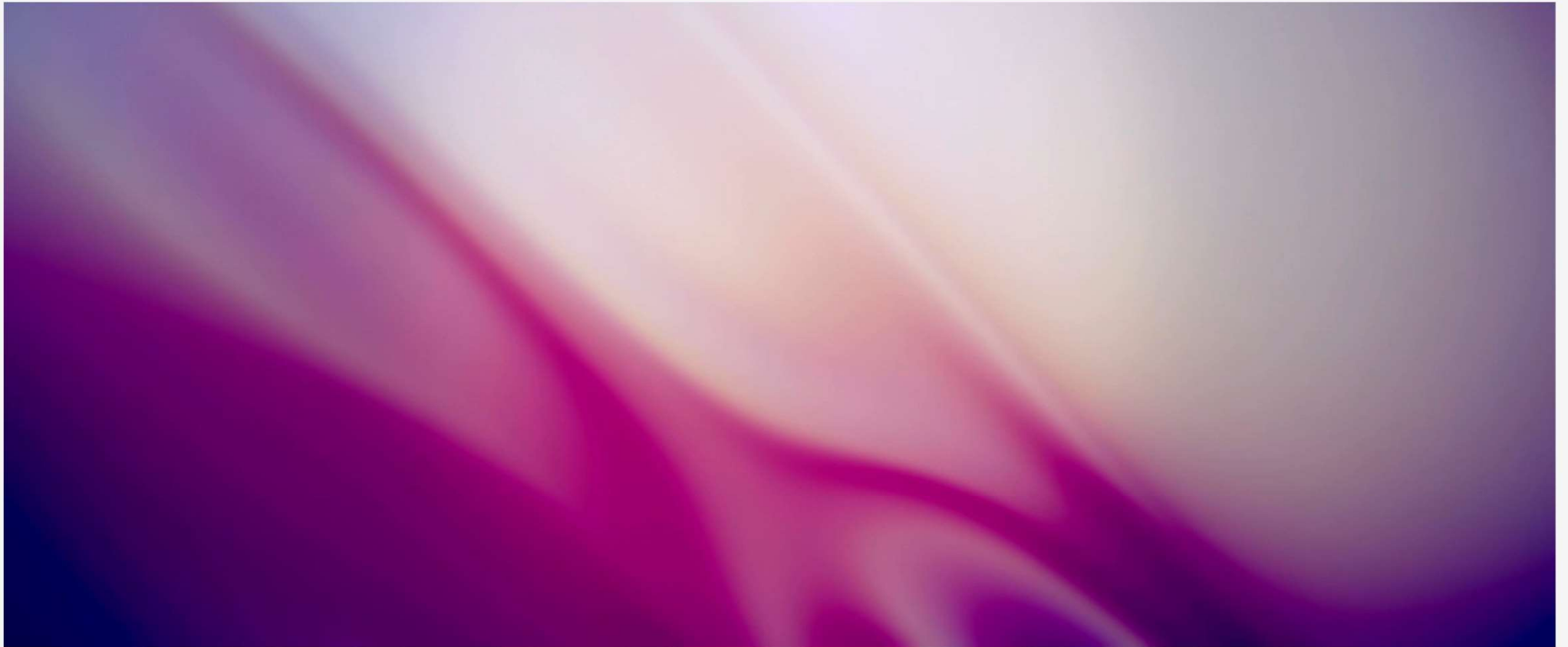


## Day – 7 Loading Backgrounds and Images

---



# Loading Background Image

---



**bgpic(picname=None):** Sets background image or returns name of current background image.



picname -- a string, name of a gif-file or "nopic".

If picname is a filename, set the corresponding image as background.

If picname is "nopic", delete background image, if present.

If picname is None, return the filename of the current background image.



```
>>> turtle.bgpic()
```

```
'nopic'
```

```
>>> turtle.bgpic("landscape.gif")
```

```
>>> turtle.bgpic()
```

```
'landscape.gif'
```

## Turtle Shape with custom images

---

✓ **screen.addshape(name, shape=None):** Adds a turtle shape to TurtleScreen's shapelist.

✓ Arguments:

- name is the name of a gif-file and shape is None. Installs the corresponding image shape.

!! Image-shapes DO NOT rotate when turning the turtle,  
!! so they do not display the heading of the turtle!

- name is an arbitrary string and shape is a tuple of pairs of coordinates. Installs the corresponding polygon shape
- name is an arbitrary string and shape is a (compound) Shape object. Installs the corresponding compound shape. To use a shape, you have to issue the command `shape(shapename)`.

call: `register_shape("turtle.gif")`

--or: `register_shape("tri", ((0,0), (10,10), (-10,10)))`

Example:

```
>>> turtle.register_shape("triangle", ((5,-3),(0,5),(-5,-3)))
```

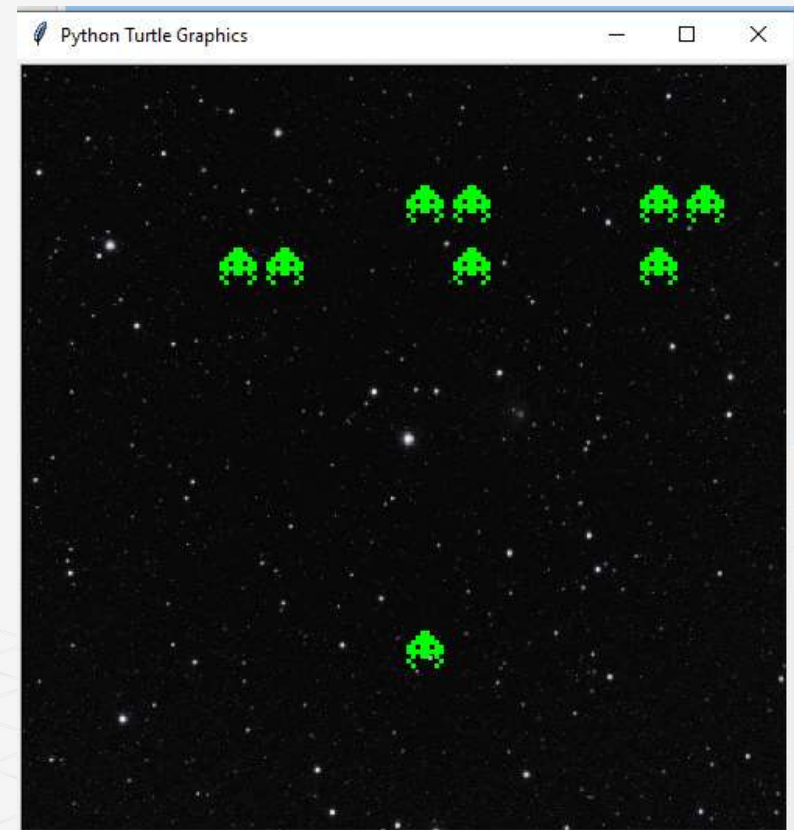
# Load Invaders Randomly

```
import turtle
import random

enemies = []

turtle.setup(500,500)
turtle.bgpic("bg.gif")
turtle.register_shape("invader.gif")

for i in range(10):
    enemies.append(turtle.Turtle())
for enemy in enemies:
    x = random.randint(-200, 200)
    y = random.randint(0, 200)
    enemy.penup()
    enemy.setposition(x, y)
    enemy.shape("invader.gif")
```



# What You Learned

---

✓ At this point you should

– Understand

- Loading background images
- Customized shapes for turtle

# Lab Time

---

