## **Exception Handling** Throwable - Runtime problems - All the errors and exceptions - Error, Exception 1. Checked Exception try try(){ 2. UnChecked Exception catch throw throws Add on finally interface Closeable extends AutoClosable -> 1.5 interface Autocloseable -> 1.7 heap stack p1 city pincode name -0X200 Person p1 = new Person(); pune 411057 April Mukesh 0X200 \_p2 Person p2 = p1; - It copies references 0X200 pincode city name 0X200 Person p1 = new Person(); Anil pune 411057 0X200 p2 pincode city name Person p2 = p1.clone(); 0X300 - It copies the object Anit 411057 pune Mukesh 0X300 protected Object clone() throws CloneNotSupportedException{

reuturn

```
Person
 Employee
                                                                 String name = "sunbeam";
                          Person p = new Person();
 Date
                          Employee e = new Employee();
 Time
                                      Heap
                                String Literal/Constant Pool
                                                              SLP/ SCP
                                               "sunbeam"
        s1 -
                                      100
         s2 -
                                                                                 SLP
public static void main(String[] args) {
     String s1 = "sunbeam";
                                                                                   "sunbeam"
                                                                    s1 -
     String s2 = new String("sunbeam");
     System.out.println("s1 - " + s1);
     System.out.println("s2 - " + s2);
     System.out.println("s1==s2 - " + (s1 == s2)); // false
                                                                     s2
     System.out.println("s1.equals(s2) - " + s1.equals(s2));
                                                                                  'sunbeam"
                                                    // true
                                                                               SLP
  public static void main(String[] args) {
      String s1 = "sunbeam";
                                                                                 "sunbeam"
      String s2 = "sun" + "beam";
                                                                  s1 -
                                                                   s2
      System.out.println("s1 - " + s1);
      System.out.println("s2 - " + s2);
                                           true
      System.out.println("s1==s2 - " + (s1 == s2));
```

true

System.out.println("s1.equals(s2) - " + s1.equals(s2));

```
SLP
public static void main(String[] args) {
                                                                                  "sunbeam"
    String s1 = "sunbeam";
                                                                   s1
    String s2 = "sun";
                                                                                   "sun"
                                                                    s2-
    String s3 = s2 + "beam"; // runtime
                                                                                    "beam"
    System.out.println("s1 - " + s1);
    System.out.println("s3 - " + s3);
                                           // false
                                                                                sunbeam
                                                                     s3
    System.out.println("s1==s3 - " + (s1 == s3));
    System.out.println("s1.equals(s3) - " + s1.equals(s3));
                                                 //true
                                                                                        heap
                                                                               SLP
 public static void main(String[] args) {
     String s1 = "SUNBEAM";
                                                                                 "SUNBEAM"
                                                                     s1
     String s2 = "sunbeam";
     s2.toUpperCase();
                                                                                "sunbeam"
                                                                      s2-
     System.out.println("s1 - " + s1);
     System.out.println("s2 - " + s2);
     System.out.println("s1==s2 - " + (s1 == s2));
     System.out.println("s1.equals(s2) - " + s1.equals(s2));
                                                                             "SUNBEAM"
     // System.out.println("s3 - " + s3);
     // System.out.println("s1==s3 - " + (s1 == s3));
     // System.out.println("s1.equals(s3) - " + s1.equals(s3));
 public static void main(String[] args) {
                                                                                SLP
     String s1 = "SUNBEAM";
                                                                                  "SUNBEAM"
     String s2 = "sunbeam";
                                                                      s1
     String s3 = s2.toUpperCase();
                                                                                  "sunbeam"
                                                                       s2-
     System.out.println("s1 - " + s1);
     System.out.println("s2 - " + s2);
                                          //false
     System.out.println("s1==s2 - " + (s1 == s2));
     System.out.println("s1.equals(s2) - " + s1.equals(s2));
                                                 //false
                                                                              "SUNBEAM"
                                                                      s3
     System.out.println("s3 - " + s3);
                                         //false
     System.out.println("s1==s3 - " + (s1 == s3));
     System.out.println("s1.equals(s3) - " + s1.equals(s3));
                                                //true
```

```
public static void main(String[] args) {
   String s1 = "SUNBEAM";
   String s2 = "sunbeam";
   s2 = s2.toUpperCase();

   System.out.println("s1 - " + s1);
   System.out.println("s2 - " + s2);
   System.out.println("s1==s2 - " + (s1 == s2));
   System.out.println("s1.equals(s2) - " + s1.equals(s2));
   // true
```

```
s1 "SUNBEAM"

s2 "sunbeam"

"SUNBEAM"
```

```
SLP
public static void main(String[] args) {
    String s1 = "SUNBEAM";
                                    run time operation that creates
                                                                                     "SUNBEAM"
                                    a new String object on heap is now forced to be
                                                                        s1
    String s2 = "sunbeam";
                                    created on SLP
    s2 = s2.toUpperCase() .intern();
                                        // SUNBEAM
                                                                                     "sunbeam"
    System.out.println("s1 - " + s1);
    System.out.println("s2 - " + s2);
    System.out.println("s1==s2 - " + (s1 == s2));
    System.out.println("s1.equals(s2) - " + s1.equals(s2));
```

```
class Voter{

Voter v1 = new Voter("Anil","Pune","411057","Hinijewadi");

String name;

String city;

String pincode;

String area;
```

1 lakh objects of Voter -> Their could be 4 lakh total objects of String

1 lakh objects of Voter -> Their could be < 1 lakh total objects of String

## Mutable Strings

- 1. StringBuffer (Thread safe)
- 2. StringBuilder (Not Thread safe)

