

Aula 6

Linguagem de Programação Aplicada

Prof. Renan Portela Jorge

1

Conversa Inicial

2

O que estudaremos?

- Veremos:
 - Ferramentas de suporte ao desenvolvedor
 - Testes unitários (*unit test*)
 - Geração de executável

3

Ferramentas de suporte: *code completion* e *refactorings*

4

Code complete

- Básico
- TAB
- Pós-fixos
- Type-matching*
- F-strings*

5

Code refactoring

- Refatorar menu
- Renomear
- Extrair variáveis
- Refatorar concertos rápidos
- Refatorar no lugar

6

Ferramentas de suporte: *code assistance* e *debug*

Code assistance

- Restaurar e remover código
- Formatar código
- Informação dos Parâmetros
- *Popups* rápidos
- Editor de assistente de código

Debug

- Configurando e entendendo o fluxo do *debug*

Começando com *unit test*

O que é um teste unitário (*unit test*)

- É uma biblioteca de teste unitário incorporada ao Python que fornece uma estrutura para escrever e executar testes automatizados
- Os testes unitários são usados para verificar se partes específicas do código estão funcionando corretamente

Quando usar o *unit test*

- Testar funções e métodos
- Garantir a consistência
- Facilitar a manutenção
- Dar suporte à prática de desenvolvimento orientado a testes (TDD)
- Partes críticas do código que são propensas a erros ou com dependências externas

Quando usar o *unit test*

- Quando se trata de quando usar o *unit test*, é recomendado escrever testes unitários para partes críticas e complexas do código, especialmente aquelas que são propensas a erros ou que possuem dependências externas

13

PyTest

14

PyTest

- É uma biblioteca baseada no *unit test*, mas com uma sintaxe mais simples e mais recursos:
 - Sintaxe simplificada
 - Asserções avançadas
 - Descoberta automática de testes

15

PyTest

- O *pytest* é uma biblioteca de teste que se baseia no *unit test*, mas oferece uma sintaxe mais simples e recursos avançados

16

Gerando executável do Python para Windows

17

Gerando executável

- Para transformar um projeto Python em um executável no Windows, você pode usar ferramentas como PyInstaller ou cx_Freeze
- Ambas empacotam o seu código Python e geram um executável independente que pode ser executado sem a necessidade de ter o Python instalado

18

