

## Introdução

O que já vimos?

Para o desenvolvimento de jogos, utilizamos uma série de ferramentas. Vimos as ferramentas do *pipeline* de arte nas unidades anteriores e também os motores de jogo (*game engine*), bem como algumas bibliotecas e SDKs (Software Development Kits). Nosso foco neste material são as IDEs, conhecidas como ambientes de desenvolvimento integrado. Se você quer facilitar o seu trabalho, aprenda a usar uma IDE e verá como é útil para a eficiência na arte de programar.

## IDEs

Para que servem as IDEs?

IDEs são ambientes de desenvolvimento muito úteis para edição de código, para testar e gerar executáveis (*build*). Podemos, ainda, criar projetos, integrar bibliotecas, e sistemas de controle de versão (como o Git).

Elas também colore o código para ajudar na codificação e na visualização dele, identificam erros no código enquanto escrevemos, fornecem-nos o recurso de autocompletar código, como, por exemplo, nomes de variáveis. Ainda existem outras funcionalidades que podem ser adicionadas por meio de *plug-ins*, como suporte a várias linguagens de programação, integração com servidores, acesso a bibliotecas *on-line*, acesso a repositórios de código etc. Essas ferramentas visam a facilitar a vida do programador.

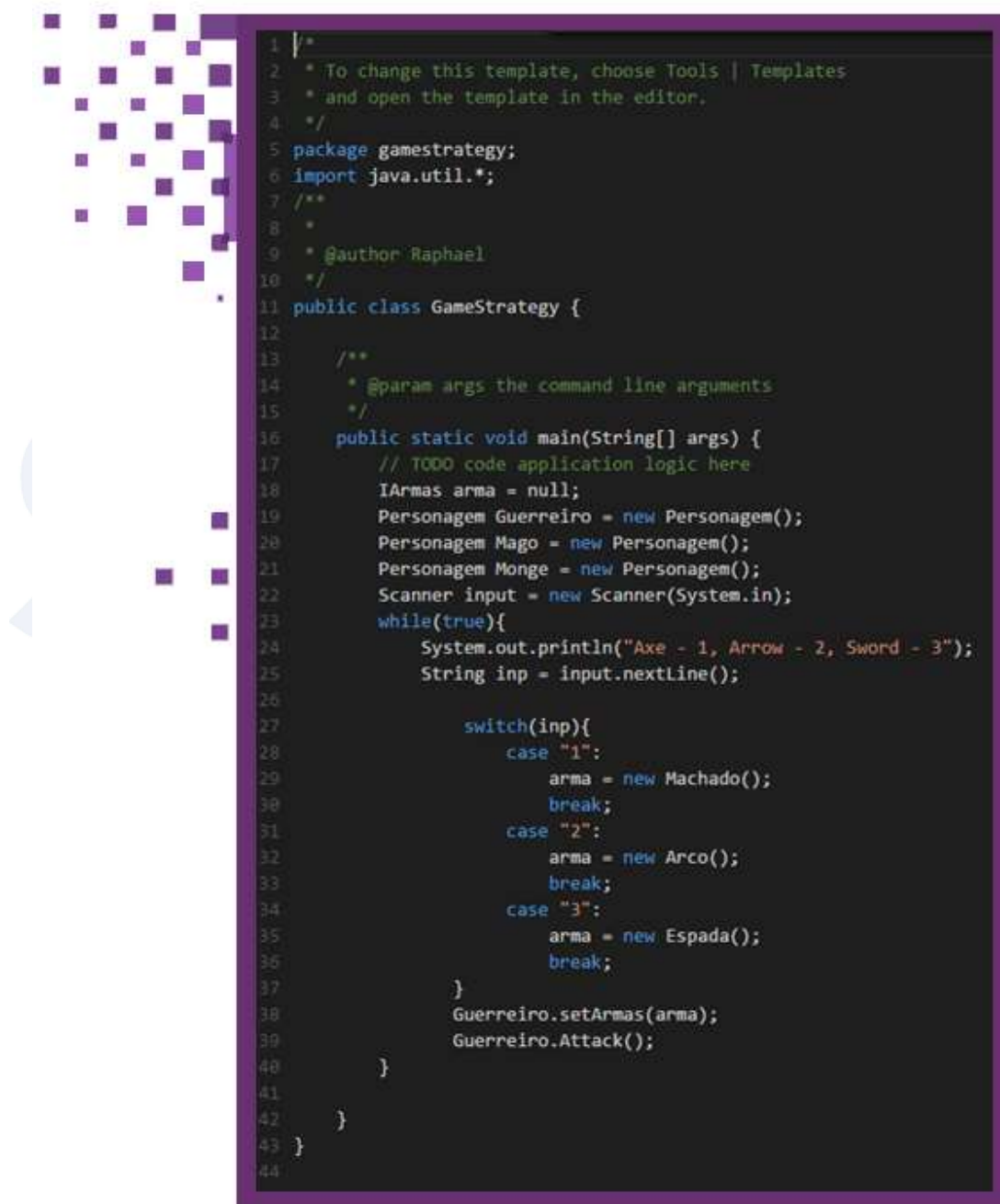


Figura 1 – Imagem de código fonte colorido, identificando comandos, variáveis e valores

Temos uma série de IDEs para desenvolvimento disponíveis gratuitamente e por meio de pagamento. As IDEs pagas normalmente possuem uma versão gratuita, porém com algumas funcionalidades a menos, mas ainda assim com uma série de recursos que facilitam a programação.

**Qual nós vamos utilizar?** Vamos com calma e chegaremos lá. É óbvio que o desenvolvimento de jogos pode ser realizado com diferentes ferramentas, entre elas *engines* e SDKs. Os grandes *players* no mercado de IDEs são a Oracle com a IDE NetBeans, a Microsoft com o Visual Studio, a Zend com a Zend Studio, e o Eclipse (totalmente gratuito de código aberto). Mas qual devemos usar?

Pesquise sobre cada uma destas IDEs para saber seu foco de desenvolvimento e suas características. Procure outras IDEs associadas à linguagem C# e JavaScript.

## Unity 3D e IDEs

A Unity 3D tem todos os benefícios que as IDEs proporcionam? Vamos desvendar esse mistério!

A Unity 3D possui uma IDE própria para programação, o editor MonoDevelop-MD, que pode ser integrado ao Visual Studio-VS, da Microsoft. Quais são os benefícios dessa integração? Por que não utilizar o MonoDevelop? Bom, ambos possuem diversos recursos em comum. Um fator relevante é o valor da licença. Se você já paga a licença da Unity 3D, e o MonoDevelop atende às necessidades do time, não tem sentido pagar a licença do Visual Studio. Apenas se você estiver interessado na série de recursos adicionais para o gerenciamento de equipe do Visual Studio Enterprise ou nos recursos do Community, os quais você já deve ter pesquisado (se não pesquisou ainda, faça isso agora). Você deve estar querendo a resposta para qual escolher, mas a resposta é: experimente.

Mas existe alguma vantagem? Sim, do VS sobre o MD, pois o primeiro é mais estável (falha menos), tem melhor *auto-complete* de código, é atualizado mais frequentemente, pode ser integrado com uma série de ferramentas que a comunidade proporciona. Veja o que a comunidade de desenvolvedores Unity fala sobre a comparação entre os editores e contribua também.

## Integrando a Unity com o VS

Não tem mistério!

Na instalação da Unity 5 já vem o pacote de ferramentas UnityVS do Visual Studio 2015. Basta instalar o VS e, na *engine*, ir em **Edit; Preferences**; selecionar **External Tools**; e, no item **External Script Editor**, selecionar a instalação do Visual Studio.

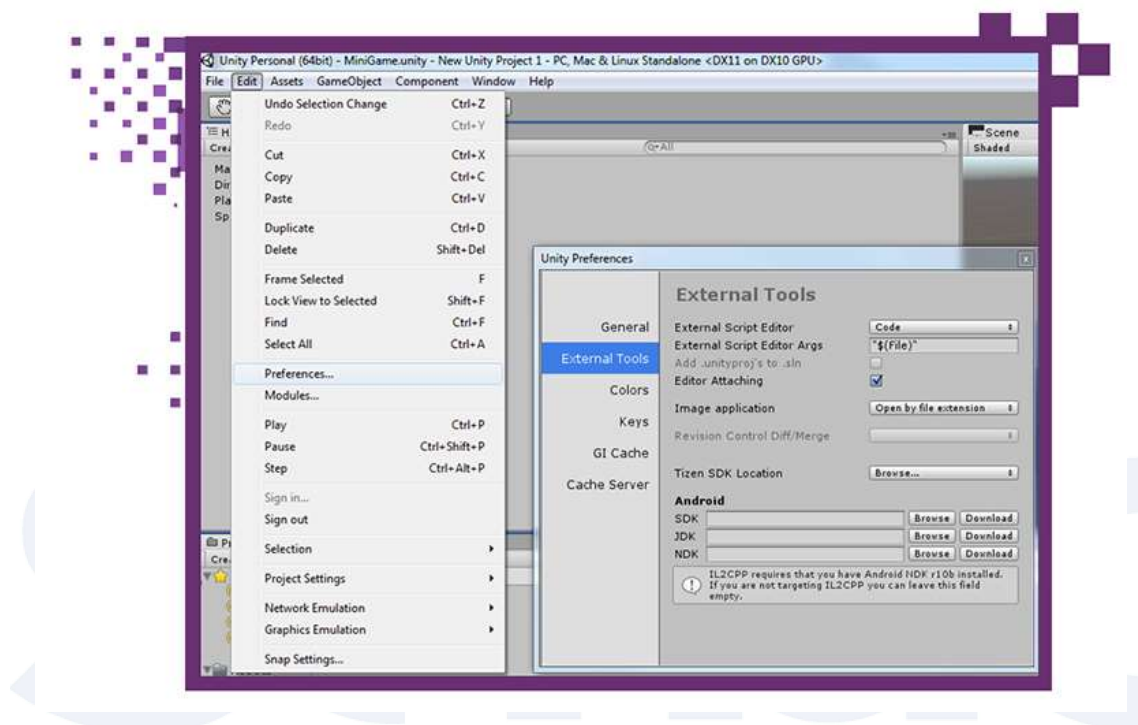
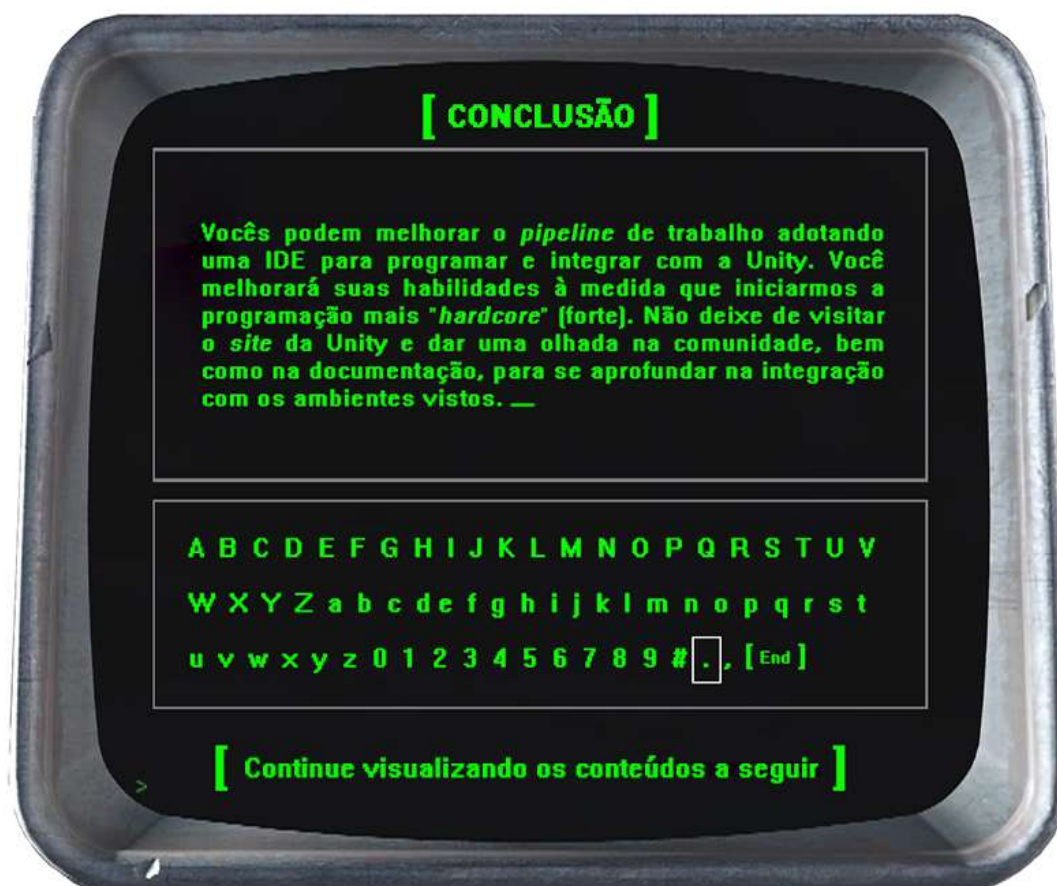


Figura 2 – Imagem do menu **Edit / Preferences** e da tela de preferências ao lado



## Referências

ECLIPSE. Disponível em: <<https://eclipse.org> (<https://eclipse.org>)>. Acesso em: 4 dez. 2015.

NETBEANS. Disponível em: <<https://netbeans.org/kb/index.html> (<https://netbeans.org/kb/index.html>)>. Acesso em: 4 dez. 2015.

UNITY DOCUMENTATION. **Unity Manual**. Visual Studio C# Integration. Disponível em: <<http://docs.unity3d.com/Manual/VisualStudioIntegration.html> (<http://docs.unity3d.com/Manual/VisualStudioIntegration.html>)>. Acesso em: 4 dez. 2015.

UNITY. **Forums**. Disponível em: <<http://forum.unity3d.com/> (<http://forum.unity3d.com/>)>. Acesso em: 4 dez. 2015.

VISUAL STUDIO. Disponível em: <<https://www.visualstudio.com/> (<https://www.visualstudio.com/>)>. Acesso em: 4 dez. 2015.