Project 2 Solar System

20191177 Hwang Minho

```
class Sun(pygame.sprite.Sprite):
    def __init__(self):
        pygame.sprite.Sprite.__init__(self)
        self.image0 = sun
        self.image0.set_colorkey(BLACK)  #검은색 투명하게
        self.image = self.image0.copy()
        self.rect = self.image.get_rect()
        self.rect.center = [WIDTH / 2, HEIGHT / 2]
        self.speedx = 0
        self.rot = 0
        self.rot_speed = 1
        self.last_update = pygame.time.get_ticks()

    def update(self):
        now = pygame.time.get_ticks()
        if now - self.last_update > 100:
            self.last_update = now
            self.rot = (self.rot + self.rot_speed) % 360
            new_image = pygame.transform.rotate(self.image0, self.rot)

            old_center = self.rect.center
            self.image = new_image
            self.rect = self.image.get_rect()
            self.rect.center = old_center
```

I made class Sun. Sun rotates by itself.

```
class Earth(pygame.sprite.Sprite):
    def __init__(self):
        pygame.sprite.Sprite.__init__(self)
        self.image0 = earth
        self.image0.set_colorkey(BLACK)
        self.image = self.image0.copy()
        self.rect = self.image.get_rect()
        self.rect.center = (600, 100)
        self.speedx = 0
        self.rot = 0
        self.rot_speed = 1
        self.last_update = pygame.time.get_ticks()

    def update(self):
        now = pygame.time.get_ticks()
        if now - self.last_update > 2:
```

```
            self.last_update = now
            self.rot = (self.rot + self.rot_speed) % 360
            new_image = pygame.transform.rotate(self.image0, self.rot)

            old_center = self.rect.center
            self.image = new_image
            self.rect = self.image.get_rect()
            self.rect.center = old_center

            global degree
            radian = np.deg2rad(degree)
            c = np.cos(radian)
            s = np.sin(radian)
            self.cx=600 -35
            self.cy=350 +35
            x=self.rect.center[0] -self.cx
            y=self.rect.center[1] -self.cy
            self.rect.center = [x*c-y*s + self.cx, x*s+y*c + self.cy]

    def center(self):
        return self.rect.center
```

And I made class Earth also. The class Earth inherited the class Sun, but in def update(self), I did method overriding to make earth run its orbit. Along the way, I used rotation transformation. And I made def center to make moon.

```
class Moon(Earth):
    def __init__(self):
        super().__init__()
        self.image0 = moon
        self.rect.center = (600, 0)

    def update(self):
        super().update()
        global degreem
        radian = np.deg2rad(degreem)
        c = np.cos(radian)
        s = np.sin(radian)
        self.cx,self.cy = earthsp.center()
        self.cx-=15
        self.cy+=15
        x=self.rect.center[0] -self.cx
        y=self.rect.center[1] -self.cy
        self.rect.center = [x*c-y*s + self.cx, x*s+y*c + self.cy]
```

The class Moon inherits the class Earth, not the class Sun. It's because moon should move like earth for sun. And I also made the class Venus, Saturn and Titan like this.

```
img_dir = path.join(path.dirname(__file__), 'image')
sun = pygame.image.load(path.join(img_dir, "sun.png")).convert()
earth=pygame.image.load(path.join(img_dir, "earth.png")).convert()
moon=pygame.image.load(path.join(img_dir, "moon.png")).convert()
venus=pygame.image.load(path.join(img_dir, "venus.png")).convert()
saturn=pygame.image.load(path.join(img_dir, "saturn.png")).convert()
titan = pygame.image.load(path.join(img_dir, "titan.png")).convert()
galaxy= pygame.image.load(path.join(img_dir, "galaxy.png")).convert()

all_sprites = pygame.sprite.Group()
sunsp = Sun()
earthsp=Earth()
moonsp=Moon()
venussp=Venus()
saturnsp=Saturn()
titansp=Titan()

all_sprites.add(sunsp)
all_sprites.add(earthsp)
all_sprites.add(moonsp)
all_sprites.add(venussp)
all_sprites.add(saturnsp)
all_sprites.add(titansp)
```

I loaded images, made sprite group, and added the planets to a sprite group.

```
 all_sprites.update()

    # Draw / render
    screen.fill(BLACK)
    for i in range(100):
        x=random.randint(0,1200)
        y=random.randint(0,700)
        pygame.draw.circle(screen, WHITE, (x,y), 3)
    all_sprites.draw(screen)
```

Lastly, I updated all sprites, made a hundred small stars, and draw the planets

This is my GitHub repositories address : https://github.com/Minho3256.