



## Тема 25. Подсчет очков. Подключение музыки

### 1. Вывод текста на экран

Класс **Font** предназначен для работы со шрифтами и текстом. Конструктору Font надо передавать имя файла шрифта и размер шрифта. В pygame есть шрифт по-умолчанию. Чтобы использовать его, вместо имени файла в конструктор надо передать объект

**None:**

```
score=pygame.font.Font(None,46)
```

Метод **render()** создает поверхность (экземпляр Surface), на которой "написан" переданный в качестве аргумента текст, шрифтом, к которому применяется метод. Вторым аргументом указывается сглаживание (0 – нет, 1 – есть), третьим – цвет текста.

```
text_score=score.render("text",1,(255,255,255))
```

В метод **blit** необходимо передать имя переменной с «текстом» и координаты размещения текста. Координаты передаются в качестве кортежа (в дополнительных скобках).

```
sc.blit(text_score,(20,30))
```

**Не забудьте переводить числа в строки ( str() ) в методе render().**

### 2. Работа с аудио

В pygame для работы с аудио предназначены модули **pygame.mixer** и **pygame.mixer.music**. Модули похожи, однако **pygame.mixer** в первую очередь адаптирован для добавления и настройки звуковых эффектов в игре. В то время как **pygame.mixer.music** – для добавления фоновой музыки.

Для того, чтобы добавить **фоновую музыку**, необходимо подключить файл и запустить его:

```
pygame.mixer.music.load('fon.mp3')  
pygame.mixer.music.play()
```

Для того, чтобы добавить **звук спецэффекта**, необходимо подключить файл и запустить его в определенный момент игры:

```
sound_udar = pygame.mixer.Sound('udar.wav')
```

```
if pygame.sprite.spritecollideany(car1, cars):  
    print("Авария!!!")  
    sound_udar.play()  
    game_over=0  
    menu_def()
```



