Ch2: Understand Python

305172 Computer Programming Laboratory

Jiraporn Pooksook
Naresuan University

How Python works

- It runs line by line from the first line to the end. If there is a function call, it jumps to run the function definition. After finishing the code in function, it comes back to the point where the function call is.

Try this code:

```
print ("What \'s up 305172")

def functionA():
    x = input('Enter your name: ')
    print(x)

print("before function")
functionA()
print("end program")
```

Python vs C (Compile)

```python
print ("What \'s up 305172")

def functionA():
    x = input('Enter your name: ')
    print(x)

print("before function")
functionA()
print("end program")
```

Run as this sequence
Line 1, 7, 8, 3 – 5, 8, 9

```
What 's up 305172
before function
Enter your name mod
mod
end program
```

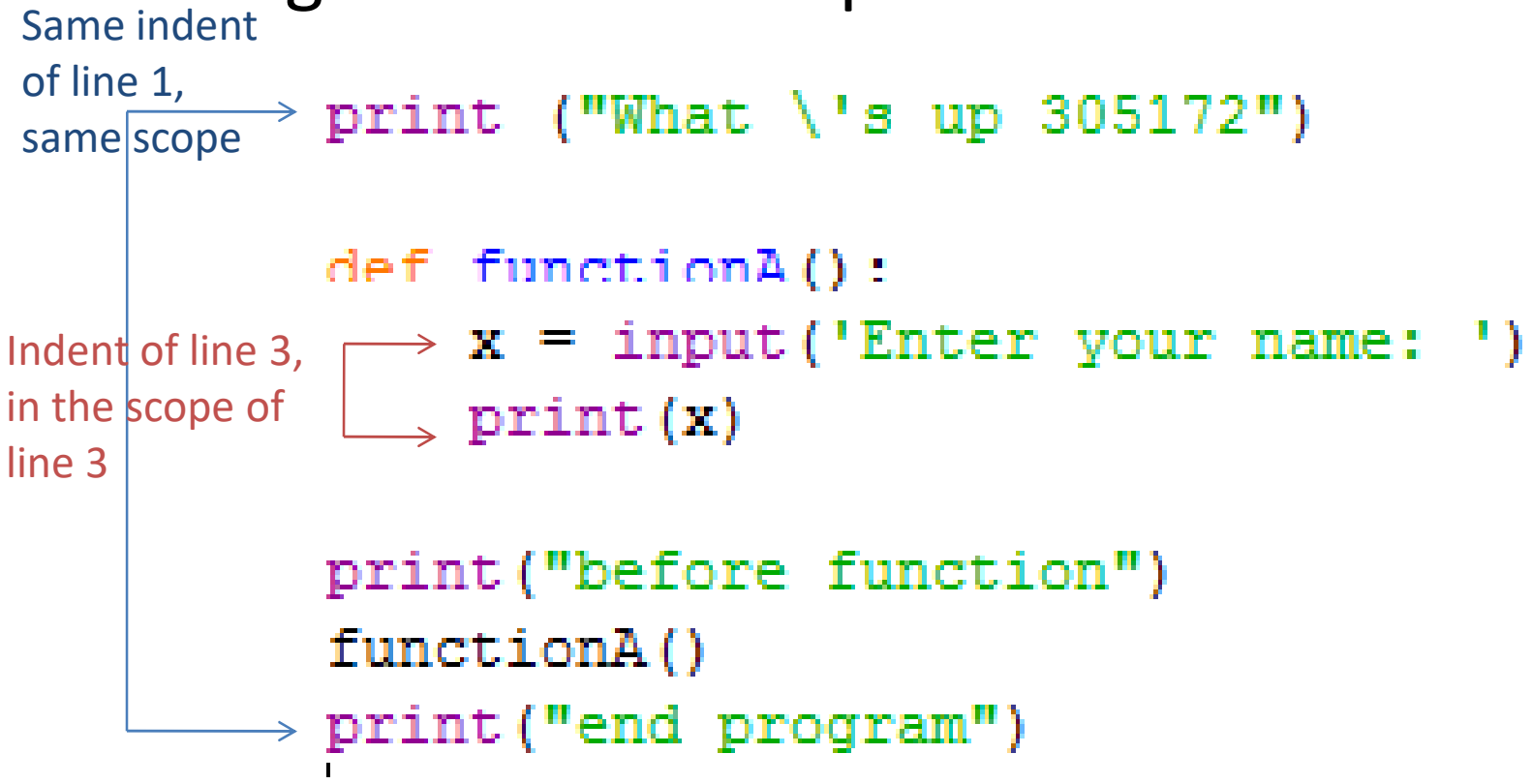
```c
12 void hello()
13 {
14     printf("Hello World \n");
15 }
16 int main()
17 {
18     printf("What\'s up 305172 \n");
19     printf("before function\n");
20     hello();
21     printf("end program\n");
22     return 0;
23 }
```

Run as this sequence
Line 16 – 19, 20, 12 – 15, 20, 21-23

```
What's up 305172
before function
Hello World
end program
```

Scope of program

- Using indent as a scope.



Python vs C (Scope)

- Using Indent

```
print ("What \'s up 305172")

def functionA():
    x = input('Enter your name: ')
    print(x)

print("before function")
functionA()
print("end program")
```

This indent is similar to having { } as in code of C line 13-15

- Using curly brackets

```
12  void hello()
13  {
14      printf("Hello World \n");
15  }
16  int main()
17  {
18      printf("What\'s up 305172 \n");
19      printf("before function\n");
20      hello();
21      printf("end program\n");
22      return 0;
23  }
```

Python vs C (Variables Type)

```python
print ("What \'s up 305172")

def functionA():
    x = input('Enter your name: ')
    print(x)

print("before function")
functionA()
print("end program")
```

Declare variable without type

```c
void hello()
{
    char name[100];
    printf("Enter your name: ");
    scanf("%s",&name);
    printf("%s\n",name);
}
int main()
{
    printf("What\'s up 305172 \n");
    printf("before function\n");
    hello();
    printf("end program\n");
    return 0;
}
```

Static type for variable declaration

Python vs C (in general)

Python	C
• Compiler • Dynamic type • Using indent to define scope of program • Each instruction ends with new line	• Interpreter • Static type • Using { } to define scope of program • Each instruction ends with ;

Exercise: function call

```python
print ("What \'s up 305172")

def functionA():
    x = input('Enter your name: ')
    print(x)

print("before function")
functionA()
print("end program")
```

Delete this line then run

```c
void hello()
{
    char name[100];
    printf("Enter your name: ");
    scanf("%s",&name);
    printf("%s\n",name);
}
int main()
{
    printf("What\'s up 305172 \n");
    printf("before function\n");
    hello();
    printf("end program\n");
    return 0;
}
```

Delete this line then run

Exercise: multiple function call

```python
print ("What \'s up 305172")

def functionA():
    x = input('Enter your name: ')
    print(x)

print("before function")
functionA()
print("end program")
```

Duplicate this line then run

```c
void hello()
{
    char name[100];
    printf("Enter your name: ");
    scanf("%s",&name);
    printf("%s\n",name);
}
int main()
{
    printf("What\'s up 305172 \n");
    printf("before function\n");
    hello();
    printf("end program\n");
    return 0;
}
```

Duplicate this line then run