**Generics in Java**

Generics means parameterized types. The idea is to allow type (Integer, String, … etc., and user-defined types) to be a parameter to methods, classes, and interfaces. Using Generics, it is possible to create classes that work with different data types. An entity such as class, interface, or method that operates on a parameterized type is a generic entity.

|  |  |
| --- | --- |
| |  | | --- | | // Java program to show working of user defined  // Generic classes    // We use < > to specify Parameter type  **class** Test<T> {      // An object of type T is declared      T obj;      Test(T obj) { **this**.obj = obj; } // constructor  **public** T getObject() { **return** **this**.obj; }  }    // Driver class to test above  **class** Main {  **public** **static** **void** main(String[] args)      {          // instance of Integer type          Test<Integer> iObj = **new** Test<Integer>(15);          System.out.println(iObj.getObject());            // instance of String type          Test<String> sObj              = **new** Test<String>("GeeksForGeeks");          System.out.println(sObj.getObject());      }  } |   **Output**  15  GeeksForGeeks |