

Os programas criados em Python são altamente dependentes da indentação encontrada no texto do documento. Um editor de texto que ajude a montar a estrutura necessária pode ser extremamente útil na hora da programação. Venha conferir esse tutorial aonde apresentarei o Atom como IDE para Python. Esse é o IDE usado nos tutoriais sobre orientação ao objeto para Python. Assim, como é o IDE usado nos tutoriais introdutórios sobre programação em Python.

## Lista de tópicos

- [Atom](#)
- [Modificando o Atom para se tornar um IDE para Python](#)
- [Ajustando o editor de texto](#)
- [Key binds](#)
- [Plugin: atom-python-run](#)
- [Plugin: autocomplete-python](#)
- [Plugin: python-tools](#)
- [Plugin: hyperclick](#)
- [Plugin: minimap](#)
- [Plugin: minimap-highlight-selected](#)
- [Plugin: Magic Python](#)
- [Pronto! Você tem um super editor de texto](#)

## Atom

O uso de uma IDE adequada pode aumentar o desempenho dos programadores em qualquer linguagem. O uso de um editor de texto que encontre algumas ferramentas (e.g., auto completar e de navegação pelo documento) melhoram a produtividade dos programadores. O Atom é um editor de texto moderno que pode ser totalmente customizado, podendo ser usado para programar em muitas linguagens de programações diferentes.

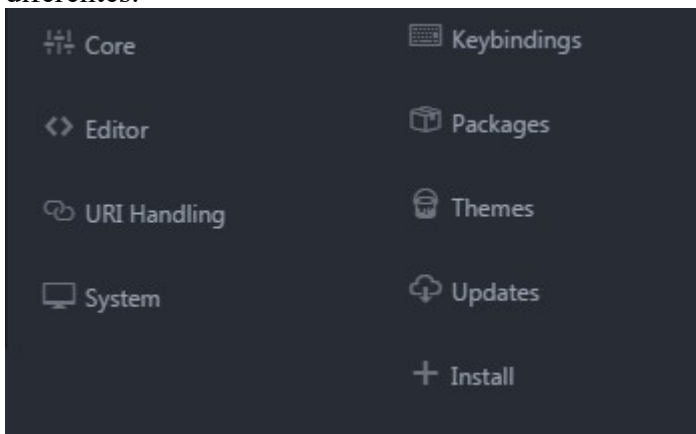


Figura 1. Configurações possíveis do atom: núcleo, editor de texto, teclas de atalhos, temas, pacotes.

Se é a sua primeira vez usando esse editor, ele vai exibir na tela inicial o arquivo de boas vindas e com esse documento você pode chegar no documento de configurações (figura 1).

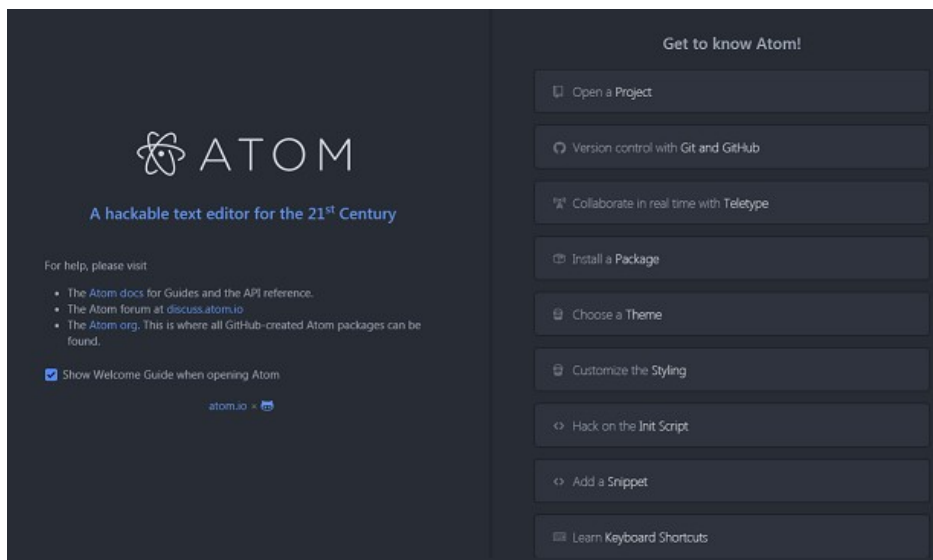


Figura 2. Arquivo de boas vindas do Atom Caso você tenha fechado a aba de boas vindas você sempre conseguirá reabrir o documento de configurações indo em file>settings.

## Modificando o Atom para se tornar um IDE para Python

Vamos configurar o Atom para ser um IDE para Python. O Atom possui milhares de pacotes que podem adicionar novas funcionalidades. Nesse tutorial vamos instalar alguns pacotes que nos interessam e os atalhos de teclado também podem ser customizados. No atom as teclas de atalhos são chamadas de keybinds. Nesse tutorial iremos apresentar alguns atalhos do atom que já vem configurados e alguns que vem na instalação dos novos plugins. O Atom também possui a funcionalidade de modificar o layout do editor de texto. Eu particularmente gosto do layout preto mas você pode configurar como você quiser. Ok! vamos tornar o Atom um IDE para Python.

## Ajustando o editor de texto

A primeira coisa que vamos ajustar é o editor de texto para ser otimizado para Python. Para isso vá até o arquivo de configurações (figura 1) e clique na aba editor. O primeiro ajuste é marcar as caixas: "show line number" e "show indent Guide". Essas duas Configurações servem para deixar mais fácil de visualizar as identações nos arquivos. Após a configuração, coloque o valor 4 no campo Tab length. Essa configuração faz com que cada vez que você aperte a tecla tab, o editor de texto coloque 4 espaços. O valor 4 foi definido no PEP 8. Você pode visualizar essas duas configurações na figura 3.

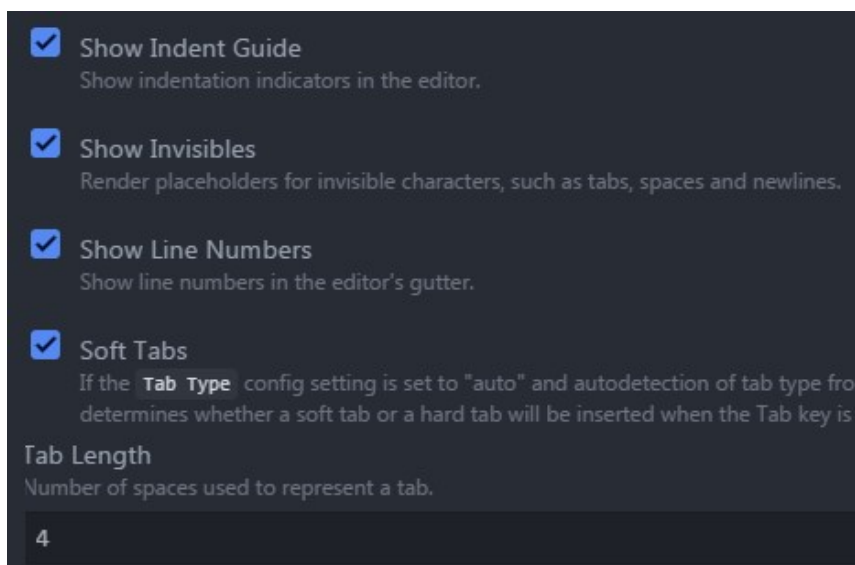


Figura 3. Configuração do editor de texto do atom para melhor programar em python

Após a configuração você vai perceber que agora o editor está mostrando um ponto no lugar de espaço (figura 4b). E possui uma linha guia na indentação do documento (figura 4b)

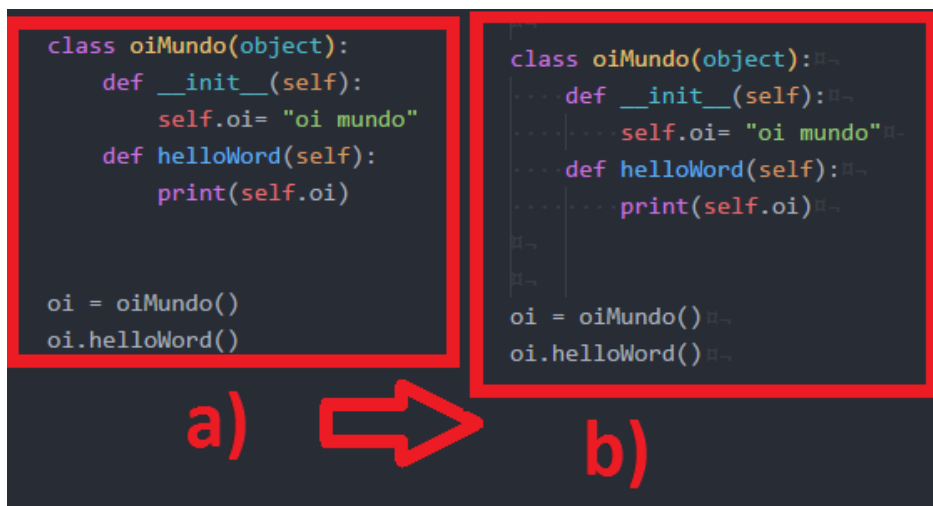


Figura 4. Editor de texto configurado a) sem as configuração explicadas b) editor apos as configurações

## Key binds

O atom possui uma série de atalhos no teclado para as mais diversas funções. Você pode observar tudo isso no documento de configurações (figura 1) na aba keybinds . Alguns atalhos que já vem configurados e podem ser uteis para programadores Python são:

Selecionar o texto + tab : faz com que o texto selecionado avance uma indentação  
 Selecionar o texto + shift-tab : faz com que o texto selecionado retorne uma indentação  
 selecionar o texto + ctrl-/ : faz com que o texto selecionado vire comentário. Caso ele seja um comentario faz ele se tornar código  
 ctrl+s : salva o documento  
 Plugins para ser instalado Agora iremos instalar os plugins que ajudarão ainda mais o seu desenvolvimento em Python. Para instalar basta ir no arquivo de configurações (figura 1) e ir na aba install. Nela digite o nome do plugins indicado nos subtítulos a seguir:

## Plugin: atom-python-run

O plugin atom-python-run serve para rodar os programas em python direto do editor de texto. Para configurar esse plugin vá no documento de configurações (figura 1), vá na aba packages encontre esse plugin e clique no botão setting dele. Clicar nele vai te direcionar para o documento desse plugin e dentro dele coloque no campo “F5 comando” o path de aonde esta instalado o python que você deseja rodar. Geralmente o Python esta instalando em “C:\Users[nome do usuario]\AppData\Local\Programs\Python”.

## Plugin: autocomplete-python

Esse é o plugin de autocompletar do Python. Para a instalação desse plugin vai aparecer as opções entre jedi e keti escolha a opção Jedi:

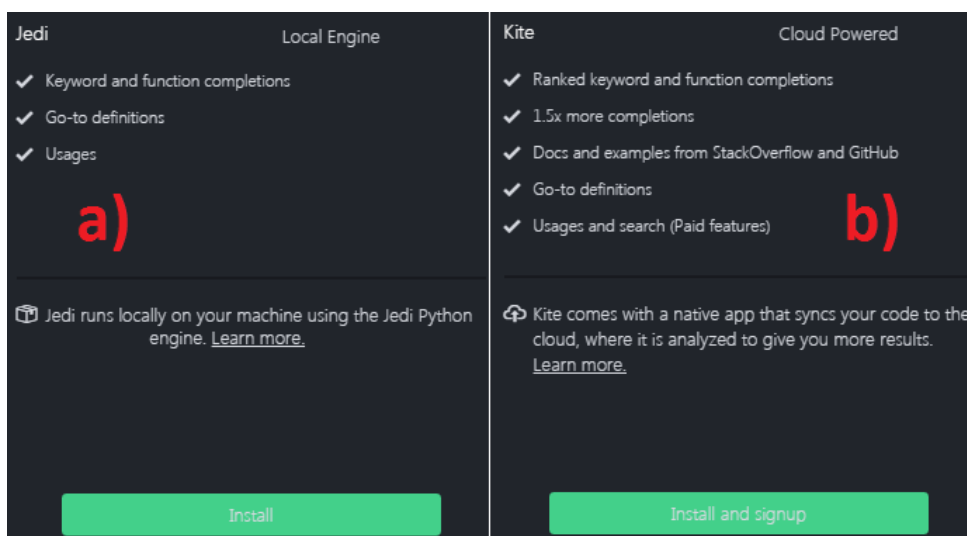


Figura 5. Opções de instalação do plugin autocomplete-python a)jedi b)keti

Após a instalação do plugin de auto completar você irá perceber que vai aparecendo, em um menu próximo ao cursor de escrita, as funções em python que podem ser auto-completadas enquanto você escreve seu código (figura 6). Caso esse menu se feche você também pode usar o atalho “ctrl+espaco” para abri-lo.

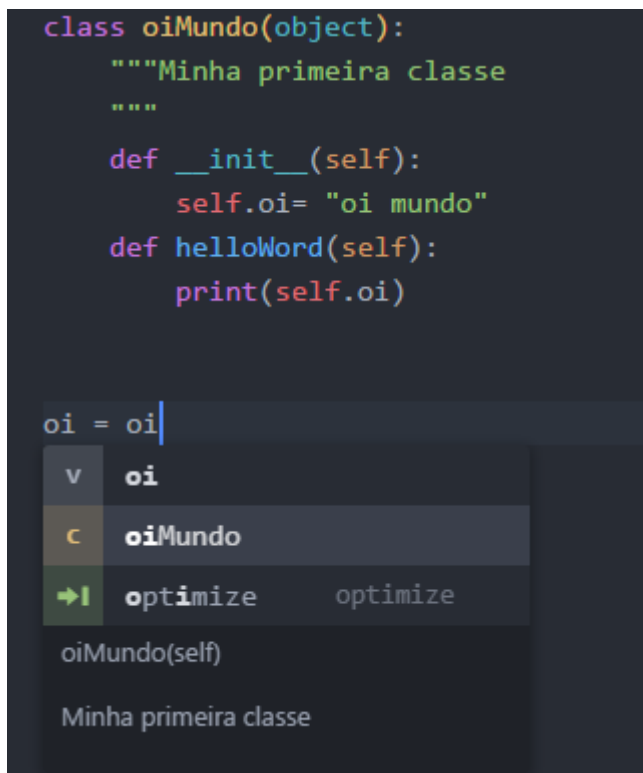


Figura 6. Menu de auto completar código do Atom

## Plugin: python-tools

O plugin python-tools serve para rodar os programas em python serve para movimentar entre funções. Para isso basta estar em um documento “\*.py” com a extensão do python e clicar:

- cursor em cima da função/variável + ctrl-alt-g: faz o cursor ir para a definição da função/variável
- cursor em cima da definição de uma função + ctrl-alt-u : resalta todas as usagens da função escolhida
- selecionar o texto + ctrl+/: faz com que o texto selecionado vire comentário. Caso ele seja um comentário faz ele se tornar código
- ctrl+s : salva o documento
- ctrl+espaco : faz abrir o snipete de auto completar

## Plugin: hyperclick

Ao instalar o plugin hyperclick e já estiver instalado o Plugin:python-tools faz gerar um novo atalho. O novo atalho faz com que você vá para definição da função usando o atalho ctrl-click na função desejada.

## Plugin: minimap

O plugin minimap habilitará no lado direito do documento um minimapa de todo o documento aberto.

## Plugin: minimap-highlight-selected

O plugin minimap-highlight-selected habilita a funcionalidade de realçar o texto no minimapa de qualquer texto que o usuário selecionar. Isso ajuda muito a encontrar variáveis, funções no código.

## Plugin: Magic Python

O plugin language python, que vem instalado, já consegue destacar a sintaxe da linguagem de programação do Python nos arquivos definidos de Python (.py) . Porém, para algumas sintaxes novas da linguagem esse plugin não consegue realçar adequadamente. Em alguns tutoriais aqui do site iremos utilizar essas novas sintaxes. Para então realça-las utilizamos o plugin Magic Python . Para fazer ele funcionar você deve instalar esse package como fez com os outros. Porém, após instalar você vai ter que ir na aba de packages e desabilitar o package language python que vem instalado de fábrica.

## Pronto! Você tem um super editor de texto

Pronto agora você tem um IDE para Python que irá te ajudar a otimizar seu trabalho. Agora é hora de usar esse editor. Caso você seja novo em Python, vá para as aulas introdutórias de Python. Caso você já tenha mais familiaridade vá para os tutoriais de opencv.