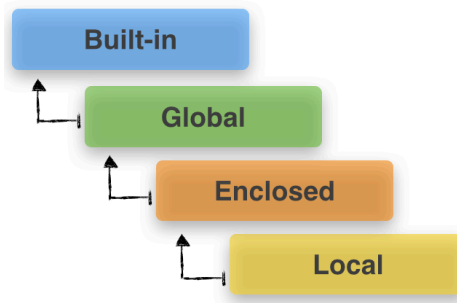- Readings: *Liang* section 6.9 – 6.10

Scope of variables in Python program. Think `LEGB`:

Graphic Source: Sebastian Raschka, *A Beginner's Guide To Python*[1]

- Local: Variables declared locally (within) a function are only visible to that function.

```
def f():
    s = "Beat Army!"
    print(s)

s = "Go Navy!"
f()
print(s)
```

- Enclosed: Variable scope within nested blocks of code is determined by starting at the inner-most block and working outward. What does this print?

- Global: Variables declared (or imported) at the top-level of a file are visible by any code, including nested code, within that file.

- Builtin: Python names pre-assigned in built-in modules : `open`, `range`, `SyntaxError`, ...

There ways to override `LEGB` behaviors by using the `nonlocal` and `global` keywords in front of variables we declare, but we'll see examples of that later.

---

[1] https://sebastianraschka.com/Articles/2014_python_scope_and_namespaces.html