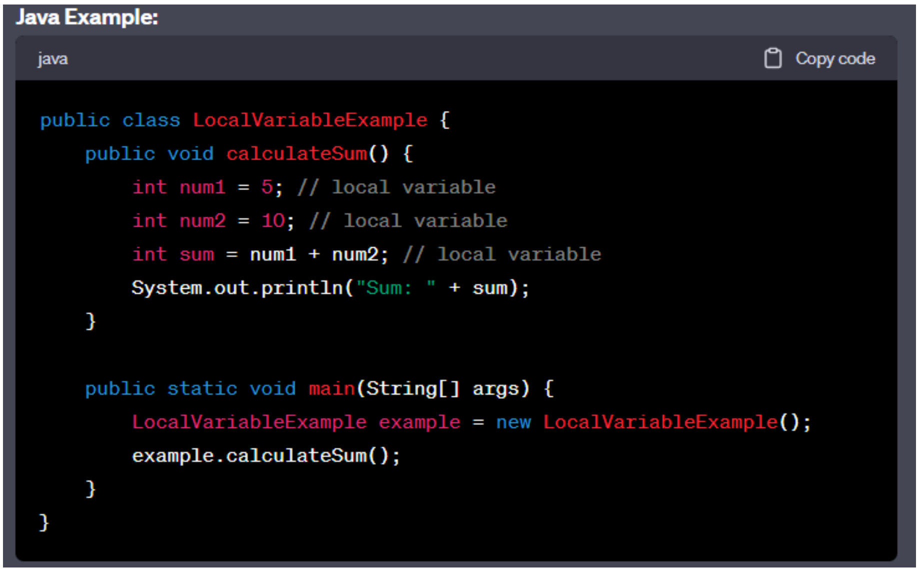
Local and instance variable

Local and instance variables are two types of variables used in programming languages to store and manipulate data. Here's an explanation of each type with examples in both Java and Python:

Local Variables:

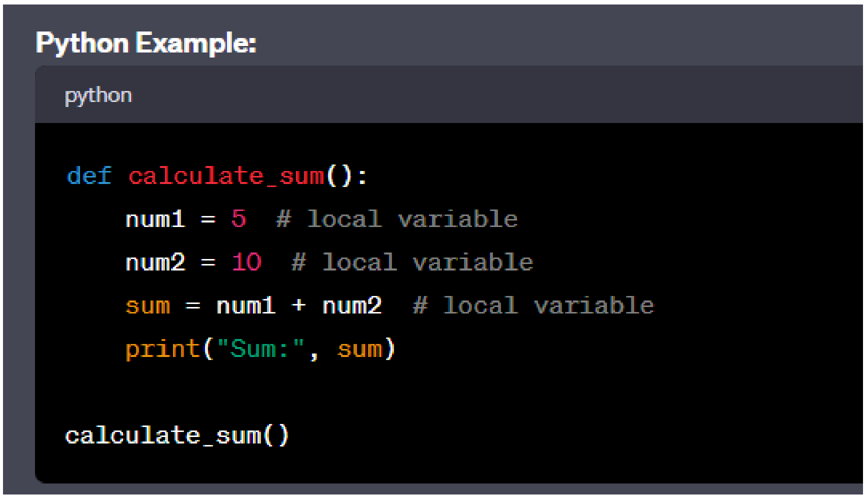
- Definition: Local variables are variables declared and defined within a specific scope, such as within a function, method, or block of code. They have limited visibility and can only be accessed within the scope in which they are declared.
- Lifetime: Local variables exist as long as the scope in which they are defined is active. Once the scope is exited, the local variables are destroyed.
- Usage: Local variables are typically used for temporary storage within a specific context, and their values are not accessible outside that context.



Java Example:

```java
public class LocalVariableExample {
    public void calculateSum() {
        int num1 = 5; // local variable
        int num2 = 10; // local variable
        int sum = num1 + num2; // local variable
        System.out.println("Sum: " + sum);
    }

    public static void main(String[] args) {
        LocalVariableExample example = new LocalVariableExample();
        example.calculateSum();
    }
}
```



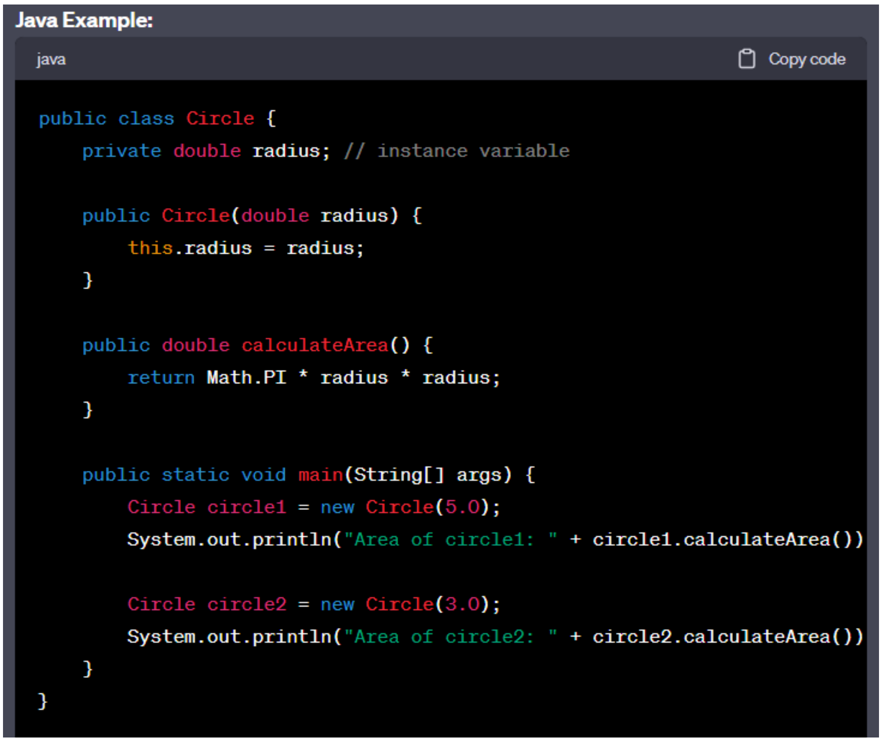
Python Example:

```python
def calculate_sum():
    num1 = 5  # local variable
    num2 = 10  # local variable
    sum = num1 + num2  # local variable
    print("Sum:", sum)

calculate_sum()
```

Instance Variable:

- Definition: Instance variables are variables declared within a class but outside of any method or constructor. They are associated with instances (objects) of the class and have different values for each object. Instance variables define the attributes or properties of an object.
- Lifetime: Instance variables are created when an object of the class is instantiated and exist as long as the object exists.
- Usage: Instance variables store data specific to each object of a class. They represent the state or characteristics of the objects.



Java Example:

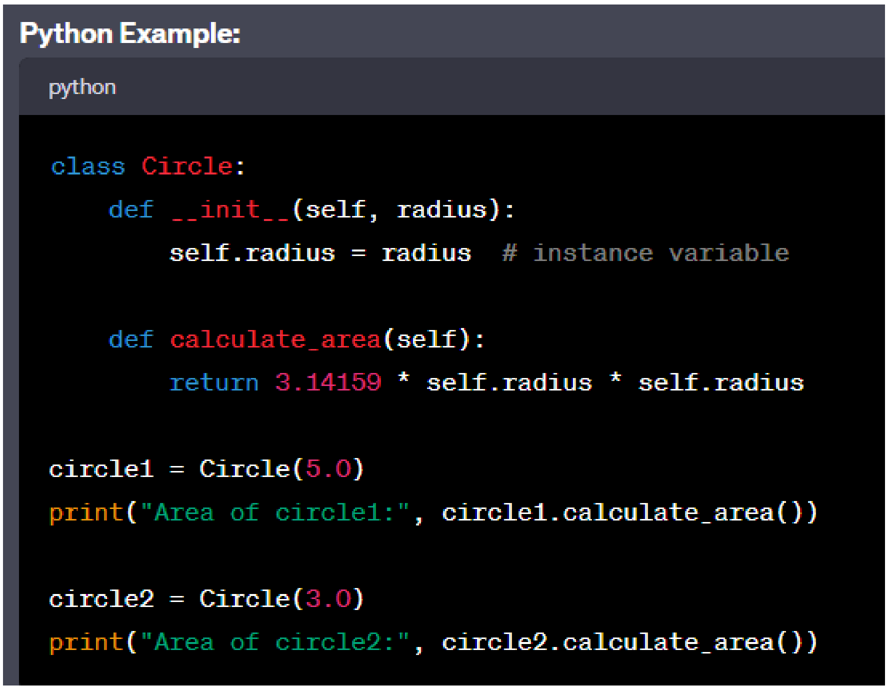
```java
public class Circle {
    private double radius; // instance variable

    public Circle(double radius) {
        this.radius = radius;
    }

    public double calculateArea() {
        return Math.PI * radius * radius;
    }

    public static void main(String[] args) {
        Circle circle1 = new Circle(5.0);
        System.out.println("Area of circle1: " + circle1.calculateArea())

        Circle circle2 = new Circle(3.0);
        System.out.println("Area of circle2: " + circle2.calculateArea())
    }
}
```



Python Example:

```python
class Circle:
    def __init__(self, radius):
        self.radius = radius  # instance variable

    def calculate_area(self):
        return 3.14159 * self.radius * self.radius

circle1 = Circle(5.0)
print("Area of circle1:", circle1.calculate_area())

circle2 = Circle(3.0)
print("Area of circle2:", circle2.calculate_area())
```

In both examples, local variables are used within a method or function and have a limited scope. On the other hand, instance variables are declared within a class and have different values for each instance (object) of the class.