### Static vs Dynamic Typing:

- **Dynamic Typing (Python)**: Types change during execution.
- **Static Typing**: Fixed types before runtime.

### Static vs Dynamic Binding:

- **Static Binding**: Types fixed at compile-time.
- **Dynamic Binding**: Types associated at runtime, changeable during execution.

### Stylish Declaration Techniques:

- Clear, efficient variable/structure declarations in Python.
- Use Python's flexibility for concise, readable code.

In [1]:
```
# c
# int a = 5;
```

In [1]:
```
place = "Kolkata"
```

In [2]:
```
print(place)
```

```
Kolkata
```

In [4]:
```
number = 1.5
print(number)
```

```
1.5
```

### No Var Decl in Python

- Use vars **w/o prior decl**.
- Assign val directly; Python auto-creates.

In [ ]:
```
# Dynamic Typing (Python)
a = 5;

# Static Typing (Java)
int a = 5;
```

In [6]:
```
print(place)
```

```
Kolkata
```

In [7]:
```
name = "Rutik"
print(name)
type(name)
```

```
Rutik
```

Out[7]:
```
str
```

In [8]:

```
name = 2.5
print(name)
type(name)
```

```
2.5
```

Out[8]:

```
float
```

In [9]:

```
# Dynamic Binding Example
a = 5.6         # Assign integer
print(a)
print(type(a))

a = "Hrishikesh"   # Reassign string
print(a)
print(type(a))
```

```
5.6
<class 'float'>
Hrishikesh
<class 'str'>
```

In [18]:

```
# Static Binding Example
int a = 5
```

```
  Cell In[18], line 2
    a = 5 b =2
          ^
SyntaxError: invalid syntax
```

In [6]:

```
# Special Syntax
#declaration techniques
a = 1; b = 2; c = 3  # multiple variable in single line
print(a)
print(b)
print(c)
```

```
1
2
3
```

In [7]:

```
 a, b, c = 1, 2, 3    #

print(a)
print(b)
print(c)
```

```
1
2
3
```

In [27]:

```
a = b = c = 6

print(a, b, c)
```

```
6 6 6
```

In [17]:

```
num_1 = 20
num_2 = 30

print("Before Swaping: ", num_1, num_2)
```

```
Before Swaping:  20 30
```

In [24]:

```
num_1,num_2 = num_2,num_1

print("After Swaping: ", num_1,num_2)
```

```
After Swaping:  30 20
```

In [27]:

```
temp = num_1
num_1 = num_2
num_2 = temp
print("After Swaping: ", num_1,num_2)
```

```
After Swaping:  30 20
```

In [3]:

```
print("Ali")
```

```
Ali
```

In [11]:

```
first_name = 1
```

In [10]:

```
print(first_name)
type(first_name)
```

```
Ali
```

Out[10]:

```
str
```

In [5]:

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[5], line 1
----> 1 k

NameError: name 'k' is not defined
```

In [ ]: