CLASSTIME Pg No
Date / /

Ans 1 Mutable String means once we created a String object and try to change String object not change will be reflect on existing object not on new object

Eg: StringBuffer S = new StringBuffer ("Sahil");
S.append ("arora")
Sout(S);

Ans-2
```
Class A {
PSVm (String[] args) {
String S = "PWSKILL";
String S1 = "";
for (int i = S.length()-1; i>=0; i--) {
S1 = S1 + charAt(i);
}
Sout(S1);
}
}
```

Ans-3
```
Class A {
PSVm (String[] args) {
String S = "Think Skill";
String S1 = "";
for (int i = S.length()-1; i>=0; i--) {
S1 = S1 + charAt(i);
}
Sout(S1);
}
}
```

Ans 4
```
Class A {
PSVm (String[] args) {
String S = "Think Skill";
Char ch[] = S.toCharArray();
Arrays.sort(ch);
String qwe = new String(ch);
Sout(qwe);
```

Ans 1 mutable String means once we created a String object and try to change String object not change will be reflect on existing object not on new object

Ex: StringBuffer S = new StringBuffer("Sahil");
S.append("arora")
Sout(S);

Ans-2
```
Class A {
  PSVm (String[] args) {
  String S = "PWSKILL";
  String S1 = "";
  for (int i = S.length()-1; i>=0; i--) {
      S1 = S1 + charAt(i);
  }
  Sout(S1);
  }
}
```

Ans-3
```
Class A {
  PSVm (String[] args) {
    String S = "Think Skill";
    String S1 = "";
    for (int i = S.length()-1; i>=0; i--) {
        S1 = S1 + charAt(i);
    }
    Sout(S1);
  }
}
```

Ans 4
```
Class A {
  PSVm (String[] args) {
  String S = "Think Twice";
  char abc[] = S.toCharArray();
  Arrays.sort(abc);
  String qwe = new String(abc);
  Sout(qwe);
```