(Program on SLC and GLC where the return type of the method is String :

```
Circle.java(SLC)

package com.rvci.blc;

//SLC
public class Circle
{
    public static String areaOfCircle(double radius)
    {
        if(radius<0)
        {
            System.err.println("Error : radius cannot be 0 OR Negative ");
            System.exit(0);
        }
        final double PI = 3.14;
        double area = PI * radius * radius;
        return ""+area;
    }
}

DemoOfCircle.java

package com.rvci.glc;

import com.rvci.blc.Circle;

//GLC
public class DemoOfCircle
{
    void main()
    {
        IO.println("Enter the radius :");
        double radius = Double.parseDouble(IO.readln());

        String area = Circle.areaOfCircle(radius);

        IO.println("Area of the Circle is :"+area);
    }
}
```

One more program on String return type :

```
//SLC  Employee.java
public class Employee
{
    public static String getEmployeeDetails(int ID, String name, double salary)
    {
        return "[Employee Id is :"+ID+", Name Is :"+name+", Salary Is :"+salary+"]";
    }
}

package com.rvci.glc;

import com.rvci.blc.Employee;

//GLC  EmployeeDetails.java
public class EmployeeDetails
{
    void main()
    {
        IO.print("Enter Employee Id :");
        var id = Integer.parseInt(IO.readln());

        IO.print("Enter Employee Name :");
        var name = IO.readln();

        IO.print("Enter Employee Salary :");
        var salary = Double.parseDouble(IO.readln());

        String employeeInfo = Employee.getEmployeeDetails(id, name, salary);
        IO.println("Employee Details :");
        IO.println(employeeInfo);
    }
}
```

Types of Variable in JAVA :
----
* Based on the **data type**, We have only **2 types of variable in java** :

    1) Primitive Variable
    2) Reference Variable

Primitive Variable :

* If we **declare** any variable with **primitive data type** like byte, short, int, long, float, double, char and boolean then it is called Primitive Variable.
    Example :
    int x = 10; //Primitive Variable

* On primitive Variable, We cannot assign null literal as well as we cannot call any method by using primitive variable.
    Example :

    int y = null; //Invalid

    int z = 23;
    z.m1(); //Invalid (We cannot call any method)

Reference Variable :

If a variable is declared with reference data type then it is called reference variable.

Example :
    Student s; //s is reference variable
    Employee emp = new Employee(); //emp is reference variable
    Integer y = null; //y is reference variable

On reference Variable we can assign null as well as we can call methods.

    String s = new String("Hyderabad");
    s.length(); //Valid
    s = null; //Valid

* Now, based on the **declaration position, Variables are further classified into 4 categories :**

```
                        Variables
                       /         \
                      /           \
              Class Level      Method Level
              /      \          /       \
             /        \        /         \
        static field  Non static field  Local    Parameter
                                        Variable  Variable
```

Classification of Variables :
----
1) At class Level [Which we can declare inside the class but outside of the method]
        a) Class Variable OR Static Field
        b) Instance Variable OR Non static field

2) At Method Level [Which we can declare inside the method body OR parameter]
        a) Local Variable
        b) Parameter Variable

    Note : Block level variables are also local variable but the scope is within the same block only