ASSIGNMENT – 10

submitted by,
Santhanam .L

1. Write a Java program to get the character at the given index within the String:

PROGRAM:

```java
package assignement10;

import java.util.Scanner;

public class TogetChar {

public static void main(String[] a)

{

Scanner scan=new Scanner(System.in);

System.out.println("Enter the Word: ");

String word=scan.nextLine();

for(int i=0;i<word.length();i++)

{

char ch=word.charAt(i);

System.out.println("Letter: "+ch +" Possition "+i);

}

char F=word.charAt(0);
char L=word.charAt(word.length()-1);

System.out.println("First Letter: "+ F + "\n" +
"Last Letter " +L);

System.out.println("Enter the possition num");

int num=scan.nextInt();

char c=word.charAt(num);

System.out.println("This position letter: "+c);

}

}
```

OUTPUT:

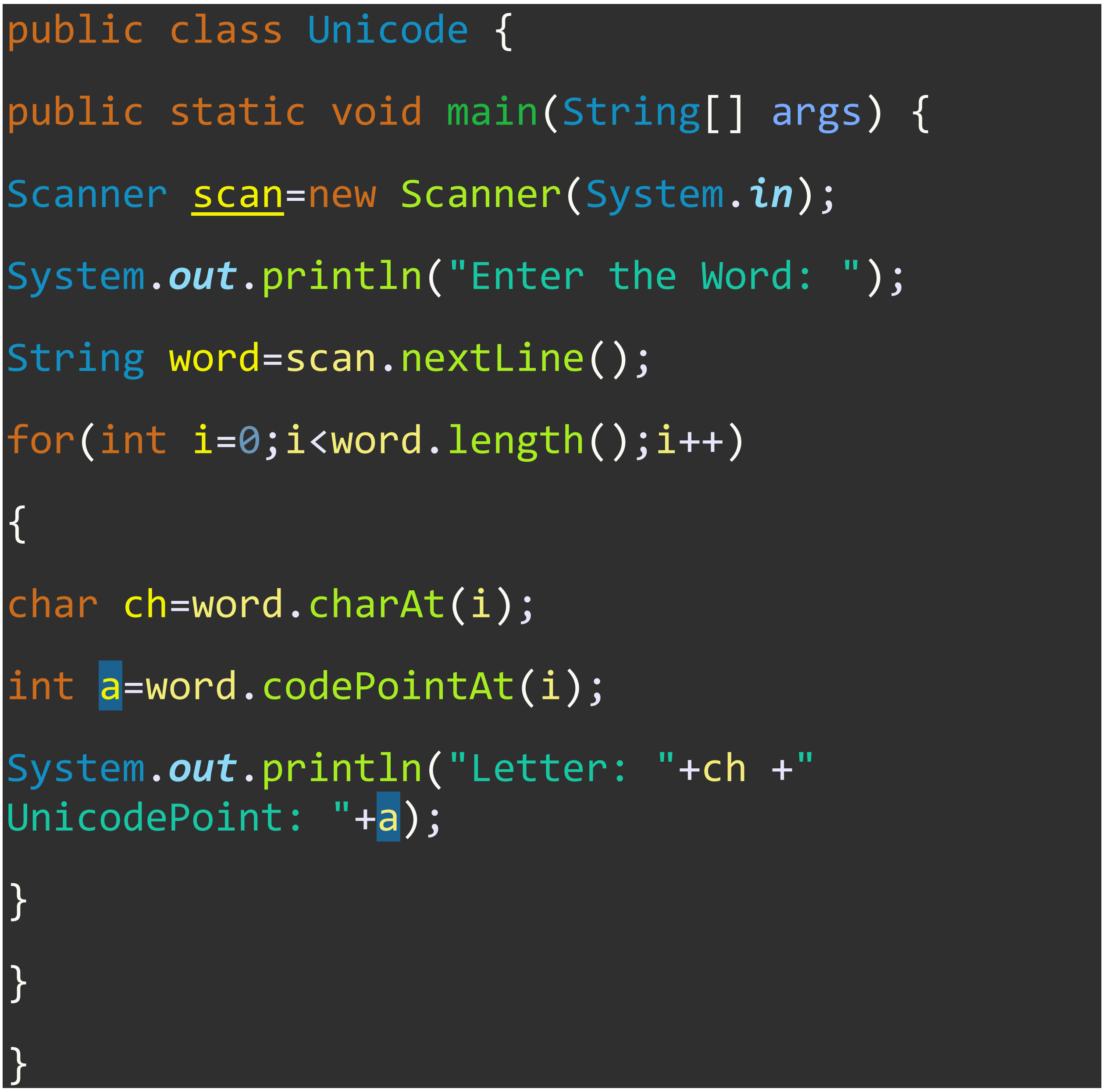
```
Enter the Word:

sandel

Letter: s Possition 0

Letter: a Possition 1

Letter: n Possition 2

Letter: d Possition 3

Letter: e Possition 4

Letter: l Possition 5
```

```
First Letter: s

Last Letter l

Enter the possition num

2

This position letter: n
```

2. Write a Java program to get the character (Unicode code point) at the given index within the String

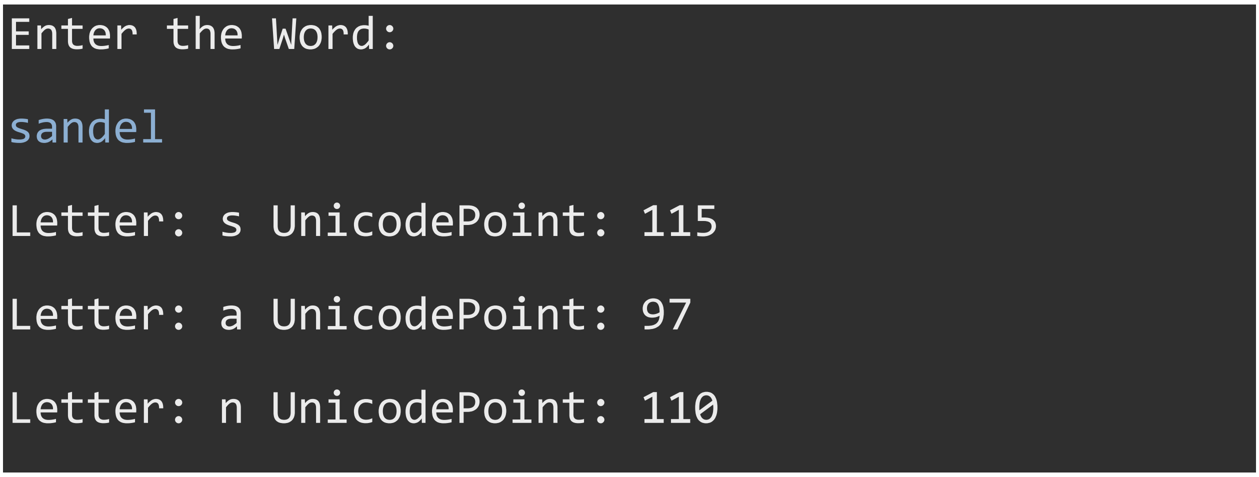
PROGRAM:

```
package assignement10;

import java.util.Scanner;
```



```java
public class Unicode {

public static void main(String[] args) {

Scanner scan=new Scanner(System.in);

System.out.println("Enter the Word: ");

String word=scan.nextLine();

for(int i=0;i<word.length();i++)

{

char ch=word.charAt(i);

int a=word.codePointAt(i);

System.out.println("Letter: "+ch +" UnicodePoint: "+a);

}

}

}
```

OUTPUT:



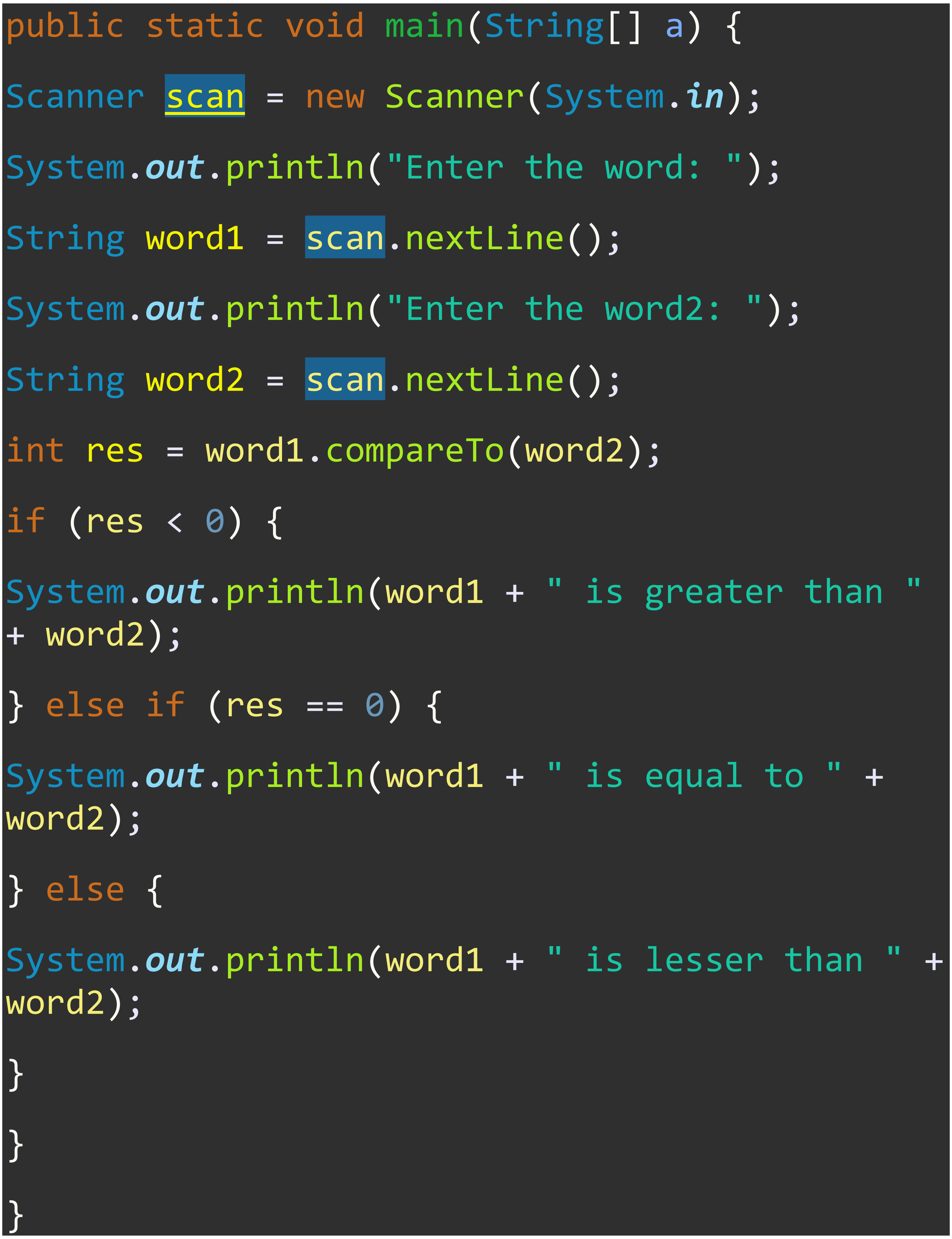
```
Enter the Word:

sandel

Letter: s UnicodePoint: 115

Letter: a UnicodePoint: 97

Letter: n UnicodePoint: 110
```

```
Letter: d UnicodePoint: 100

Letter: e UnicodePoint: 101

Letter: l UnicodePoint: 108
```

3. Write a Java program to compare two strings lexicographically. Two strings are lexicographically equal if they are the same length and contain the same characters in the same positions

PROGRAM:

```
package assignement10;

import java.util.Scanner;

public class CompareString {
```



```
public static void main(String[] a) {
Scanner scan = new Scanner(System.in);
System.out.println("Enter the word: ");
String word1 = scan.nextLine();
System.out.println("Enter the word2: ");
String word2 = scan.nextLine();
int res = word1.compareTo(word2);
if (res < 0) {
System.out.println(word1 + " is greater than "
+ word2);
} else if (res == 0) {
System.out.println(word1 + " is equal to " +
word2);
} else {
System.out.println(word1 + " is lesser than " +
word2);
}
}
}
```

OUTPUT:

```
Enter the word:

Santhanam

Enter the word2:

Santhanam

Santhanam is equal to Santhanam
```

4.Write a Java program to counts occurrences of a certain character in a given string

PROGRAM:

```
package assignement10;

import java.util.Scanner;

public class Occurrences {

public static void main(String[] a)

{

int count=0;

Scanner scan=new Scanner(System.in);

System.out.println("Enter the word: ");

String word=scan.nextLine();
System.out.print("Enter the Letter:");

char ch=scan.next().charAt(0);

for(int i=0;i<word.length();i++)

{

char L=word.charAt(i);

if(L==ch)

{

count++;

}

}

System.out.println("count: "+count);

}

}
```

OUTPUT:

```
Enter the word:

santhanam

Enter the Letter:a
```

```
count: 3
```

5.Write a Java program to concatenate a given string with itself of a given number of times.

PROGRAM:

```
package assignement10;

import java.util.Scanner;

public class Concantenat {

public static void main(String[] args) {

Scanner scan=new Scanner(System.in);

System.out.println("Enter the word");

String word=scan.nextLine();

System.out.println("Enter the Number you want:");

int num=scan.nextInt();

String res;

for(int i=0;i<num;i++)

{

System.out.print(word);

}

}

}
```

OUTPUT:

```
Enter the word
santhanam
Enter the Number you want:
3
santhanamsanthanamsanthanam
```

6. **Write a Java program to sort in ascending and descending order by length of the given array of strings.**

Sample Output:
Original unsorted colors: [Green,
White, Black, Pink, Orange, Blue, Champagne, Indigo, Ivory]
Sorted color (descending order): [Champagne, Orange, Indigo,
Green, White, Black, Ivory, Pink, Blue]
Sorted color (ascending order): [Pink, Blue, Green, White, Black,
Ivory, Orange

PROGRAM:

```java
package assignement10;
import java.lang.reflect.Array;
import java.util.*;
public class ArraySorting {
```

```java
public static void main(String[] a)
{
Scanner scan=new Scanner(System.in);
System.out.println("Enter the Array size");
int num=scan.nextInt();
String str[]=new String[num];
for(int i=0;i<num;i++)
{
str[i]=scan.next();
}
System.out.println("Unorder list: "+ Arrays.toString(str));
Arrays.sort(str, Comparator.comparingInt(String::length));
System.out.println("Sorted color (descending order): : " + Arrays.toString(str));
Arrays.sort(str, Comparator.comparingInt(String::length).reversed());
System.out.println("Sorted color (ascending order): : " + Arrays.toString(str));
}
```

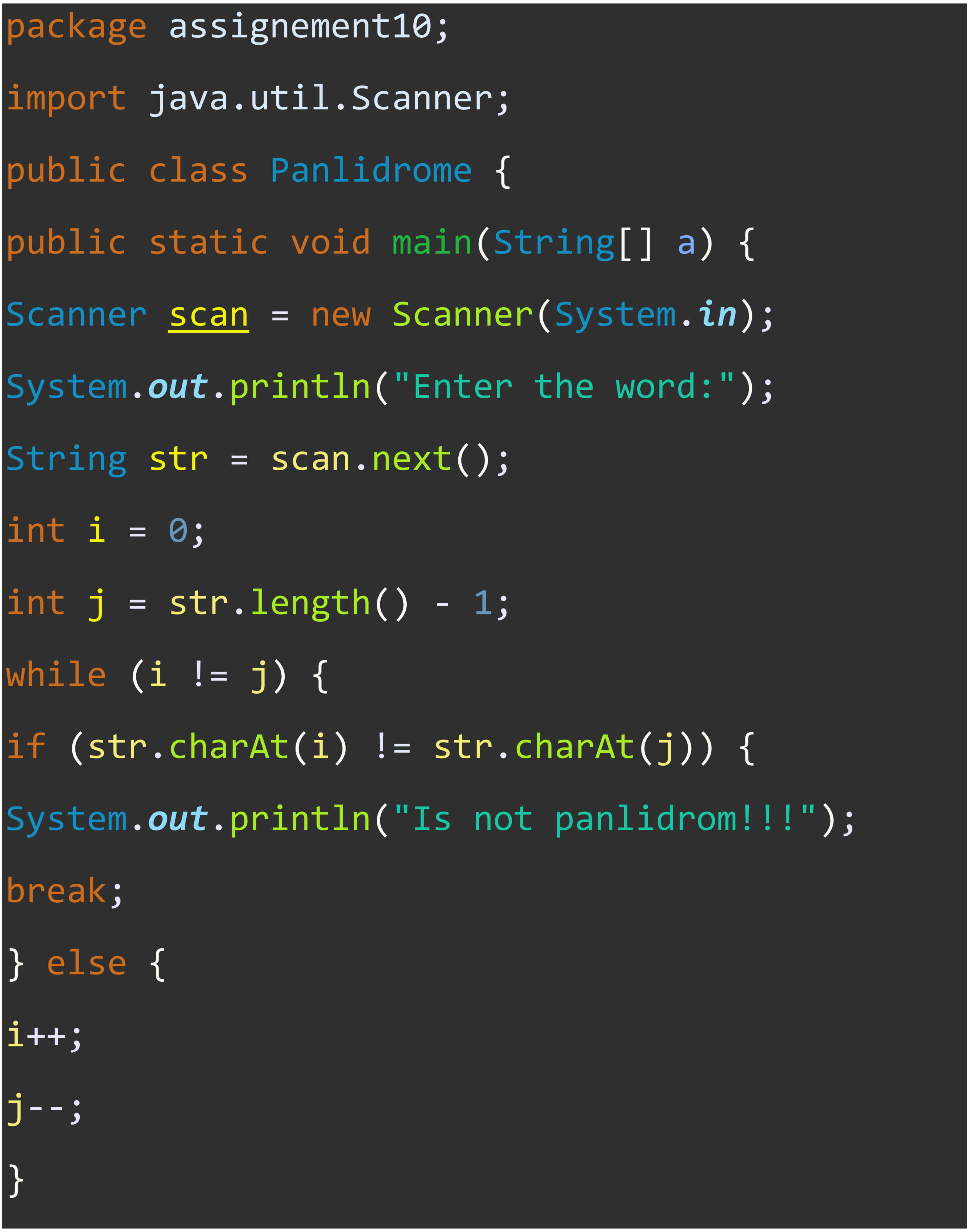
```
}
```

OUTPUT:

```
 Enter the Array size

9

Green

white

Black

Pink

Blue

Orenge

champagne

Indigo

ivory

Unorder list: [Green, white, Black, Pink, Blue,
Orenge, champagne, Indigo, ivory]

Sorted color (descending order): : [Pink, Blue,
Green, white, Black, ivory, Orenge, Indigo,
champagne]

Sorted color (ascending order): : [champagne,
Orenge, Indigo, Green, white, Black, ivory,
Pink, Blue]
```

7. check the given string is panlidrome or not

PROGRAM:



```
package assignement10;
import java.util.Scanner;
public class Panlidrome {
public static void main(String[] a) {
Scanner scan = new Scanner(System.in);
System.out.println("Enter the word:");
String str = scan.next();
int i = 0;
int j = str.length() - 1;
while (i != j) {
if (str.charAt(i) != str.charAt(j)) {
System.out.println("Is not panlidrom!!!");
break;
} else {
i++;
j--;
}
}
System.out.println("this is Panlidrom");
}
}
```

OUTPUT:

```
Enter the word:
level
this is Panlidrom
```

8. Java Program to prove that strings are immutable in java

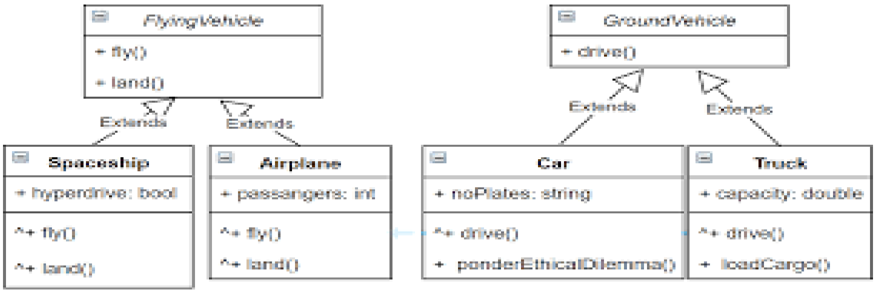
PROGRAM:

```
package assignement10;
public class Immutable {
public static void main(String[] args) {
String s1="hello";
String s2=s1;//s2 poiting s1----"hello"
System.out.println(s1.hashCode()+" --->"+s1);//hello
System.out.println(s2.hashCode()+"--->"+s2);//hello
System.out.println("Both memory address are same");
s1="hello world";
System.out.println("after modify");
System.out.println(s1.hashCode()+" -->"+s1);//hello world
System.out.println(s2.hashCode() + "---> still poiting "+ s2);// still poiting s1 "hello
}
}
```

OUTPUT:

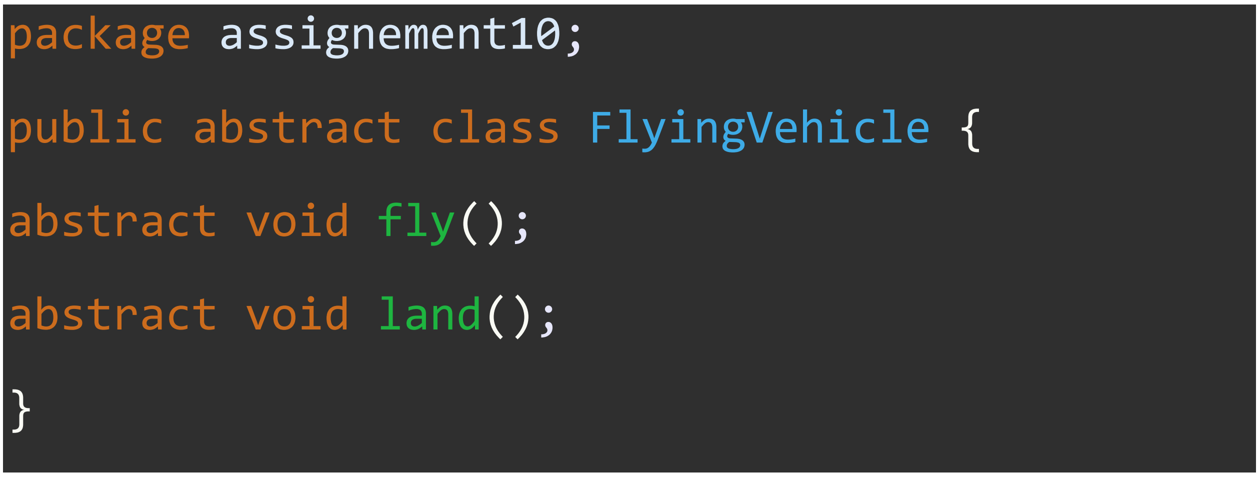
```
99162322 --->hello
99162322---> hello
Both memory address are same
after modify
1794106052 -->hello world
99162322---> still poiting hello
```

9.Java program to implement below classes using inheritance



PROGRAM:

Parent class: Flying Vehicle



```
package assignement10;
public abstract class FlyingVehicle {
abstract void fly();
abstract void land();
}
```

Child class: Spaceship

```
package assignement10;

public class SpaceShip extends FlyingVehicle {

boolean status;

public SpaceShip(boolean status) {

this.status = status;

}

@Override

void fly() {

System.out.println(" Flight Ready to Fly....
status: "+status);

}

void land() {

System.out.println("Flight Ready
Land.....status: "+status);

}

}
```

Child class: AirPlane

```
package assignement10;

public class Airplane extends FlyingVehicle {

int passanger;

public Airplane(int passanger) {

this.passanger = passanger;

}

void fly() {

System.out.println("Welcome !!! Ready to Fly
Plese check your *Seatbelt!!*");

System.out.println("Happy journey");

}

@Override

void land() {

System.out.println(" Ready to Land Plese check
your *Seatbelt!!*");

System.out.println("*Welcome Back*");
}
}
```

Parent class: Ground Vehicle

```
package assignement10;
public abstract class GroundVehicle {
abstract void drive();
}
```

Child class: Car

```
package assignement10;
public class Car extends GroundVehicle {
String noPlate;
public Car(String noPlate) {
this.noPlate = noPlate;
System.out.println("mycar number: "+noPlate);
}
@Override
void drive() {

System.out.println("Self Drive Cars");

}

void AutoDrive() {

System.out.println("Autometic Drive Cars");

System.out.println("this is unSafe for Indian
Roads");

}

}
```

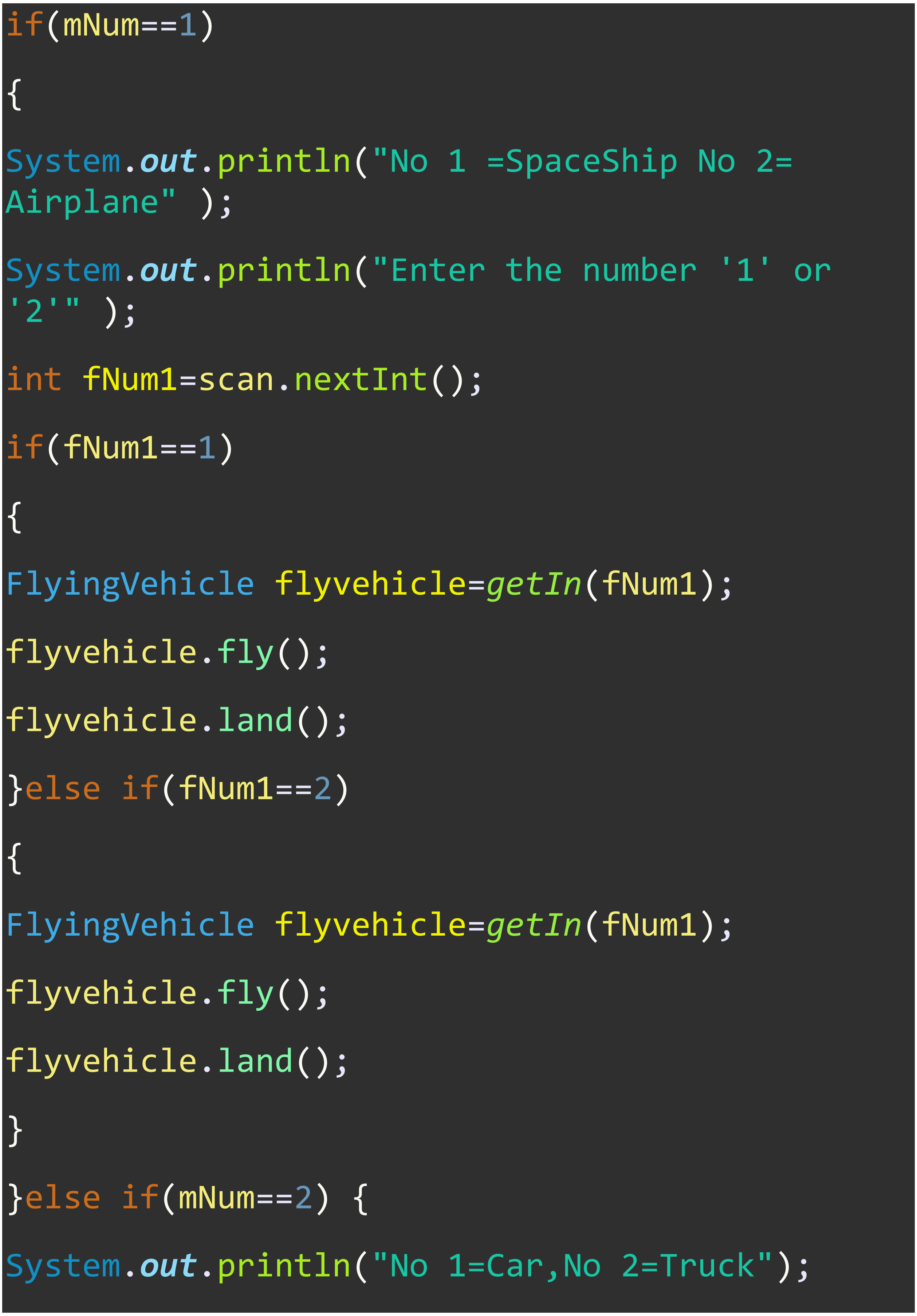
Child class: Truck

```
package assignement10;

public class Truck extends GroundVehicle{

double capacity;

public Truck(double capacity) {

this.capacity = capacity;

System.out.println("capacity: "+capacity);

}

@Override

void drive() {
```

```
System.out.println("it is Drive very Hard so
");

}

void Loadcargo() {

System.out.println("it can be handle exprience
driver only!! "+ capacity);

}

}
```

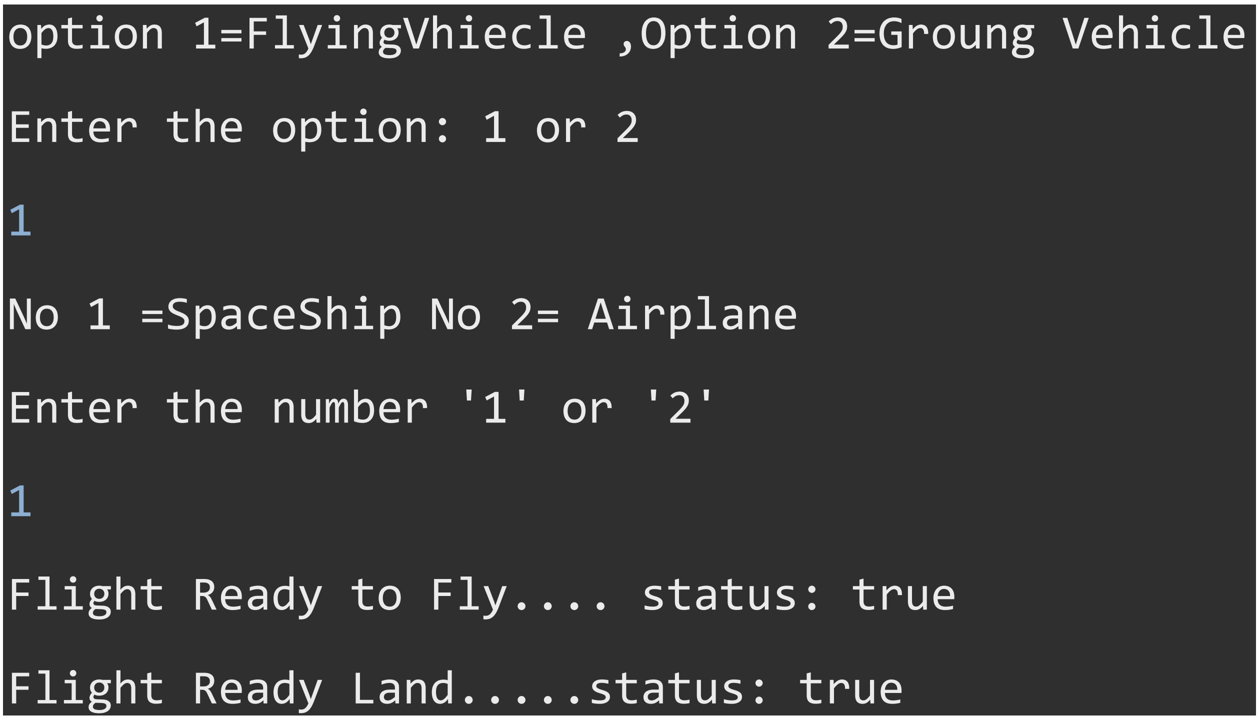
Test Class:

```
package assignement10;

import java.util.*;

public class TestVehicles {

static FlyingVehicle getIn(int a) {

if(a==1) {

return new SpaceShip(true);

}

else if(a==2)

{

return new Airplane(100);
}
return null;
}
static GroundVehicle get(int a) {
if(a==1) {
return new Car("TN 39 AF8965");
}
else if(a==2)
{
return new Truck(500.6);
}
return null;
}
public static void main(String[] a) {
Scanner scan=new Scanner(System.in);
System.out.println("option 1=FlyingVhiecle ,Option 2=Groung Vehicle");
System.out.println("Enter the option: 1 or 2");
int mNum;
mNum=scan.nextInt();
```

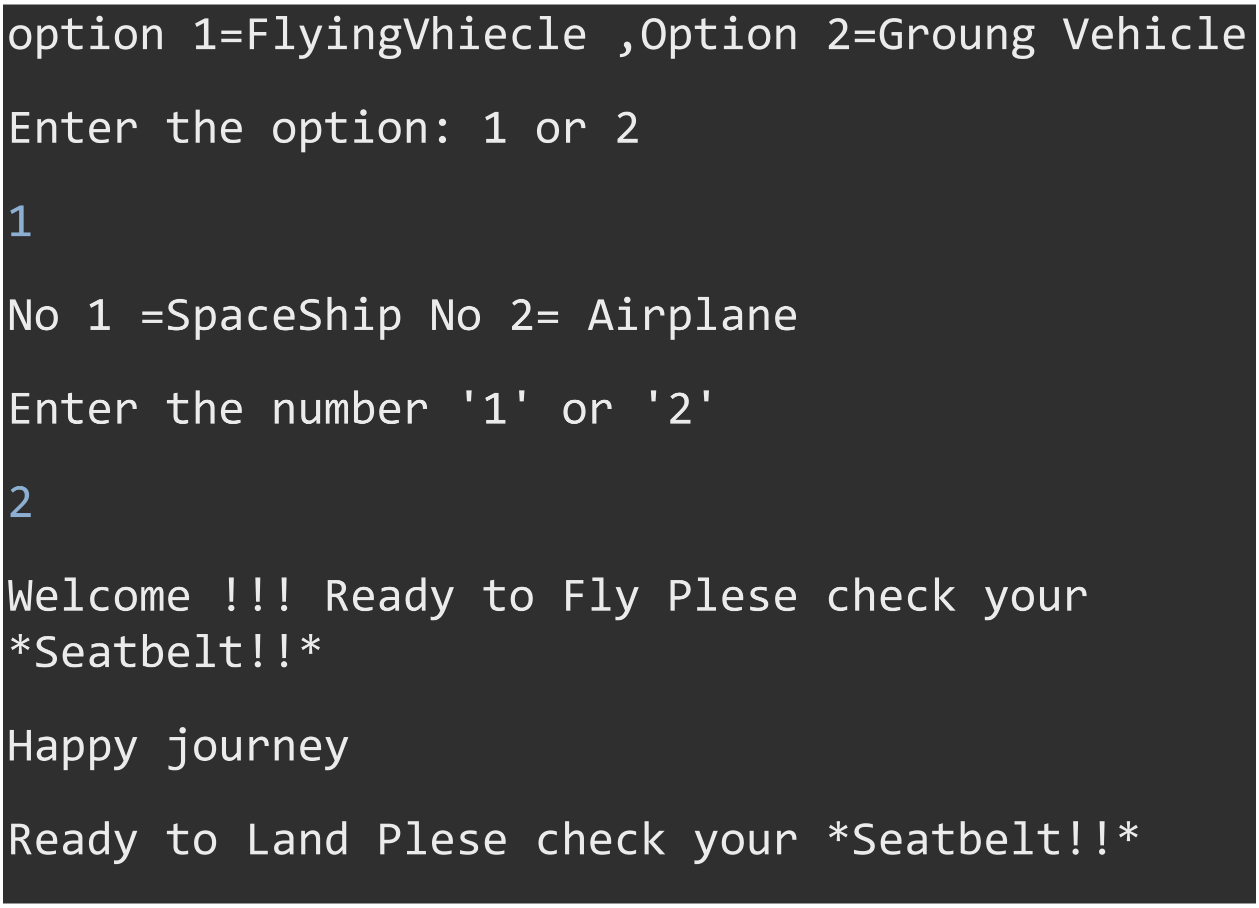


```java
if(mNum==1)

{

System.out.println("No 1 =SpaceShip No 2= Airplane" );

System.out.println("Enter the number '1' or '2'" );

int fNum1=scan.nextInt();

if(fNum1==1)

{

FlyingVehicle flyvehicle=getIn(fNum1);

flyvehicle.fly();

flyvehicle.land();

}else if(fNum1==2)

{

FlyingVehicle flyvehicle=getIn(fNum1);

flyvehicle.fly();

flyvehicle.land();

}

}else if(mNum==2) {

System.out.println("No 1=Car,No 2=Truck");
System.out.println("Enter the no '1' or '2'
:");

int gNum=scan.nextInt();

if(gNum==1)

{

GroundVehicle groundvehile=get(gNum);

groundvehile.drive();

}

else if(gNum==2)

{

GroundVehicle groundvehile=get(gNum);

groundvehile.drive();

}

}

scan.close();

}

}
```

OUTPUT:



```
option 1=FlyingVhiecle ,Option 2=Groung Vehicle

Enter the option: 1 or 2

1

No 1 =SpaceShip No 2= Airplane

Enter the number '1' or '2'

1

Flight Ready to Fly.... status: true

Flight Ready Land.....status: true
```



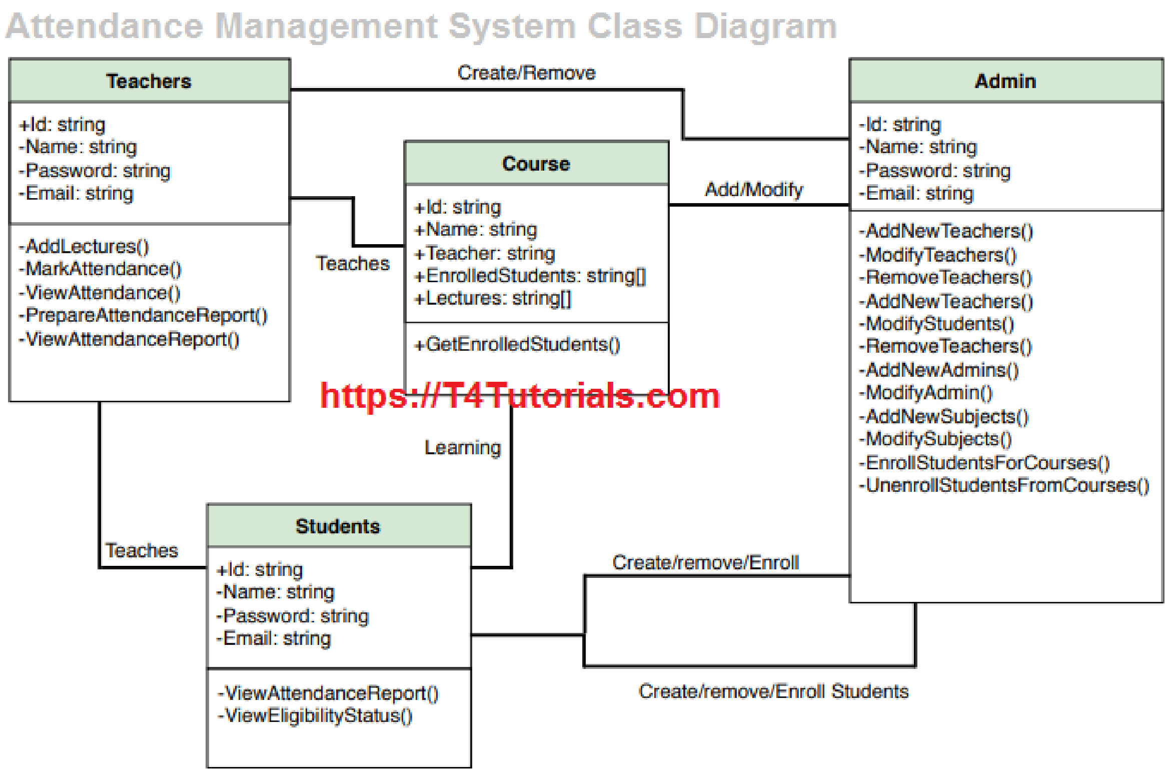
```
option 1=FlyingVhiecle ,Option 2=Groung Vehicle

Enter the option: 1 or 2

1

No 1 =SpaceShip No 2= Airplane

Enter the number '1' or '2'

2

Welcome !!! Ready to Fly Plese check your
*Seatbelt!!*

Happy journey

Ready to Land Plese check your *Seatbelt!!*
```

```
*Welcome Back*
```

```
option 1=FlyingVhiecle ,Option 2=Groung Vehicle

Enter the option: 1 or 2

2

No 1=Car,No 2=Truck

Enter the no '1' or '2' :

1

mycar number: TN 39 AF8965

Self Drive Cars
```

```
option 1=FlyingVhiecle ,Option 2=Groung Vehicle

Enter the option: 1 or 2

2

No 1=Car,No 2=Truck

Enter the no '1' or '2' :

2

capacity: 500.6

it is Drive very Hard so
```

10.Write a java program to implement the below diagram



Attendance Management System Class Diagram
Teachers
+Id: string
-Name: string
-Password: string
-Email: string
-AddLectures()
-MarkAttendance()
-ViewAttendance()
-PrepareAttendanceReport()
-ViewAttendanceReport()
Create/Remove
Admin
-Id: string
-Name: string
-Password: string
-Email: string
-AddNewTeachers()
-ModifyTeachers()
-RemoveTeachers()
-AddNewTeachers()
-ModifyStudents()
-RemoveTeachers()
-AddNewAdmins()
-ModifyAdmin()
-AddNewSubjects()
-ModifySubjects()
-EnrollStudentsForCourses()
-UnenrollStudentsFromCourses()
Course
+Id: string
+Name: string
+Teacher: string
+EnrolledStudents: string[]
+Lectures: string[]
+GetEnrolledStudents()
Add/Modify
Teaches
https://T4Tutorials.com
Learning
Teaches
Students
+Id: string
-Name: string
-Password: string
-Email: string
-ViewAttendanceReport()
-ViewEligibilityStatus()
Create/remove/Enroll
Create/remove/Enroll Students