

# frogger.py

<https://github.com/joetechem/frogger>

**Pay very close attention to the syntax:**

**()[]{},"':;.**

## Part One: Importing Python Files

```
import pygame,math,sys,os
from pygame.locals import *
from random import randint
```

**\*\* JUMP TO LINE 30 \*\***

## Part Two: Defining the Game Screen Object

```
def game_screen(self):
    pygame.init()

    #screen size and background
    self.size = self.width, self.height = 640,640
    self.bg = pygame.image.load("Sprites/background.png")
    self.go = pygame.image.load("Sprites/gameover.png")
    self.black = 0,0,0

    #initialize text
    self.font = pygame.font.SysFont("monospace",25)

    #variables for object generation
    self.object_list = []
    self.temp_object = []

    #variables for car generation
    self.counter = 0
    self.car_list = []
    self.temp_car = []
```

### **Part Three:** Defining the Frog Start Object and its Lives in a List

```
def frog_start(self):
```

```
    #creating lives image
```

```
    self.lives_list = []
```

```
    self.lives_list.append(Lives(self,560))
```

```
    self.lives_list.append(Lives(self,580))
```

```
    self.lives_list.append(Lives(self,600))
```

**\*\* JUMP TO LINE 323 \*\***

### **Part Four:** Calling the Objects We Defined Earlier

```
def main(self):
```

```
    self.game_screen()
```

```
    self.frog_start()
```

**\*\* JUMP TO LINE 426 \*\***

### **Part Five:** Event Handlers for Car/Frog Collision and Lives Running Out

```
for car in self.car_list:
```

```
    car.tick()
```

```
    self.inbounds(car,self.temp_car)
```

```
    self.player_hit(car)
```

```
    self.player_hit2(car)
```

```
if self.lives == 0:
```

```
    self.frog_restart()
```

```
    continue
```