**Day 7: 24-10-2025:**

**String :** String is a pre defined or also known as reference data types

Creating syntax for String class object

String str1 = “Welcome to Java”;

String str2 = new String(“Welcome to Java”);

== : it check contents and memory code

equals : it check only contents doesn’t matter same memory or different memory.

equals() method is part of Object class. String class as well as all Wrapper classes internally overrided equals method to check to object value and their memory code

String is known as immutable class. means we can’t change the value.

StringBuffer and StringBuilder : these two classes are type String mutable classes.

StringBuffer by default methods are synchronized. So it is thread safe but slow in performance in multithreading environment.

StringBuilder : not a synchronized so not thread safe. But fast in performance.

All these 3 classes and final. So we can’t inherits those classes.

**Generics**

Generics is way by which programmer can specify the type of object that class can work with Via parameter passes at declaration time and evaluate at compile as well as run time.

Non generic code in Collection framework

List ll = new ArrayList();

ll.add(10);

ll.add(10.20);

ll.add(“ravi”);

Object obj = ll.get(0);

integer i = (Integer)obj;

int n = i.intValue();

Generic code in Collection framework

List<Integer> ll = new ArrayList<>();

ll.add(10);

ll.add(20);

ll.add(30);

int n = ll.get(0);

Generic class : The generic class is a mechanism to specific the type of relationship between the component type and its object.

The syntax of declaration of generic class as normal class with angular brackets. The type of parameter are declared.

E -🡪 Element type

K 🡪 Key type

N 🡪 Number type

T 🡪 return type

V 🡪 value type

class Test<T> {

Test(T obj) {

System.out.println(obj);

}

}

Test t1 = new Test(10);

Test t2 = new Test(10.20);

Test t3 = new Test(“Ravi”);

If we create generic class with Type as T or any alphabets that T we can use inside constructor as well as methods.

If we need type for particular method. Then you can use generic method.

**Generics wild card**

In Generics we use the wild card as ? it represents an unknown types. This wild card we can use in variety of situation as a type of parameter, fields, local variable as well as return type.

1. ? any type
2. ? extends Type (upper bond)
3. ? super Type (lower bond)