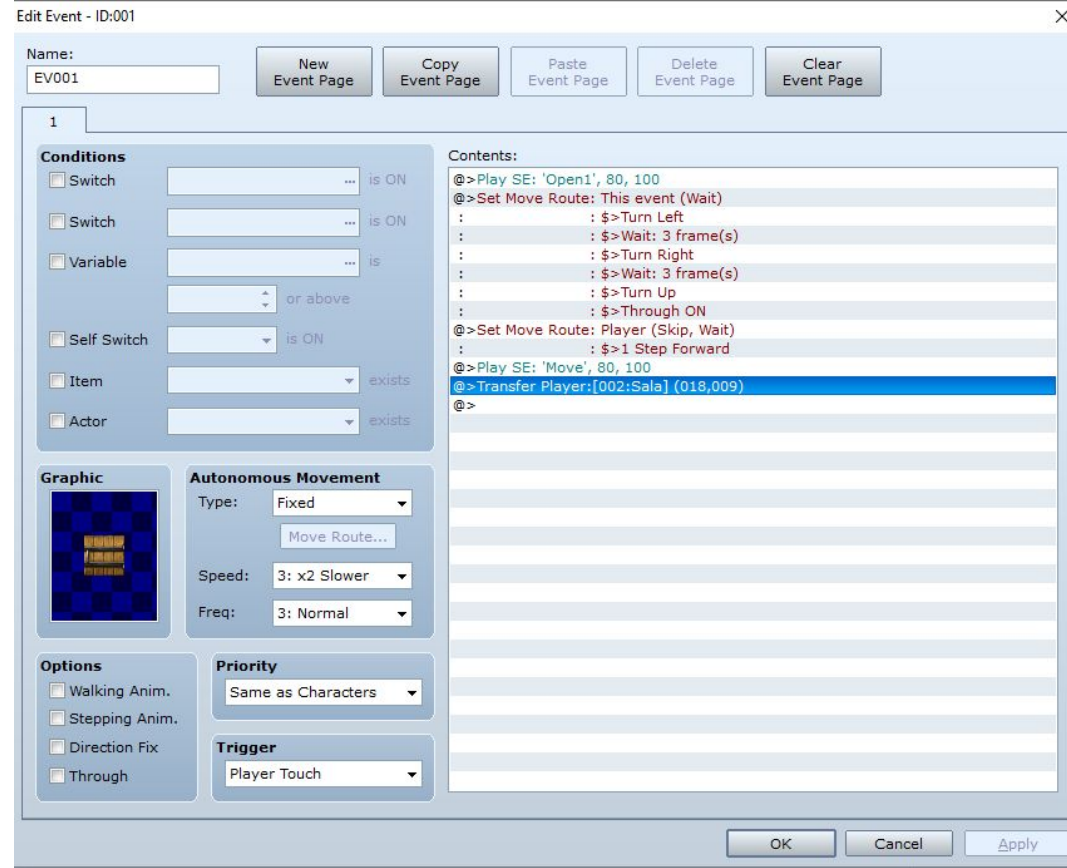
- ❑ Agora vamos criar o segundo quarto parecido com o primeiro, fazendo uma cópia deste, clicando com o botão direito e escolhendo **Copy** e depois cole com a opção **Paste**. Vamos chamá-lo de **Quarto 02**;
- ❑ Use o modo de edição de mapa (tecla **F5**) para deixá-lo um pouco diferente do primeiro.

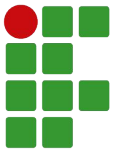




Conhecendo o RPG Maker VX Ace

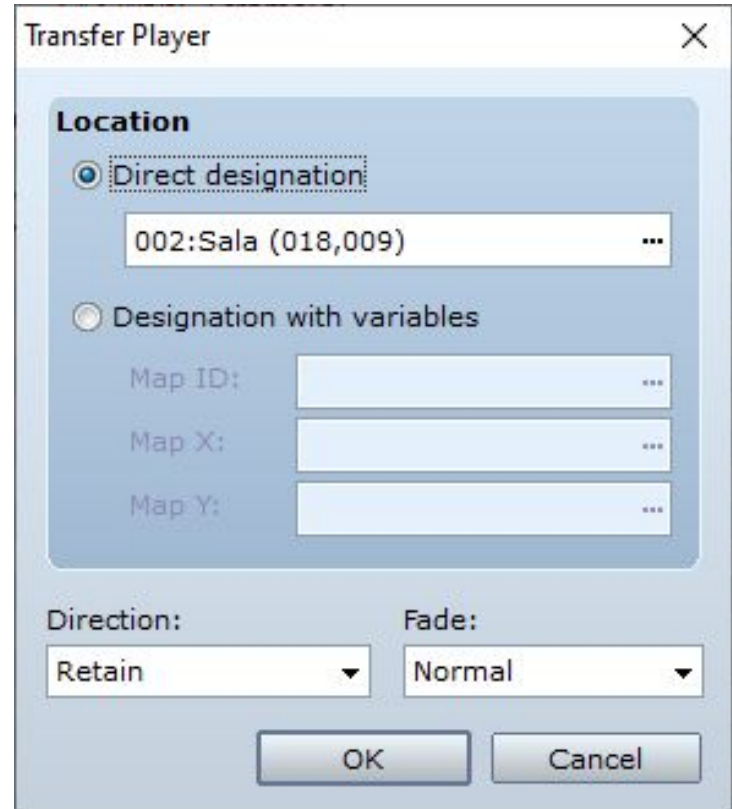
- ❑ Agora vamos “corrigir” o evento da porta, para que o jogador saia no lugar correto;
- ❑ Para isso, entre no modo de edição de evento, clique com o botão direito em cima do evento da porta e escolha **Edit Event...**;
- ❑ Clique com o botão direito na linha **Transfer Player...** e escolha **Edit...**.

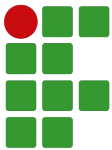




Conhecendo o RPG Maker VX Ace

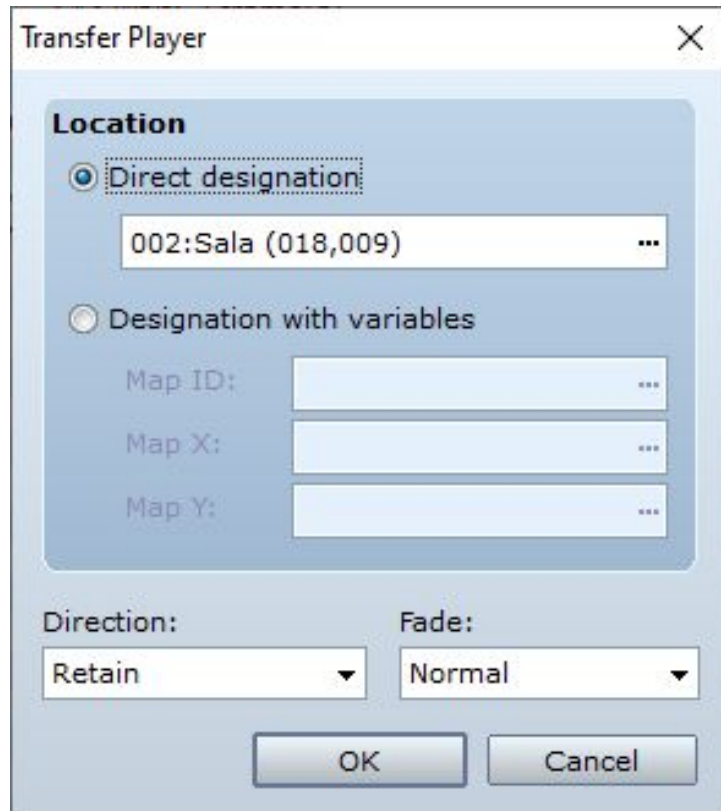
- ❑ Dê duplo clique no campo abaixo de **Direct Designation**. Isso levará para uma tela mostrando a lista de mapas e o mapa selecionado;
- ❑ Selecione o mapa **Sala** e escolha o local onde o jogador deve aparecer após usar a porta, usando duplo clique.
- ❑ Aperte **OK** para fechar a janela **Transfer Player** e outra vez **OK** para fechar o **Edit Event**.

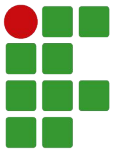




Conhecendo o RPG Maker VX Ace

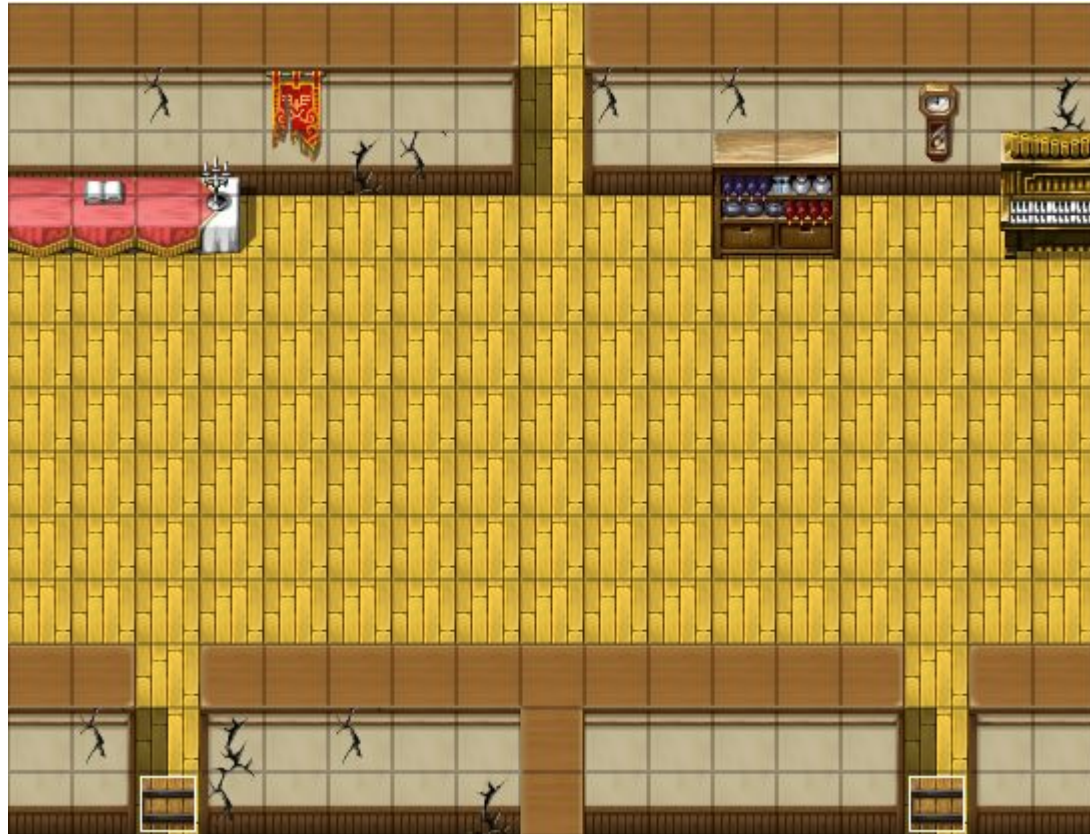
- ❑ Dicas:
 - ❑ Caso queira escolher a direção que o personagem deve estar olhando após ser transferido, escolha no campo abaixo de **Direction**;
 - ❑ No campo abaixo de **Fade** é possível escolher um efeito de transição ao ir de um mapa para outro: escurecendo (Normal), clareando (White) ou sem efeito (None).

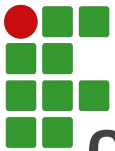




Conhecendo o RPG Maker VX Ace

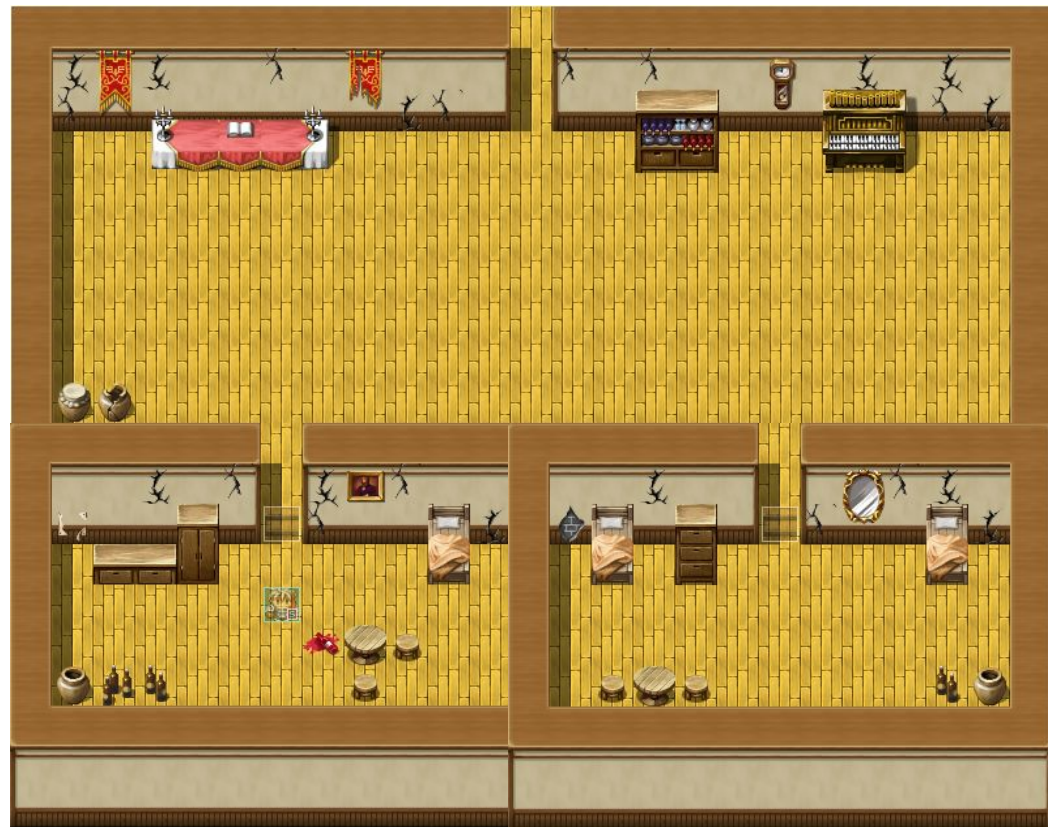
- ❑ Agora, vá até o mapa **Sala**, copie o evento da porta que leva até o **Quarto 01**, cole no vão da porta para o outro quarto e siga o mesmo procedimento para editar o evento, apontando para o local correto para o jogador teleportar.

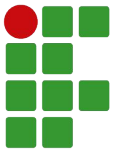




O que criamos até agora?

- ❑ Um projeto de jogo;
- ❑ Três mapas;
- ❑ Algumas portas e sua mecânica.

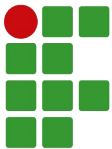




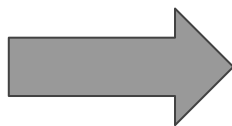
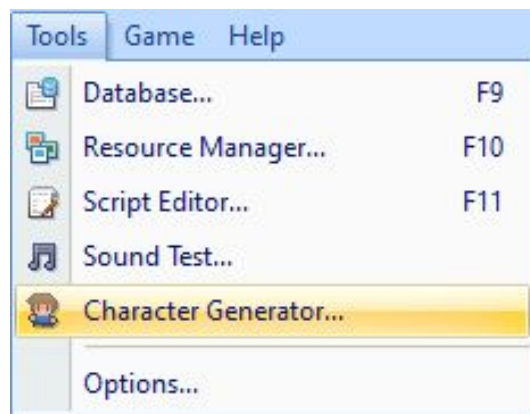
O que criaremos amanhã?

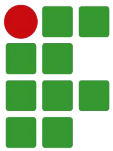
- ❑ O protagonista;
- ❑ Um personagem secundário;
- ❑ Alguns objetos interativos;
- ❑ Alguns eventos de diálogo;
- ❑ Outros eventos do jogo.

Segundo Dia!



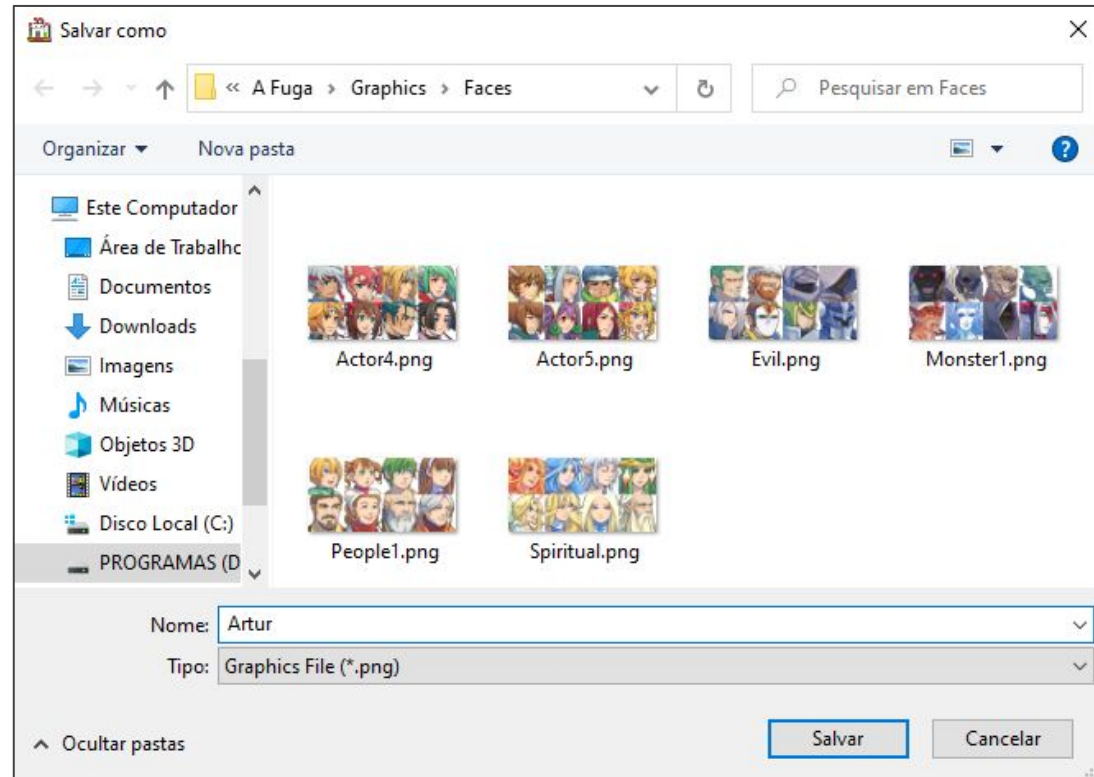
Criando o Protagonista

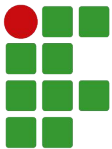




Criando o Protagonista

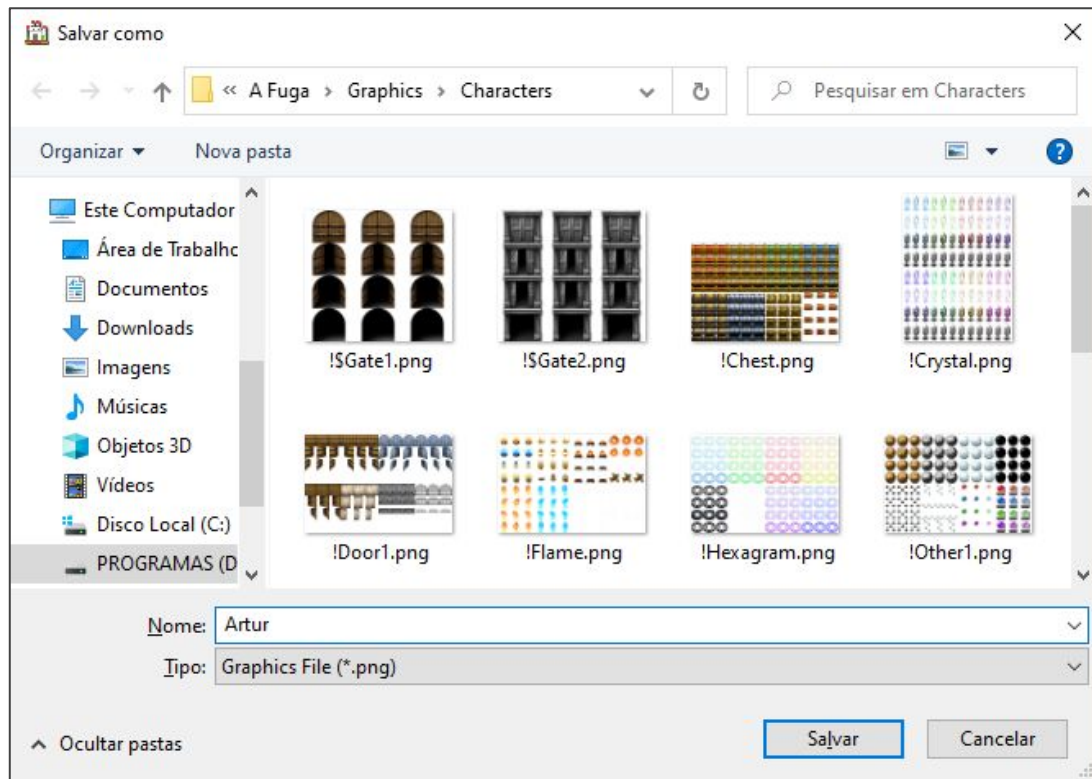
- ❑ Clique no botão “Output Face...”;
- ❑ Veremos uma janela para salvar um arquivo PNG com o rosto do nosso protagonista, apontando o caminho:
A Fuga > Graphics > Faces
- ❑ Vamos salvar o arquivo com o nome “Artur” (nosso protagonista).

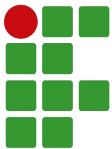




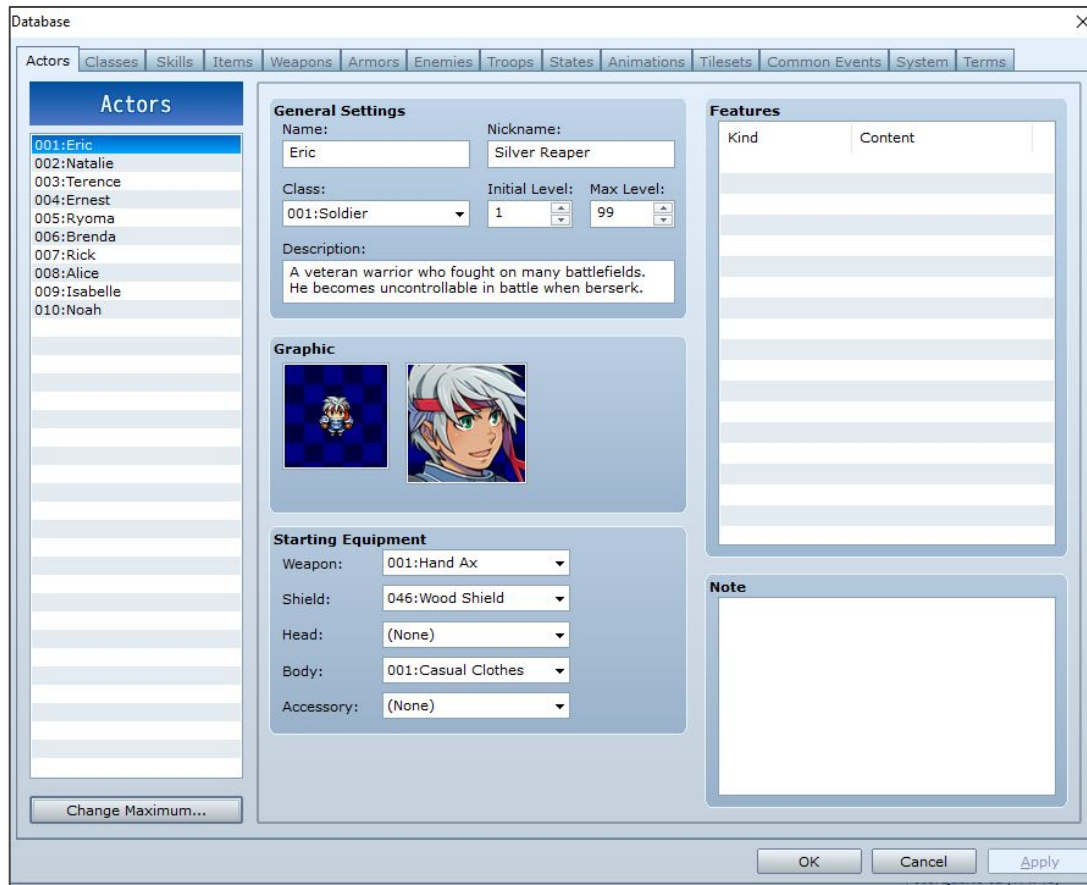
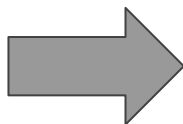
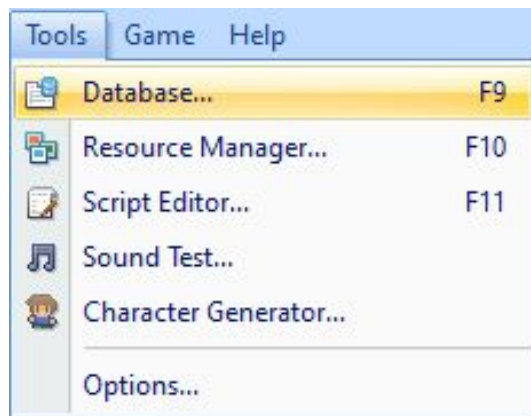
Criando o Protagonista

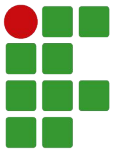
- ❑ Clique no botão “Output Character...”;
- ❑ Veremos uma janela para salvar um arquivo PNG com o rosto do nosso protagonista, apontando o caminho:
A Fuga > Graphics > Characters
- ❑ Vamos salvar também este arquivo com o nome “Artur”;
- ❑ Podemos fechar essa janela (botão “Close”).





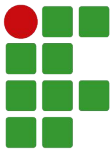
Criando o Protagonista





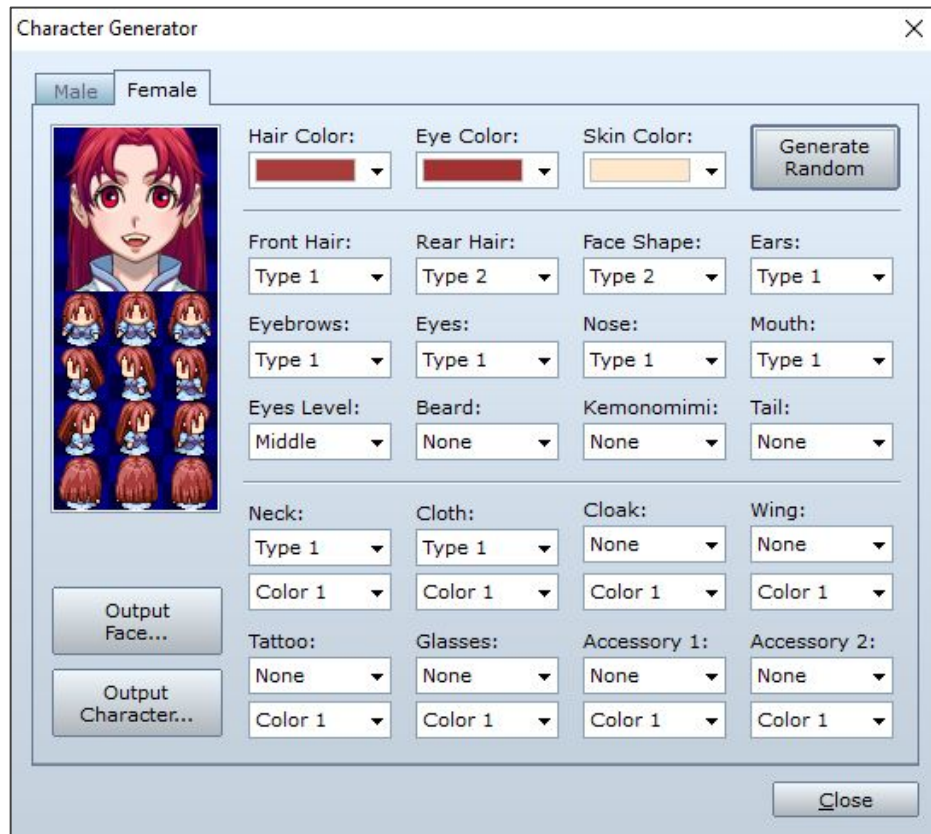
Criando o Protagonista

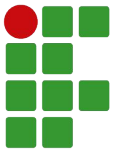
- ❑ Na aba Actors, com o primeiro “ator” selecionado...
- ❑ Altere o nome (name) para “Artur”;
- ❑ Altere o apelido (nickname) para “(Perdido)”;
- ❑ Altere a classe (class) para “Detetive” (vá na aba Class e altere o campo name da primeira classe);
- ❑ Altere a descrição (description) para “Ele não lembra como ele veio parar nesta casa.”;
- ❑ Altere o gráfico de movimentação (duplo clique) para “Artur”;
- ❑ Altere o gráfico do rosto (duplo clique) para “Artur”;
- ❑ Por agora é só. Clique no botão “OK” e salve o projeto.



Criando um Personagem Secundário

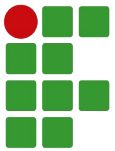
- ❑ Abra novamente o gerador de personagens (*character generator*) e crie um segundo personagem;
- ❑ Crie do jeito que preferir (o meu será uma mulher chamada **Sara**);
- ❑ Lembre de salvar os gráficos do rosto e do personagem andando;
- ❑ Caso queira que ela seja jogável em algum ponto do jogo, use o menu *Tools > Database...* > aba *Actors* para defini-la como um segundo “ator”.





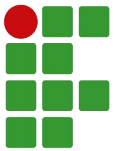
Criando um Personagem Secundário

- ❑ Agora vamos criar um evento na sala para exibir Sara;
- ❑ Podemos usar eventos para criar várias coisas:
 - ❑ Personagens não-jogáveis (e até os jogáveis, enquanto não são jogáveis);
 - ❑ Objetos animados, móveis, pegáveis, quebráveis, transformáveis;
 - ❑ Diálogos entre personagens;
 - ❑ Sequências animadas (cutscenes);
 - ❑ E mais.



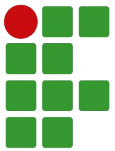
Criando um Personagem Secundário

- ❑ No RPG Maker, um evento pode ter vários comportamentos, cada um definido por uma página (*page*). Cada página de um evento possui:
 - ❑ **Conditions** - as condições necessárias para que aquele comportamento possa ocorrer;
 - ❑ **Graphic** - o gráfico que será exibido no cenário na posição ocupada por aquele evento;
 - ❑ **Movement** - a forma como o evento se move pelo cenário;



Criando um Personagem Secundário

- ❑ Cada página de um evento possui (cont.):
 - ❑ **Options** - define algumas opções sobre a forma como a animação do evento ocorre e se ele pode atravessar tiles não caminháveis;
 - ❑ **Priority** - qual o seu nível/altura em relação ao jogador (abaixo dele, na mesma altura ou acima);
 - ❑ **Trigger** - gatilho que dispara a sequência de ações deste evento (comportamento);
 - ❑ **Contents** - a sequência de ações definida para esta página.



Criando um Personagem Secundário

- ❑ Vá até a “Sala”, entre no modo de edição de evento (F6), clique no meio da sala com o botão direito e escolha “New Event...”;
- ❑ A janela ao lado aparecerá:

New Event - ID:002

Name: EV002

Buttons: New Event Page, Copy Event Page, Paste Event Page, Delete Event Page, Clear Event Page

1

Conditions

- Switch [] is ON
- Switch [] is ON
- Variable [] is
- Self Switch [] is ON
- Item [] exists
- Actor [] exists

Graphic

Autonomous Movement

Type: Fixed

Speed: 3: x2 Slower

Freq: 3: Normal

Options

- Walking Anim.
- Stepping Anim.
- Direction Fix
- Through

Priority

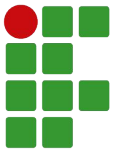
Below Characters

Trigger

Action Button

Contents: @>

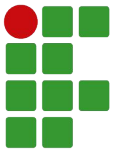
Buttons: OK, Cancel, Apply



Criando um Personagem Secundário

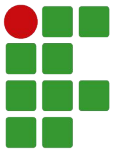
- ❑ Altere o nome (*name*) para **Sara**;
- ❑ Altere o gráfico (*graphic*) para **Sara**;
- ❑ Altere o tipo de movimento (*movement*) para aleatório (*random*);
- ❑ Altere a prioridade (*priority*) para igual ao jogador (*same as characters*);
- ❑ Altere o gatilho (*trigger*) para pressionar tecla (*action button*);
- ❑ Pressione “OK” para fechar essa janela. Salve e teste o jogo.





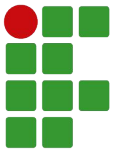
Criando Objetos Interativos

- ❑ Objetos no cenário que podem ser adicionados, removidos, empurrados ou transformados devem ser criados como eventos;
- ❑ E se quisermos que, ao pegar um item, o mesmo apareça em nosso inventário, devemos criar um item para ele no *Database* (do tipo *normal* ou *key item*);
- ❑ Em nosso jogo, o jogador poderá ler uma partitura, pegar uma faca (para soltar as teclas do órgão), tocar um órgão e empurrar uma estátua!



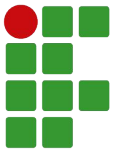
Criando Objetos Interativos

- ❑ Vamos criar primeiro a partitura!
- ❑ Vá até o “Quarto 02” e, no modo de edição de eventos, clique com o botão direito sobre a mesa e escolha “New Event...”;
- ❑ Nomeie como “Partitura”, escolha como gráfico o desenho da carta no “Tileset-C”, prioridade “*same as characters*” e gatilho “*action button*”;
- ❑ Agora, façamos o protagonista ler a partitura. Para isso, usaremos um dos vários *comandos* disponíveis no RPG Maker.



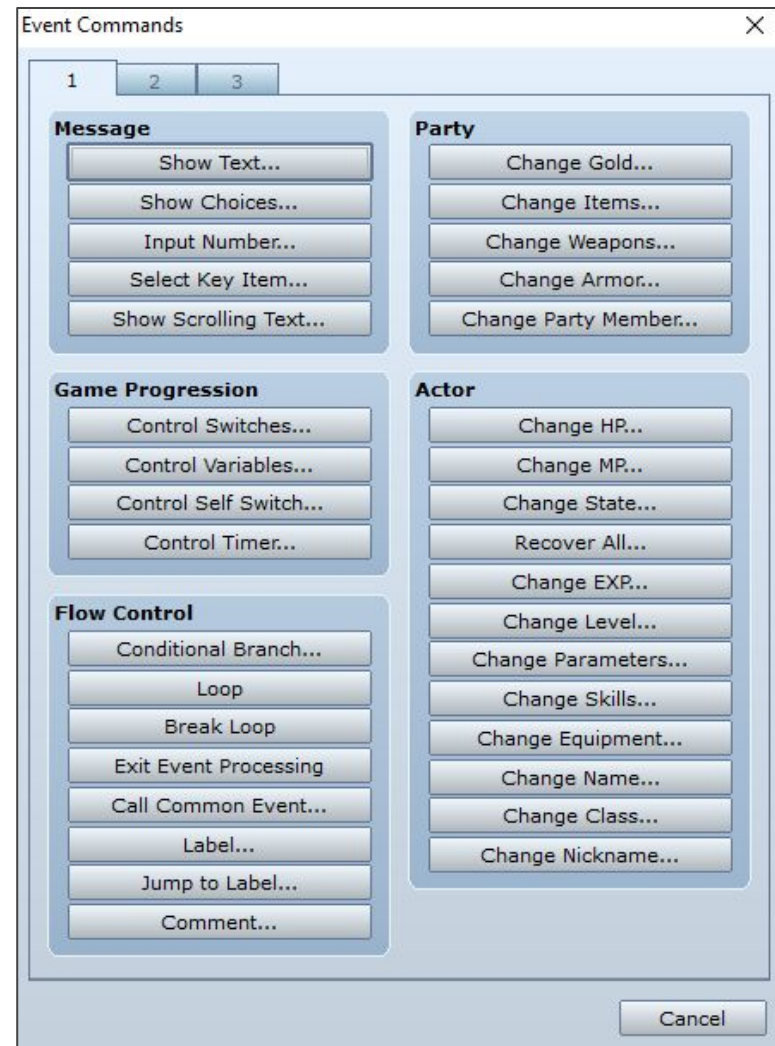
Criando Objetos Interativos

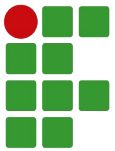
- ❑ Comandos são ações que podem ser usadas para exibir mensagens na tela (simples textos ou diálogos), alterar informações no jogo (variáveis, interruptores e timers), definir condições, repetir comandos, alterar dados da equipe, de um ator, transferir o protagonista para outro local, definir uma rota de movimento, efeitos de tela (fade-in, fade-out, piscar, mudar a tonalidade/cor), exibir uma imagem, controlar efeito sonoro ou música, iniciar uma batalha, iniciar sistema de loja, alterar configurações do jogo etc.



Criando Objetos Interativos

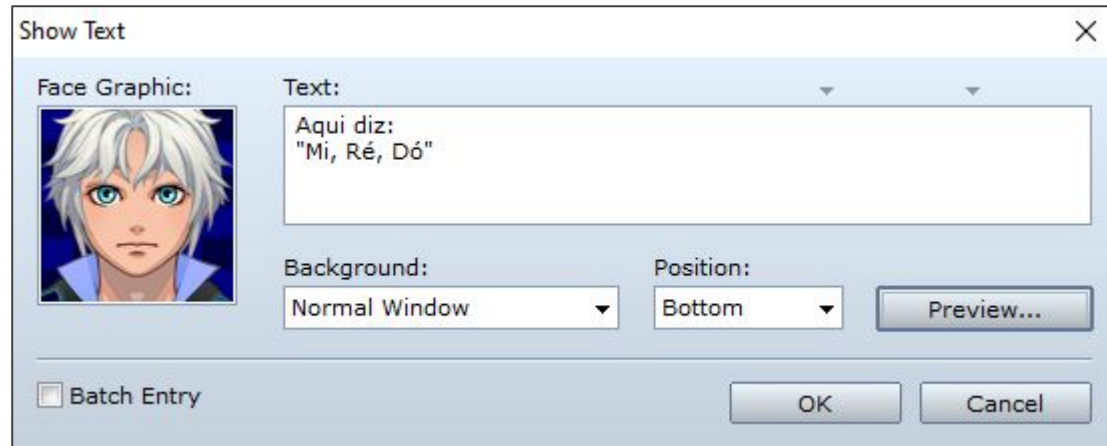
- Na área *contents* (à direita), clique com o botão direito e escolha “Insert...” (ou aperte *Enter*);
- Você verá a tela ao lado:

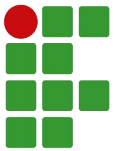




Criando Objetos Interativos

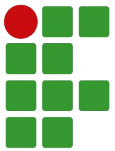
- ❑ Na primeira aba, clique no botão “Show Text...”;
- ❑ Você verá a tela abaixo;
- ❑ Escolha o rosto de “Artur” e digite o texto:
Aqui diz:
“Mi, Ré, Dó”
- ❑ Clique em “OK”.





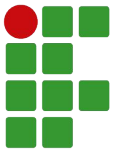
Criando Objetos Interativos

- ❑ Agora vamos criar o segundo objeto, a faca!
- ❑ O protagonista a verá no chão do primeiro quarto. Ao interagir, perguntará se devemos pegá-la ou não. Se escolher sim, ganharemos um item “Faca” e ela sumirá do cenário. Para isso, usaremos um comando para exibir escolhas (*choices*) para o jogador.
- ❑ Vamos primeiro criar um item “Faca”. Para isso, vá em: *Tools > Database... > aba Items*.



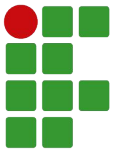
Criando Objetos Interativos

- ❑ Clique no primeiro item e configure-o da seguinte forma:
 - ❑ Name - Faca;
 - ❑ Icon - Espada curta (index: 147);
 - ❑ Description - “Pode ser usada para cortar ou separar coisas”;
 - ❑ Tipo - *key item* (item-chave);
 - ❑ Consume - *no*.
 - ❑ Occasion - *never*.



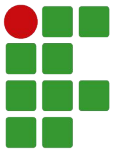
Criando Objetos Interativos

- ❑ Agora vá até o “Quarto 01”, clique no primeiro *tile* de chão, à esquerda dos móveis, com o botão direito e escolha “New Event..”;
- ❑ Nomeie como “Faca”, escolha como gráfico o desenho da espada curta no “Tileset-C”, prioridade “*same as characters*” e gatilho “*action button*”;
- ❑ Na área *contents*, insira um comando “Show Text..”, mostrando o rosto de Artur com a mensagem “Encontrei uma faca. Devo pegar?”.



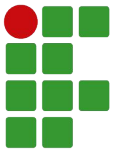
Criando Objetos Interativos

- ❑ Logo abaixo vamos inserir outro comando, dessa vez usando “Show Choices...”. Lá, defina as opções “Sim” e “Não”;
- ❑ Selecione a linha abaixo da opção “Sim” (*When [Sim]*) e insira um novo comando, dessa vez o “Change Items...”;
- ❑ Escolha o item “Faca”, com a opção aumentar (*increase*) e o valor constante (*constant*) 1. Isso vai aumentar a quantidade de facas que temos em um. Clique em “OK”;
- ❑ Logo abaixo do comando para adicionar uma faca aparecer a mensagem “Você pegou uma faca.” (Sem rosto).



Criando Objetos Interativos

- ❑ Estamos quase lá, agora precisamos remover a faca do chão. Para isso, crie uma nova página no botão “New Event Page”;
- ❑ Na área de condições (conditions), marque “Item” e selecione a “Faca”;
- ❑ Isso fará com que, caso tenhamos a faca, este evento tenha o comportamento da página 2, isto é, sem gráfico (não aparecendo no cenário) e ficando abaixo do protagonista (*below characters*, assim não atrapalha a movimentação do jogador).



Criando Objetos Interativos

Edit Event - ID:002

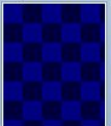
Name:

1 2

Conditions

- Switch ... is ON
- Switch ... is ON
- Variable ... is
or above
- Self Switch is ON
- Item exists
- Actor exists

Graphic



Autonomous Movement

Type:

Speed:

Freq:

Options

- Walking Anim.
- Stepping Anim.
- Direction Fix
- Through

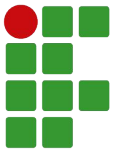
Priority

Trigger

Contents:

@>

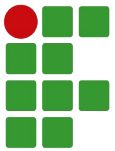




Criando Objetos Interativos

- ❑ Agora vamos criar a interação com o terceiro objeto, o piano!
- ❑ Em nosso jogo, três estátuas bloquearão a porta de saída (cada estátua será composta por dois eventos, o de cima com a prioridade “above characters” e o de baixo com a prioridade “same as characters”);
- ❑ Nossa sala ficará assim (os gráficos das três estátuas encontram-se no “Tileset-C”). Vamos por partes...



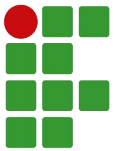


Criando Objetos Interativos

Estátua à esquerda:

- ❑ Evento de cima:
 - ❑ Name: Estatua1_Cima
 - ❑ Priority: Above Characters
- ❑ Evento de baixo:
 - ❑ Name: Estatua1_Baixo
 - ❑ Priority: Same As Characters
 - ❑ Contents: Faça com que Artur diga “Como posso tirar essa estátua?”





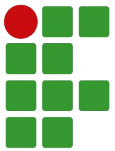
Criando Objetos Interativos

Estátua do meio:

- ❑ Evento de cima:
 - ❑ Name: Estatua2_Cima
 - ❑ Priority: Above Characters
 - ❑ Options: deixe marcadas *Walking anim.* e *Through*

- ❑ Evento de baixo:
 - ❑ Name: Estatua2_Baixo
 - ❑ Priority: Same As Characters
 - ❑ Contents: Faça com que Artur diga “Como posso tirar essa estátua?”



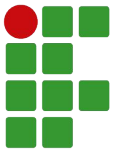


Criando Objetos Interativos

Estátua à direita:

- ❑ Evento de cima:
 - ❑ Name: Estatua3_Cima
 - ❑ Priority: Above Characters
- ❑ Evento de baixo:
 - ❑ Name: Estatua3_Baixo
 - ❑ Priority: Same As Characters
 - ❑ Contents: Faça com que Artur diga “Como posso tirar essa estátua?”

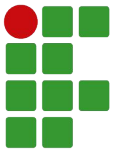




Criando Objetos Interativos

- ❑ E agora, vamos ao órgão. Crie um novo evento sobre as teclas do meio do mesmo. Nomeie esse evento “Órgão” e defina a prioridade como “mesma que os personagens” (*same as characters*) e o gatilho, botão de ação (*action button*);
- ❑ Nessa primeira página, vamos colocar o comportamento padrão do protagonista (isto é, enquanto ele não tiver a faca), que será o Artur dizer “Não dá para tocar, as teclas estão emperradas.”.

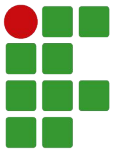




Criando Objetos Interativos

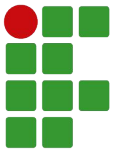
- ❑ Agora, clique no botão “New Event Page” para criar a segunda página. A condição será ter o item “Faca” e a prioridade também será a mesma que os personagens e o gatilho ainda é pressionar um botão;
- ❑ Caso o usuário pressione um botão, vamos guardar a informação de que o órgão foi reparado em um interruptor.





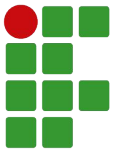
Criando Objetos Interativos

- ❑ O RPG Maker oferece duas formas para guardarmos informações: interruptores e variáveis;
- ❑ Interruptores - podem armazenar valores do tipo “verdadeiro” (ON) ou falso” (OFF). Por padrão começam com “falso”. Exemplos de situações para uso:
 - ❑ Se a porta está trancada (PortaTrancada);
 - ❑ Se o jogador conversou com alguém (ConversouCarlos).



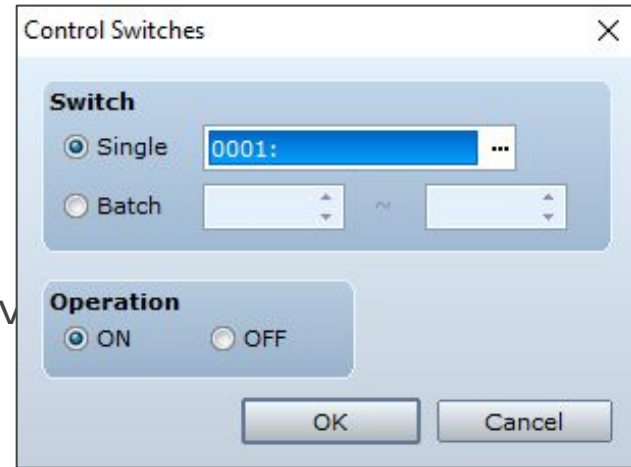
Criando Objetos Interativos

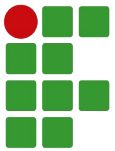
- ❑ Variáveis - podem armazenar valores numéricos ou textuais. Por padrão começam com zero (0). Exemplos de situações para uso:
 - ❑ Idade de um personagem (IdadeMike);
 - ❑ Total de chutes dados (Chutes);
 - ❑ Uma senha digitada (Senha).
- ❑ Essas informações são salvas quando o jogo é salvo.



Criando Objetos Interativos

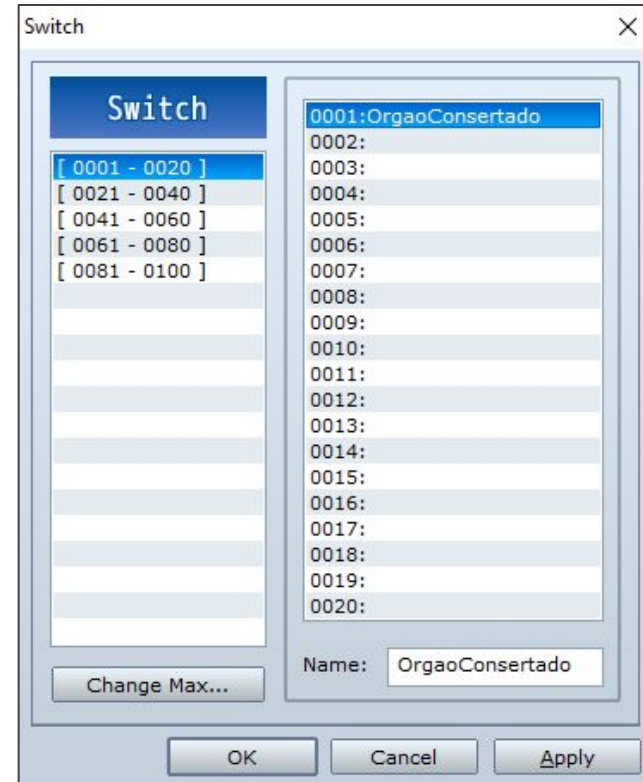
- ❑ Vamos descrever os comandos da 2ª página;
- ❑ Primeiro, Artur deve perguntar se “Dev usar a faca para consertar o órgão?”;
- ❑ Em seguida, mostre duas opções: “Sim” “Não”. Caso o jogador escolha “Sim”, use o comando “Control Switches...”. A tela ao lado aparecerá:

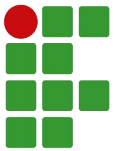




Criando Objetos Interativos

- ❑ Dê um duplo clique na caixa ao lado de “Single” para criar um novo interruptor ou escolher um já em uso. Escolha o primeiro interruptor, dê um nome a ele (OrgaoConsertado) e clique em “OK”;
- ❑ Deixe-o ligado (operation = ON). Clique em “OK” para fechar a janela “Control Switches”;
- ❑ Ainda continuando a escolha “Sim”, Artur deveria dizer “Pronto, novinho em folha.”.





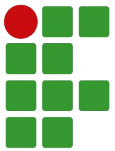
Criando Objetos Interativos

Assim ficou a 2ª página do nosso evento:

The screenshot shows a software interface for creating interactive events. At the top, there is a 'Name:' field containing 'Órgão' and five buttons: 'New Event Page', 'Copy Event Page', 'Paste Event Page', 'Delete Event Page', and 'Clear Event Page'. Below the name field are three tabs labeled '1', '2', and '3', with '2' being the active tab.

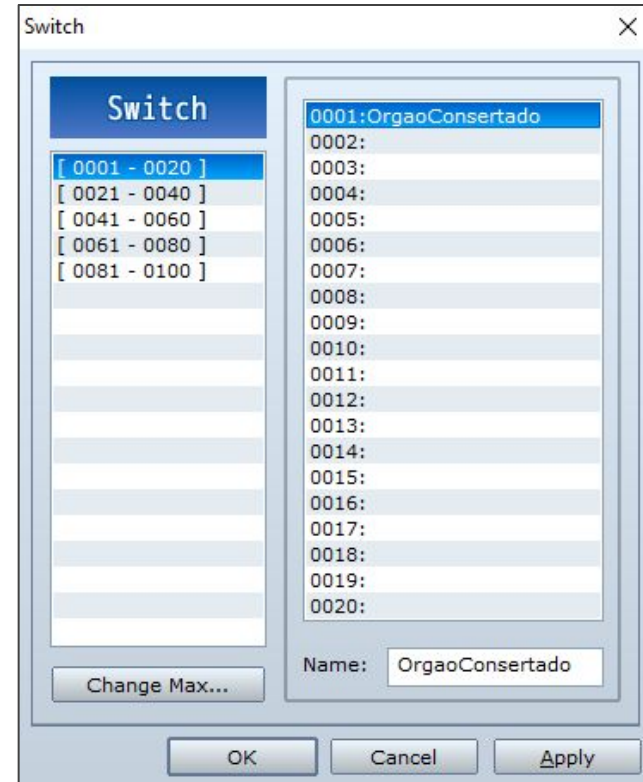
The main area is divided into two panels:

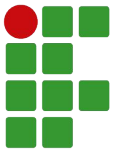
- Conditions:** A list of conditions with checkboxes and input fields:
 - Switch [] is ON
 - Switch [] is ON
 - Variable [] is
 - [] or above
 - Self Switch [] is ON
 - Item: 001:Faca exists
 - Actor [] exists
- Contents:** A list of content elements:
 - @>Text: 'Artur', 0, Normal, Bottom
: : Devo usar a faca para consertar o órgão?
 - @>Show Choices: Sim, Não
: When [Sim]
@>Control Switches: [0001:OrgaoConsertado] = ON
 - @>Text: 'Artur', 0, Normal, Bottom
: : Pronto, novinho em folha.
 - @>
 - : When [Não]
 - @>
 - : Branch End
 - @>



Criando Objetos Interativos

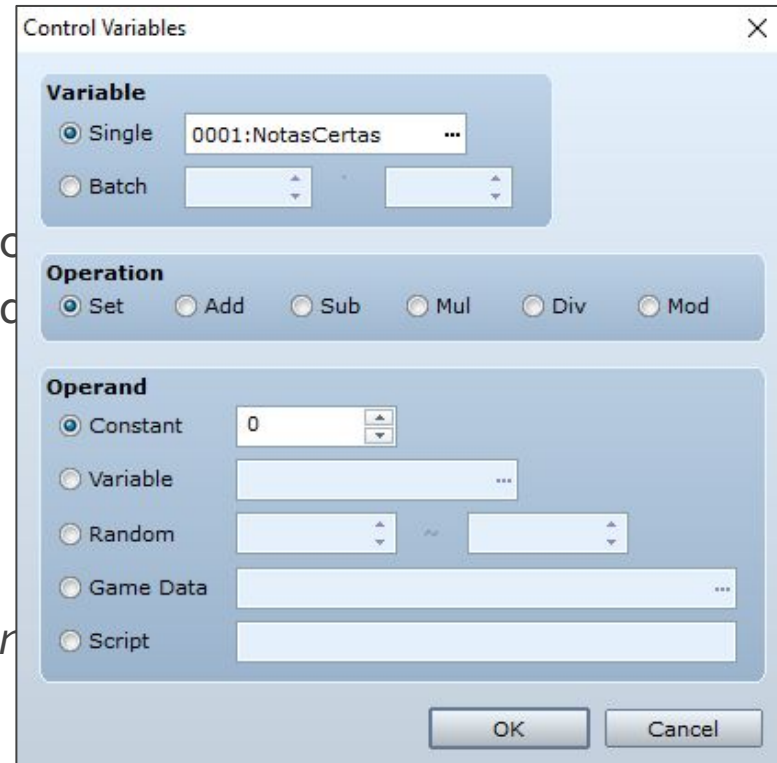
- ❑ Crie uma terceira página. Agora a condição será o switch “OrgaoConsertado” estar ligado (ON). A prioridade é a mesma que os personagens;
- ❑ Desta vez, o jogador precisará tocar três notas musicais. Se ele acertar a ordem das notas na partitura, a tela se apagará as as estátuas sumirão.

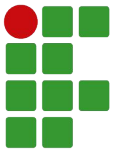




Criando Objetos Interativos

- ❏ Vamos começar configurando uma variável “NotasCertas” com o valor zero. Para isso, escolha o comando “Control Variables...”, clique no campo e defina uma primeira variável chamada “NotasCertas”, mantenha a operação como “set” (configurar) e o operando *constant* igual a 0. Clique em “OK”.





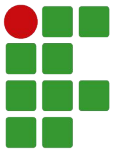
Criando Objetos Interativos

- ❑ Agora, Artur deve se perguntar “O que devo tocar?”;
- ❑ Mostre as opções “Dó”, “Ré”, “Mi” e “Fá” (desabilite o *when cancel*). Caso ele escolha “Mi”, adicione 1 ao valor de NotasCertas.

The screenshot shows a dialog box titled "Control Variables" with a close button (X) in the top right corner. The dialog is divided into three main sections:

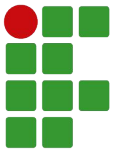
- Variable:** Contains two radio buttons. The "Single" radio button is selected, and its corresponding text field contains "0001:NotasCertas". The "Batch" radio button is unselected, and its corresponding fields are empty.
- Operation:** Contains five radio buttons: "Set", "Add", "Sub", "Mul", "Div", and "Mod". The "Add" radio button is selected.
- Operand:** Contains five radio buttons: "Constant", "Variable", "Random", "Game Data", and "Script". The "Constant" radio button is selected, and its corresponding text field contains the value "1". The other operand options are unselected and their corresponding fields are empty.

At the bottom of the dialog, there are two buttons: "OK" and "Cancel".



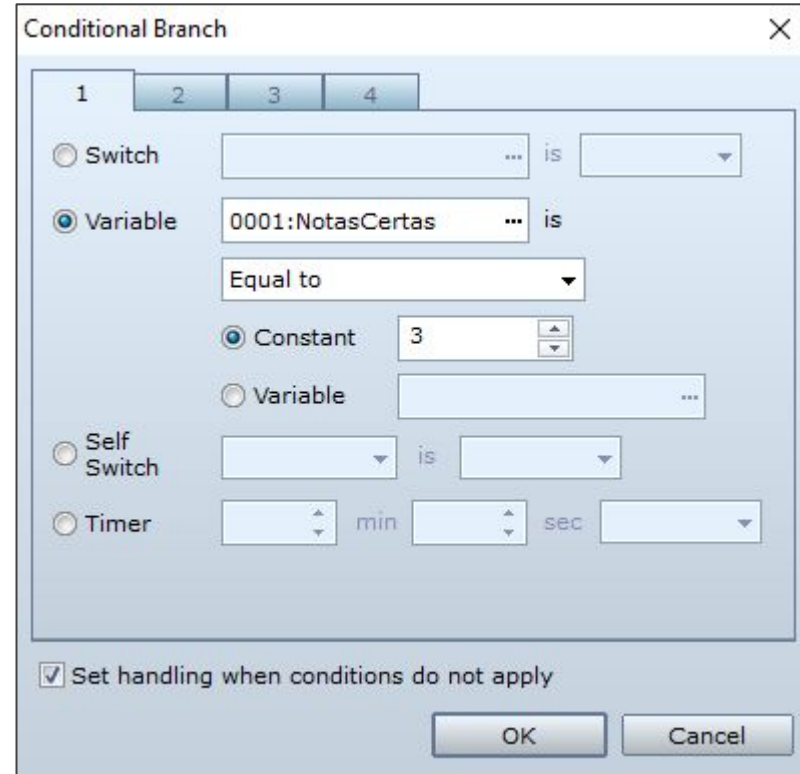
Criando Objetos Interativos

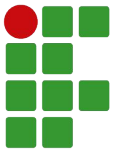
- ❑ Vá para a última linha e novamente mostre as opções “Dó”, “Ré”, “Mi” e “Fá” (desabilite o when cancel). Agora, caso ele escolha “Ré”, adicione 1 ao valor de NotasCertas;
- ❑ Mais uma vez, vá para o final da lista de comandos e mostre as opções “Dó”, “Ré”, “Mi” e “Fá” (desabilite o when cancel). Agora, caso ele escolha “Dó”, adicione 1 ao valor de NotasCertas;
- ❑ Após o jogador escolher as três notas, precisamos testar se ele acertou as três (NotasCertas = 3). Se sim, removeremos as estátuas da esquerda e da direita. Para testar se NotasCertas = 3, precisaremos de um comando condicional: Conditional Branch...



Criando Objetos Interativos

- Um comando condicional (conditional branch) permite testar uma condição e, se ela for verdadeira, executaremos um conjunto de comandos; caso contrário, podemos simplesmente não fazer nada ou executar outro conjunto de comandos;
- Crie um comando condicional (*Conditional Branch...*), marque a opção *variable*, escolha a variável *NotasCertas* e teste se ela é igual (Equal to) a constante 3. Deixe o “set handling when conditions do not apply” para especificarmos o que ocorrerá quando a condição falhar. Clique em “OK”.

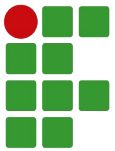




Criando Objetos Interativos

- ❑ Selecione a área cujos comandos serão executados em caso verdadeiro (abaixo do “*Conditional Branch:*”);
- ❑ Faça a tela escurecer (*Fadeout Screen*, na aba 2);
- ❑ Faça um interruptor “EnigmaResolvido” ser ligado (igual a 0N);
- ❑ Clareie novamente a tela (*Fadein Screen*, na aba 2);
- ❑ Faça Artur dizer “Ué, o que aconteceu?”

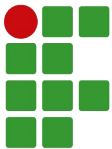
```
@>Conditional Branch: Variable [0001:NotasCertas] == 3
@>
: Else
@>
: Branch End
@>
```



Criando Objetos Interativos

- ❑ Selecione a área cujos comandos serão executados em caso falso (abaixo do “*Else*”);
- ❑ Faça Artur dizer “Nada aconteceu.”

```
@>Conditional Branch: Variable [0001:NotasCertas] == 3
  @>Fadeout Screen
  @>Control Switches: [0002:EnigmaResolvido] = ON
  @>Text: 'Artur', 0, Normal, Bottom
    :      : Ué, o que aconteceu?
  @>
: Else
  @>
: Branch End
@>
```

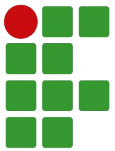


Criando Objetos Interativos

Assim ficou a 3ª página do nosso evento:

The screenshot displays a software interface for creating interactive objects, divided into three main sections:

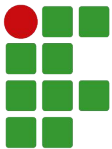
- Conditions:** A panel on the left with a 'Conditions' header. It contains several rows of controls:
 - A checked 'Switch' with the text '0001:OrgaoConsertad...' and 'is ON'.
 - An unchecked 'Switch' with a blank field and 'is ON'.
 - An unchecked 'Variable' with a blank field and 'is'.
 - A dropdown menu with 'or above' selected and 'is ON'.
 - An unchecked 'Self Switch' with a dropdown menu and 'is ON'.
 - An unchecked 'Item' with a dropdown menu and 'exists'.
 - An unchecked 'Actor' with a dropdown menu and 'exists'.
- Contents:** A central panel titled 'Contents' showing a list of object actions:
 - @>Control Variables: [0001:NotasCertas] = 0
 - @>Text: 'Artur', 0, Normal, Bottom
 - : O que devo tocar?
 - @>Show Choices: Dó, Ré, Mi, Fá
 - : When [Dó]
 - @>
 - : When [Ré]
 - @>
 - : When [Mi]
 - @>Control Variables: [0001:NotasCertas] += 1
 - @>
 - : When [Fá]
 - @>
 - : Branch End
 - @>Show Choices: Dó, Ré, Mi, Fá
 - : When [Dó]
 - @>
 - : When [Ré]
 - @>Control Variables: [0001:NotasCertas] += 1
 - @>
 - : When [Mi]
 - @>
 - : When [Fá]
 - @>
 - : Branch End
- Code:** A right-hand panel showing the underlying code for the actions:
 - @>Show Choices: Dó, Ré, Mi, Fá
 - : When [Dó]
 - @>Control Variables: [0001:NotasCertas] += 1
 - @>
 - : When [Ré]
 - @>
 - : When [Mi]
 - @>
 - : When [Fá]
 - @>
 - : Branch End
 - @>Conditional Branch: Variable [0001:NotasCertas] == 3
 - @>Fadeout Screen
 - @>Control Switches: [0002:EnigmaResolvido] = ON
 - @>Fadein Screen
 - @>Text: 'Artur', 0, Normal, Bottom
 - : Ué, o que aconteceu?
 - @>
 - : Else
 - @>Text: 'Artur', 0, Normal, Bottom
 - : Nada aconteceu.
 - @>
 - : Branch End
 - @>



Criando Objetos Interativos

- ❑ Agora vamos fazer as partes das estátuas da esquerda e da direita desaparecer quando o enigma estiver resolvido!
- ❑ Para isso, selecione cada uma das quatro partes, crie uma nova página de evento, configure o switch para EnigmaResolvido, mantenha sem gráfico e a prioridade como abaixo dos personagens;
- ❑ Vamos testar nosso jogo!





Criando Objetos Interativos

- Finalmente, vamos fazer o jogador empurrar a estátua do meio!
- Toda a interação ocorrerá com a base dela, então selecione o evento da parte de baixo (Estatua2_Baixo), escolha editar o evento e faça uma cópia da página atual (clique em *copy event page*, depois em *paste event page*);
- A condição para esta segunda página será o interruptor (*switch*) EnigmaResolvido estar ligado.

Edit Event - ID:008

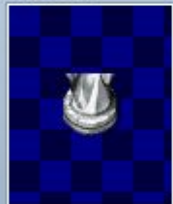
Name:

1 2

Conditions

- Switch is ON
- Switch ... is ON
- Variable is or above
- Self Switch is ON
- Item exists
- Actor exists

Graphic



Autonomous Movement

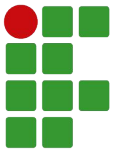
Type:

Speed:

Freq:

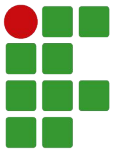
Contents:

@>



Criando Objetos Interativos

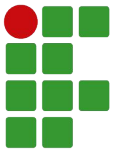
- ❑ Nossa estátua pode se mover para a esquerda e para a direita, mas não pode se mover nas outras direções. Teremos três situações:
 - ❑ Se empurrando para a esquerda;
 - ❑ Se empurrando para a direita;
 - ❑ Senão, empurrando para uma direção errada.
- ❑ Vamos resolver a primeira situação, criando um comando condicional (*conditional branch*) e verificando se o personagem está empurrando (olhando) para a esquerda (na aba 3, escolha *Character “Player” is facing “Left”*).



Criando Objetos Interativos

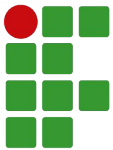
- ❑ Até agora, temos algo como a figura abaixo;
- ❑ Agora vamos descrever o passo a passo para o empurrar a estátua.

```
@> Conditional Branch: Player is Facing Left
@>
: Else
@>
: Branch End
@>
```

Criando Objetos Interativos

- ❑ Ao empurrar a estátua, nós queremos que:
 - ❑ Execute um som de algo sendo empurrado;
Play SE... → Push
 - ❑ Mova a parte de cima da estátua...
Set Move Route... → escolha Estatua2_Cima → clique “Move Left”
Desmarque “Wait for Completion” e clique “OK”
 - ❑ ...e ao mesmo tempo a parte de baixo.
Set Move Route... → escolha Estatua2_Baixo → clique “Move Left”
Mantenha “Wait for Completion” marcado e clique “OK”
- ❑ Guarde em interruptor que a estátua foi movida
Control Switches... → EstatuaMovida = ON

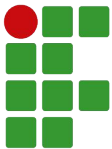


Criando Objetos Interativos

- Seu código deveria estar mais ou menos assim...

```
@>Conditional Branch: Player is Facing Left
  @>Play SE: 'Push', 80, 100
  @>Set Move Route: [Estatua2_Cima]
:                               : $>Move Left
  @>Set Move Route: [Estatua2_Baixo] (Wait)
:                               : $>Move Left
  @>Control Switches: [0003:EstatuaMovida] = ON
  @>
: Else
  @>
: Branch End
@>
```

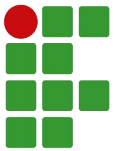
- Agora, vamos copiar esse bloco todo (clique em *Conditional Branch* e pressione Ctrl+C), colá-lo logo abaixo do *Else* (use Ctrl+V) e edite-o tal que a condição seja o jogador estar olhando para a direita (*Facing "Right"*) e mova ambas as partes para a direita (*Move Right*), ficando assim...



Criando Objetos Interativos

- ❑ Até agora, o jogador já pode empurrar a estátua para a esquerda e para a direita;
- ❑ Caso ele tente empurrar em outra direção (*Else*), Artur deve dizer “Não consigo empurrá-la nessa direção.”;
- ❑ Crie uma nova página (ou copie e cole esta), cuja condição é `EstatuaMovida = ON` e não apresente nenhum comando;
- ❑ Vamos testar o nosso jogo!

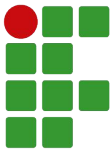
```
@> Conditional Branch: Player is Facing Left
  @> Play SE: 'Push', 80, 100
  @> Set Move Route: [Estatua2_Cima]
    :                : $>Move Left
  @> Set Move Route: [Estatua2_Baixo] (Wait)
    :                : $>Move Left
  @> Control Switches: [0003:EstatuaMovida] = ON
  @>
: Else
  @> Conditional Branch: Player is Facing Right
    @> Play SE: 'Push', 80, 100
    @> Set Move Route: [Estatua2_Cima]
      :                : $>Move Right
    @> Set Move Route: [Estatua2_Baixo] (Wait)
      :                : $>Move Right
    @> Control Switches: [0003:EstatuaMovida] = ON
    @>
  : Else
    @>
  : Branch End
  @>
: Branch End
@>
```



Criando Objetos Interativos

- ❑ Espere! Por que a estátua não se moveu?
- ❑ Apesar de não aparecerem mais (criamos uma página sem gráficos), as estátuas 1 e 3 ainda estão lá, e no RPG Maker há algumas restrições quanto a eventos se sobreporem;
- ❑ Então, precisamos apagar esses eventos!



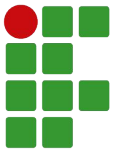


Criando Objetos Interativos

- Edite cada um dos 04 eventos, adicionando na segunda página o comando para apagar o evento atual (*Erase Event*, na aba 2) e mudando seu gatilho para execução automática (*autorun*);
- Vamos testar novamente!

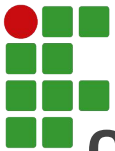
The screenshot displays the 'Edit Event' interface for event ID 004. The event name is 'Estatua1_Cima'. The interface is divided into several sections:

- Name:** Estatua1_Cima
- Buttons:** New Event Page, Copy Event Page, Pas Event Page
- Conditions:**
 - Switch: 0002:EnigmaResolvido ... is ON
 - Switch: ... is ON
 - Variable: ... is
 - Self Switch: ... is ON
 - Item: ... exists
 - Actor: ... exists
- Graphic:** A blue and black checkered pattern.
- Autonomous Movement:**
 - Type: Fixed
 - Speed: 3: x2 Slower
 - Freq: 3: Normal
- Options:**
 - Walking Anim.
 - Stepping Anim.
 - Direction Fix
 - Through
- Priority:** Below Characters
- Trigger:** Autorun
- Contents:** @>Erase Event @>



Fechando as portas!

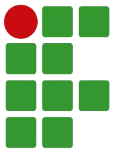
- ❑ Vamos editar os eventos das portas para os quartos para o jogador não ir para os quartos caso já tenha movido a estátua (`interruptor EstatuaMovida = ON`);
- ❑ Copia e cola a página;
- ❑ Condição switch `EstatuaMovida = ON`;
- ❑ Apague os antigos comandos;
- ❑ Artur diz: **É melhor dar o fora daqui!**



O que criamos até agora?

- ❑ O protagonista;
- ❑ Um personagem secundário;
- ❑ Alguns objetos interativos.

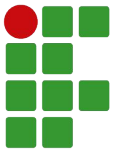




O que criaremos amanhã?

- ❑ Alguns eventos de diálogo;
- ❑ Eventos de abertura e encerramento do jogo;
- ❑ Próximos passos (como continuar aprendendo).

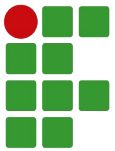
Terceiro Dia!



Criando a Abertura do Jogo

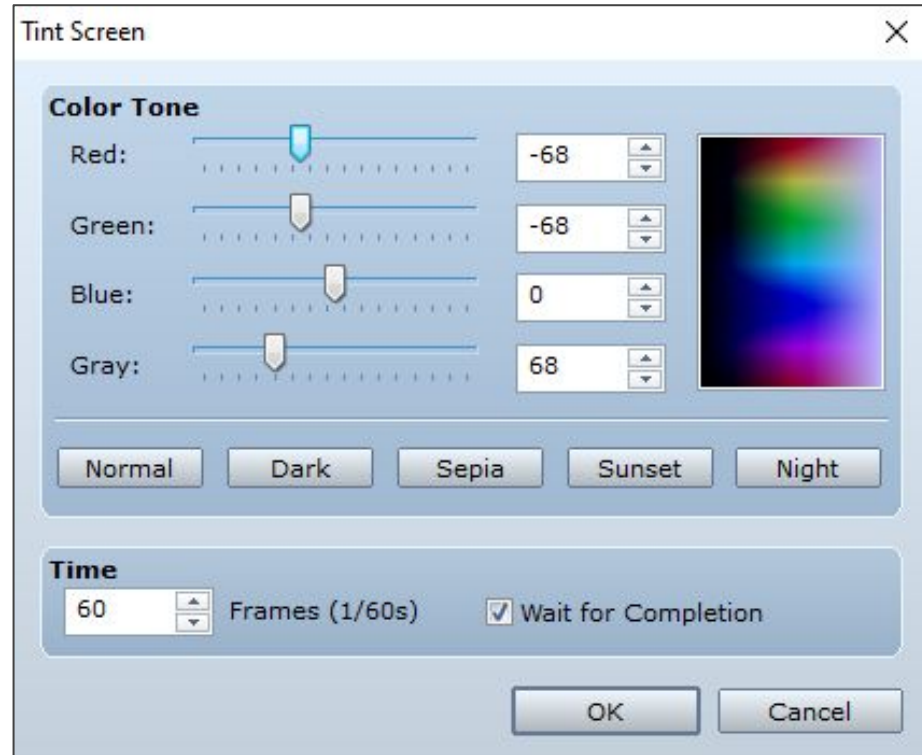
- ❑ Vamos criar uma animação de abertura para o nosso jogo, na qual o protagonista menciona não saber onde está ou como chegou ali;
- ❑ Para isso:
 - ❑ Vá para o mapa em que o jogo começa (**Quarto 01**) e escolha o modo de edição de evento (**F6**);
 - ❑ Clique com o botão direito na parte superior da cama e marque ali como posição inicial para o jogador (**Set as Starting Position** → **Player**);
 - ❑ Clique com o botão direito em um local qualquer no mapa e crie um novo evento (**New Event...**).

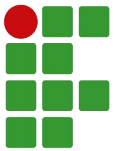




Criando a Abertura do Jogo

- ❑ Chame o novo evento de **Introducao** e configure seu trigger para **Autorun**. Agora, na área dos comandos (**Contents**), vamos executar os seguintes comandos:
 - ❑ Escurecer a tela (aba 2) **Tint Screen...** → **Night**





Criando a Abertura do Jogo

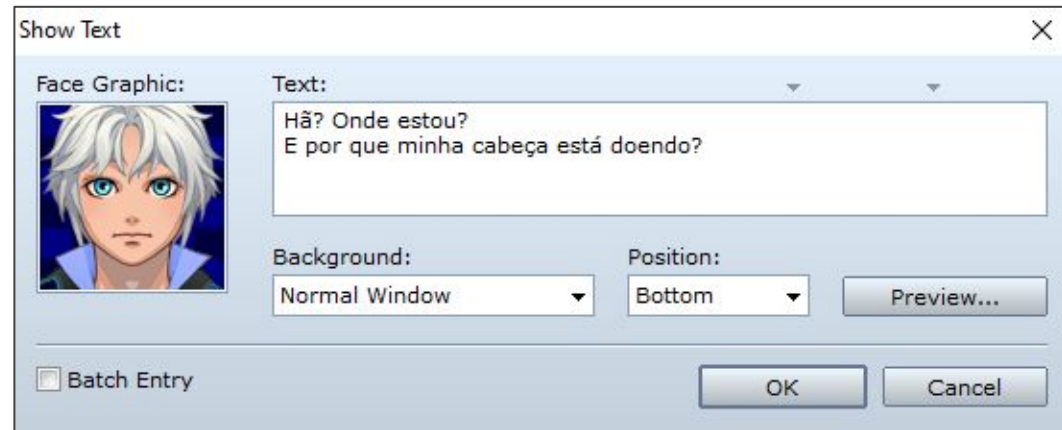
☐ (cont.):

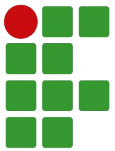
☐ Artur diz:

(aba 1) Show Text...

Face Graphic: Artur

Text: Hã? Onde estou? E porque minha cabeça está doendo?





Criando a Abertura do Jogo

- (cont.):
 - Faça Artur pular da cama:
(aba 2) **Set Move Route...**
Jump: X -1, Y +1

Move Route

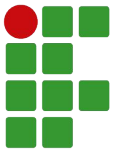
Player

\$>Jump: -1,+1

\$>

Options

- Repeat Action
- Skip If Cannot Move
- Wait for Completion



Criando a Abertura do Jogo

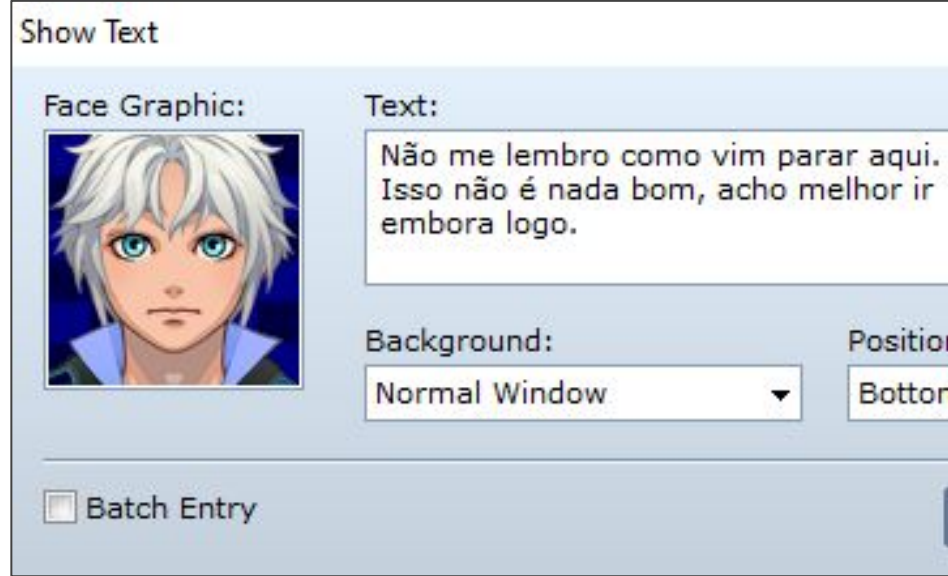
☐ (cont.):

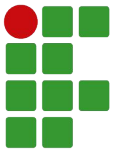
☐ Artur diz:

(aba 1) Show Text...

Face Graphic: Artur

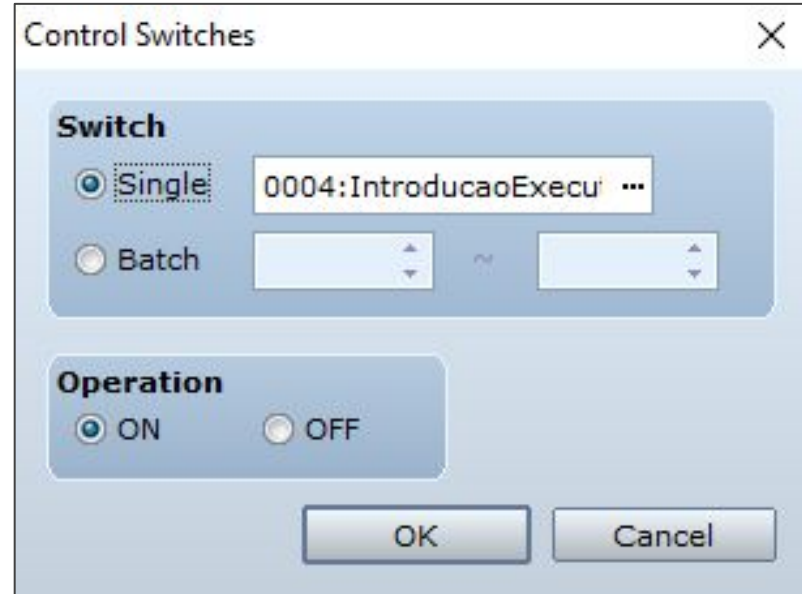
Text: Não me lembro como vim parar aqui.
Isso não é nada bom, acho melhor ir embora logo.

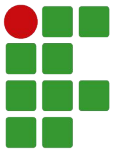




Criando a Abertura do Jogo

- ❏ (cont.):
 - ❏ Configure um interruptor para sabermos que a Introdução já foi executada:
(aba 1) **Control Switches...**
Single: 0004: **IntroducaoExecutada**
Operation: **ON**

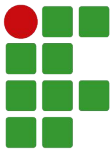




Criando a Abertura do Jogo

- (cont.):
 - Seu evento deve estar assim:

```
Contents:
@>Tint Screen: (-68,-68,0,68), @60, Wait
@>Text: 'Artur', 0, Normal, Bottom
:      : Hã? Onde estou?
:      : E por que minha cabeça está doendo?
@>Set Move Route: Player (Wait)
:      : $>Jump: -1,+1
@>Text: 'Artur', 0, Normal, Bottom
:      : Não me lembro como vim parar aqui.
:      : Isso não é nada bom, acho melhor ir
:      : embora logo.
@>Control Switches: [0004:IntroducaoExecutada] = ON
```

Criando a Abertura do Jogo

- (cont.):
 - E para encerrar esse evento:
Crie nova página (**New Event Page...**)
Defina sua condição como o switch **IntroducaoExecutada**
O trigger deve ser **Autorun**
E execute o comando **Erase Event** (aba 2)
- Agora teste e veja como ficou a abertura do nosso jogo!

Edit Event - ID:003

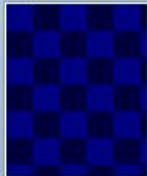
Name:

1 2

Conditions

- Switch is ON
- Switch is ON
- Variable is or above
- Self Switch is ON
- Item exists
- Actor exists

Graphic



Autonomous Movement

Type:

Speed:

Freq:

Options

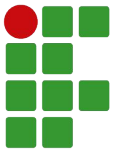
- Walking Anim.
- Stepping Anim.
- Direction Fix
- Through

Priority

Trigger

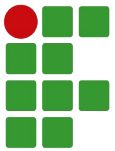
Contents:

-
-



Criando Diálogos com Sara

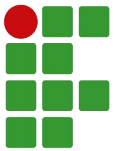
- ❑ Vamos editar o evento da personagem Sara para que o jogador converse com ela ao interagir com a mesma;
- ❑ Para isso, volte para o mapa da **Sala**, clique com o botão direito no evento de **Sara** e escolha **Edit Event...**;
- ❑ Mantenha o trigger como **Action Button** (o jogador precisa apertar o botão para interagir com a mesma).



Criando Diálogos com Sara

- ❑ Na área contents, vamos incluir os seguintes comandos:
 - ❑ Sara diz (**Show Text...**):
Quem é você?
Por que me trouxe aqui?
 - ❑ Artur diz:
Eu sou Artur e não trouxe ninguém.
Eu também não sei como vim para cá.
 - ❑ Configure uma variável (**Control Variables...**) DialogoComSara com um valor aleatório (**Random**) de 1 a 3.

The screenshot shows the 'Control Variables' configuration panel. It is divided into three sections: 'Variable', 'Operation', and 'Operand'.
- **Variable:** The 'Single' radio button is selected, and the text '0002:DialogoComSara ...' is entered in the adjacent text box. The 'Batch' option is unselected.
- **Operation:** The 'Set' radio button is selected, while 'Add', 'Sub', 'Mul', and 'Div' are unselected.
- **Operand:** The 'Random' radio button is selected. The 'Constant' and 'Variable' options are unselected. The 'Random' section has two input fields: the first contains '1' and the second contains '3', with a tilde '~' symbol between them, indicating a range from 1 to 3.



Criando Diálogos com Sara

- ❑ Vamos criar três abas com variações de diálogos;
- ❑ Copie e cole a página atual, configure como condição a variável `DialogoComSara` ser 1 ou maior;
- ❑ Altere o diálogo para Sara dizer:
A única coisa que me lembro é que estava fazendo uma ronda em uma rua tranquila...

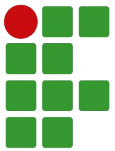
Contents:

```
@>Text: 'Sara', 0, Normal, Bottom
```

```
:      : A única coisa que me lembro é que estava
```

```
:      : fazendo uma ronda em uma rua tranquila...
```

```
@>Control Variables: [0002:DialogoComSara] = Random No. (1...3)
```



Criando Diálogos com Sara

- ❑ Outra vez, copie e cole a página atual, configure como condição a variável `DialogoComSara` ser 2 ou maior;
- ❑ Altere o diálogo para Artur dizer:
`Encontrou algum jeito de sairmos daqui?`
- ❑ Sara responde:
`Nada, ainda.`

Contents:

@>Text: 'Sara', 0, Normal, Bottom

: : Quem é você?

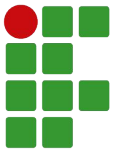
: : Por que me trouxe aqui?

@>Text: 'Artur', 0, Normal, Bottom

: : Eu sou Artur e não trouxe ninguém.

: : Eu também não sei como vim para cá.

@>Control Variables: [0002:DialogoComSara] = Random No. (1...3)



Criando Diálogos com Sara

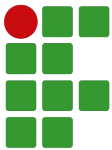
- ❑ (De novo!) Copie e cole a página atual, configure como condição a variável `DialogoComSara` ser 3 ou maior;
- ❑ Altere o diálogo para Sara dizer:
`Quem pode ter feito isso conosco?`
- ❑ Agora teste o jogo e fale com Sara algumas vezes!

Contents:

```
@>Text: 'Sara', 0, Normal, Bottom
```

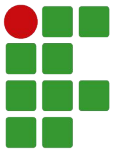
```
:      : Quem pode ter feito isso conosco?
```

```
@>Control Variables: [0002:DialogoComSara] = Random No. (1...3)
```



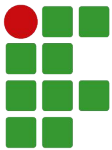
Criando o Encerramento do Jogo

- ❑ Agora, vamos ao diálogo final com Sara! Caso o jogador já tenha movido a estátua que bloqueia a saída, falaremos com Sara e ela se juntará a nós;
- ❑ Para isso, copie e cole uma nova página, configure a condição switch EstatuaMovida = ON e remova os antigos comandos;
- ❑ Sara diz: Como você conseguiu liberar a porta?
- ❑ Artur diz: Foi meio estranho, parecia um enigma...
- ❑ Artur diz: Mas não importa, vamos embora.
- ❑ Sara diz: Certo!



Criando o Encerramento do Jogo

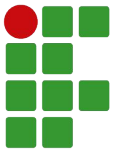
- ❑ Agora, escureça a tela (aba 2, **Fadeout Screen**);
- ❑ E mostre um texto dizendo:
Artur e Sara saem dali imediatamente...
- ❑ E outro texto dizendo:
Mal sabiam eles o que encontrariam fora dali...
- ❑ Aguarde 03 segundos:
aba 2, **Wait...** → **180** frames
- ❑ Voltar para a tela de título
Return to Title Screen



Criando o Encerramento do Jogo

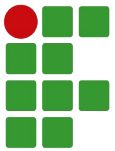
- ❑ O diálogo final ficou assim...

The screenshot shows a game engine interface with a tabbed menu at the top (1-5) and two main panels. The left panel, titled 'Conditions', lists several event triggers: 'Switch' (checked), 'Switch' (unchecked), 'Variable' (unchecked), 'Self Switch' (unchecked), 'Item' (unchecked), and 'Actor' (unchecked). The first 'Switch' condition is set to '0003:EstatuaMovida' and 'is ON'. The right panel, titled 'Contents', lists the actions for this event: '@>Text: 'Sara', 0, Normal, Bottom' with the text ': Como você conseguiu liberar a porta?'; '@>Text: 'Artur', 0, Normal, Bottom' with the text ': Foi meio estranho, parecia um enigma...'; '@>Text: 'Artur', 0, Normal, Bottom' with the text ': Mas não importa, vamos embora.'; '@>Text: 'Sara', 0, Normal, Bottom' with the text ': Certo!'; '@>Fadeout Screen'; '@>Text: -, -, Normal, Bottom' with the text ': Artur e Sara saem dali imediatamente...'; '@>Text: -, -, Normal, Bottom' with the text ': Mal sabiam eles o que encontrariam fora dali...'; '@>Wait: 180 frame(s)'; and '@>Return to Title Screen'.



Dicas Para Que Seu Jogo Se Destaque

- ❑ Não use apenas os recursos padrões do RPG Maker (sprites, tiles, animações e músicas);
- ❑ Variações de cor do ambiente podem criar bons resultados;
- ❑ Use efeitos parallax para conseguir resultados melhores.



Como criar seu primeiro jogo no RPG Maker MV

(nível básico)

— — —

Parte 1 - <https://www.youtube.com/watch?v=7GgBEIHwkv0>

Parte 2 - <https://www.youtube.com/watch?v=GRxe0a071rI>

Parte 3 - <https://www.youtube.com/watch?v=o6PRFPIZcww>

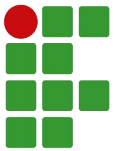
Parte 4 - <https://www.youtube.com/watch?v=gYsoiQxxdac>

Parte 5 - <https://www.youtube.com/watch?v=7v5lQlfSFHM>

Parte 6 - <https://www.youtube.com/watch?v=0dC6LTmwryU>

Parte 7 - <https://www.youtube.com/watch?v=KznggaG1H4A>

Parte 8 - <https://www.youtube.com/watch?v=eb8flYSzsfc>



Canais no YouTube para aprender sobre o RPG Maker

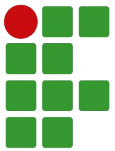
Em português

Honrado - <https://www.youtube.com/c/MuriloMaker>

Canal do Solon - <https://www.youtube.com/c/CanalDoSolon>

Canal Café Gamer - <https://www.youtube.com/c/CanalCaféGamer>

Zona RPG Maker - <https://www.youtube.com/channel/UCJTyyDUubN0wZNFu3mh6LGA>



Canais no YouTube para aprender sobre o RPG Maker

— — —

Em inglês

Driftwood Gaming - <https://www.youtube.com/c/DriftwoodGamingMV>

ToastyTime - <https://www.youtube.com/c/Toastyfunnyguy>

SRDude - <https://www.youtube.com/channel/UCyqXLDebVx9rNnoSpcnNH3Q>

Studio Blue - <https://www.youtube.com/channel/UCpWpYeBtWQAlThRfxkRDBLw>

Lvl Up Design - <https://www.youtube.com/c/LvLUpDesign>

Cazwolf - <https://www.youtube.com/c/Cazwolf>

Yanfly Engine - <https://www.youtube.com/c/YanflyEnginePlugins>

Oath Game - <https://www.youtube.com/c/OathGame>