

Pygame

Introdução a desenvolvimento de Jogos

Paulo Francisco da Conceição

Instalando o Python – windows

- Baixar o instalador: <https://www.python.org/downloads/>
- Instalar
 - Adicionar no path
- Verificar Instalação
 - `python --version`
 - `pip --version`
- Atualizar o pip
 - `python -m pip install --upgrade pip`

Instalando o PyGame

- Comando
 - `python -m pip install -U pygame --user`
- Verificando instalação
 - `python -m pygame.examples.aliens`

IDEs

- Já vem com IDLE
 - Basta executar
 - Tem limitação de autocompletar o código
- Instalar o Pycharm
 - Mais prático, autocompleta o código
 - <https://www.jetbrains.com/pt-br/pycharm/download/>

Desenvolvendo um jogo

Tela principal

```
import pygame
from sys import exit
pygame.init()
tela = pygame.display.set_mode((800,600))
pygame.display.set_caption("Meu jogo")
clock = pygame.time.Clock()
while True:
    for event in pygame.event.get():
        if event.type == pygame.QUIT:
            pygame.quit()
            exit()
    pygame.display.update()
    clock.tick(60)
```

Pygame

- O relógio
- Superfícies
 - Desenhando superfícies
 - Importando imagens
 - Texto
 - Fonte
- Converter imagens
 - Rapidez em grandes jogos

Retângulos

- Melhorar posicionamento
 - Adaptar o código para os personagens estarem em retângulos
- Detectar colisões
 - `playerRec.colliderect(cc1Rec)`
 - `playerRec.collidepoint((x,y))`
- Atividade com colisão do mouse
 - `Pygame.mouse.get_pos()`
 - `Pygame.mouse.get_pressed()`
- Eventos de mouse
 - `MOUSE MOTION`
 - `MOUSE DOWN`
 - `MOUSE UP`

Desenhos

- Centralizar o texto
 - Colocar em um retângulo
- Colocar margem
 - `pygame.draw.rect(tela, 'Green', textoRec)`

Evento de pulo

- Entrada do teclado
- Pulo e gravidade
- Criar um piso

Evento de teclado

- Direto no loop

```
key=pygame.key.get_pressed()
```

- No evento

```
if event.type == pygame.KEYDOWN:
```

```
    if event.key==pygame.K_SPACE:
```

```
        print('press')
```

Controle de gravidade

- Simular uma trajetória exponencial
 - Ter uma variável que vai aumentando a cada quadro
 - Quando pressionar uma tecla, decrescer a gravidade;
- Verificar o solo
 - Não deixar que o player ultrapasse o solo
 - Só pular quando estiver no solo

Estados do jogo

- Ativo ou inativo
 - Desativar no game over e manter a tela, etc..

Capturar o tempo

- O tempo é iniciado no init
- O tempo atual é obtido de:
 - `pygame.time.get_ticks()`

rotações e redimensionamento

- `playerP = pygame.image.load ('caminho')`
- `playerP = pygame.transform.rotozoom(playerP, 0, 2)`

Timer e eventos personalizados

- Controla o posicionamento aleatório dos inimigos usando evento de cronômetro