ASSIGNMENT 10

# 1.GET DESIRED CHARACTERS

# package assignment10;

# import java.util.Scanner;

# public class GetDesiredChar {

# public static void main(String[] args) { Scanner sc = new Scanner(System.*in*);

System.***out***.println("Enter a String");

String s = sc.next();

System.***out***.println("Enter Desired Character index"); **int** n = sc.nextInt(); **if**(n>s.length()) {

System.***out***.println("Invalid Index");

} **for**(**int** i=0;i<s.length();i++)

{ **if**(i==n) {

System.***out***.println("Desired character is "+s.charAt(i));

}

}

}

}

**2.UNICODE POINT**

**package** assignment10;

**import** java.util.Scanner;

**public** **class** GetUnicodeForDesiredChar { **public** **static** **void** main(String[] args) { Scanner sc = **new** Scanner(System.***in***); System.***out***.println("Enter a String");

String s = sc.next();

System.***out***.println("Enter Desired Character index"); **int** n = sc.nextInt(); **if**(n>s.length()) {

System.***out***.println("Invalid Index");

} **for**(**int** i=0;i<s.length();i++)

{ **if**(i==n) {

System.***out***.println("Unicode is : "+s.codePointAt(i));

}

}

}

}

**3.COMPARING LEXICOGRAPHICALLY**

**package** assignment10;

**import** java.util.Scanner;

**public** **class** GetUnicodeForDesiredChar { **public** **static** **void** main(String[] args) {

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Enter a String");

String s = sc.next();

System.***out***.println("Enter Desired Character index"); **int** n = sc.nextInt(); **if**(n>s.length()) {

System.***out***.println("Invalid Index");

} **for**(**int** i=0;i<s.length();i++)

{ **if**(i==n) {

System.***out***.println("Unicode is : "+s.codePointAt(i));

}

}

}

}

**4.COUNT OCCURENCES**

**package** assignment10;

**import** java.util.Scanner;

**public** **class** OccuranceOfDesiredCharacter { **public** **static** **void** main(String[] args) {

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Enter a String");

String s = sc.next();

System.***out***.println("Enter the index of a character for which you want to check the occurance"); **char** check=sc.next().charAt(0); **int** count=0; **for**(**int** i=0;i<s.length();i++) { **if**(s.charAt(i)==check)

count++;

}

System.***out***.println(count);

}

}

# 5.CONCATENATE N TIMES

# package assignment10;

# import java.util.Scanner;

# public class StringConcatenation {

# public static void main(String[] args) { // TODO

Auto-generated method stub

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Enter a String");

String s = sc.next();

System.***out***.println("Enter how many times you want to concatenate the string"); **int** n = sc.nextInt();

System.***out***.print(s.repeat(n));

}

}

# 6.STRINGS IN ASCENDING & DESCENDING ORDER

# package assignment10;

import java.util.Arrays; import java.util.Comparator; import java.util.Scanner; public class SortByLength { public static void main(String[] args) {

// TODO Auto-generated method stub

Scanner sc = new Scanner(System.in);

System.out.println("Ente the size of an array"); int n=sc.nextInt();

String[] a = new String[n];

System.out.println("Enter array elemets"); for(int i=0;i<a.length;i++) {

a[i]=sc.next();

}

Arrays.sort(a, Comparator.comparingInt(String::length));

System.out.println("Ascending order "+Arrays.toString(a));

Arrays.sort(a,Comparator.comparingInt(String::length).reversed()); System.out.println("Descending order :"+Arrays.toString(a));

}

}

# 7.STRING PALINDROME

# package assignment10;

**import** java.util.Scanner;

**public** **class** StringPalindrome {

**public** **static** **void** main(String[] args) { // **TODO**

Auto-generated method stub

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Enter a string"); String s1=sc.next(); String revs1=""; **for**(**int** i=s1.length()-

1;i>=0;i--) { revs1=revs1+s1.charAt(i);

} **if**(s1.equals(revs1))

{

System.***out***.println(s1+" is a palindrome");

} **else**

{

System.***out***.println(s1+" is not a palindrome");

}

}

}

# 8.STRINGS ARE IMMUTABLE

# package assignment10;

# public class ImmutableString {

# public static void main(String[] args) {

//EXAMPLE

/\* String s1 = "Mouni";

String s1 = "Adhya";

System.out.println(s1); \*/

//ERROR

/\*\*Exception in thread "main" java.lang.Error: Unresolved compilation problem:

\*Duplicate local variable s1 at assignment10.ImmutableString.main(ImmutableString.java:7)

\*/

}

}

**9.IMPLEMENTING INTERFACE package** assignment10;

**public** **class** FlyingVehicle {

**public** **void** fly() {

System.***out***.println("--Ready to fly");

}

**public** **void** land()

{

System.***out***.println("--Ready to land"); } }

**package** assignment10;

**public** **class** Spaceship **extends** FlyingVehicle{ **boolean** hyperdrive = **true**; @Override **public** **void** fly() { **super**.fly();

System.***out***.println("--Spaceship reporting");

} @Override **public** **void** land() { **super**.land();

System.***out***.println("--Spaceship reporting"); } }

**package** assignment10; **public** **class** Airplane **extends** FlyingVehicle{ **int** paasengers = 50;

@Override **public** **void** fly() { **super**.fly();

System.***out***.println("--Airplane reporting");

} @Override **public** **void** land() { **super**.land();

System.***out***.println("--Airplane reporting"); } }

**package** assignment10; **public** **class** GroundVehicle { **public** **void** drive() {

System.***out***.println("--Driving mode on"); } }

**package** assignment10;

**public** **class** Car **extends** GroundVehicle {

String noPlate="KA2405"; @Override **public** **void** drive() {

**super**.drive();

System.***out***.println("--Car reporting");

} **public** **void** PonderEthicalDilemma()

{

System.***out***.println("--From Ponder Ethical Dilemma");

}

}

**package** assignment10; **public** **class** Truck **extends** GroundVehicle { **double** capacity =

570.14;

@Override **public** **void** drive() { **super**.drive();

System.***out***.println("--Truck reporting");

} **public** **void** loadCargo()

{

System.***out***.println("--From Load Cargo"); } }

**package** assignment10; **public** **class** TestVehicles { **public** **static** **void** main(String[] args) {

FlyingVehicle flyingVehicle ;

System.***out***.println("-------Flying Vehicles------"); flyingVehicle = **new** Spaceship(); System.***out***.println("@Spaceship fly method"); flyingVehicle.fly();

System.***out***.println("@Spaceship land method"); flyingVehicle.land(); flyingVehicle = **new** Airplane(); System.***out***.println("@Airplane fly method"); flyingVehicle.fly();

System.***out***.println("@Airplane land method"); flyingVehicle.land();

System.***out***.println("-------Ground Vehicles------");

GroundVehicle groundVehicle; groundVehicle = **new**

Car();

System.***out***.println("@Car drive method"); groundVehicle.drive(); Car car = **new**

Car();

String cNoPlate=car.noPlate;

System.***out***.println("@Car number plate : "+cNoPlate); System.***out***.println("@Car Ponder Ethical Dilemma method"); car.PonderEthicalDilemma(); groundVehicle = **new** Truck(); System.***out***.println("@Truck drive method"); groundVehicle.drive(); Truck truck = **new**

Truck(); **double** tCapacity=truck.capacity;

System.***out***.println("@Truck capacity : "+tCapacity); System.***out***.println("@Truck Load Cargo method"); truck.loadCargo();

}

}