Write a program to store & print the Aadhar details

```
class Aadhardetails {
Public static void main (String [ ] args)
{
String name = "Vinitha";
String fatherName = "Srinivasa Rao";
int DOB = 13/06/2001;
int age = 21;
String address = "Kodad";
long Contact = 7981287241;
System.out.println (name);
System.out.println (fatherName);
System.out.println (DOB);
System.out.println (age);
System.out.println (address);
System.out.println (contact);
}
}
```

Type Casting

We have two types of type casting.

1. Primitive type casting.
   (i) Auto Widening.
   (ii) Explicit narrowing.
2. Non-primitive type casting.
   (i) Auto - upCasting.
   (ii) Explicit - downCasting.

1. Primitive type Casting

The process of converting the value from one Primitive data type into another primitive data type. We have two classifications of it.

(i) Auto-Widening :-

The process of converting the value from a lesser data type into larger data type.. Since, during the process there is possibility of loosing the values. Compiler automatically (or) implicitly adds the type cast operator. If developer forgets to add it from JDK 1.5 version. Hence, we call Auto-widening.

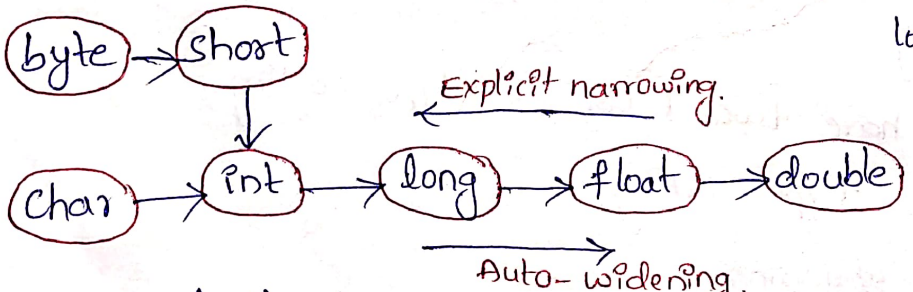
(ii) Explicit Narrowing :-

The process of converting the value from larger data type into smaller data type. Since, during the process of there is a possibility of loosing the values. during the conversion compiler throughs Compile time error (CTE). to avoid this developer must add the type cast operator. Hence, it is explicit narrowing.

Explicit narrowing

```
long l = 35.56f; //CTE
```
Explicit

```
long l = (long) 35.56f;
//no CTE
```



Auto-widening.

```
double d = 35;
float f = 10l;
double d1 = 'a';
```

Java 5 from JDK 1.5 →

```
double d1 = (double) 'a';
```

JVM → Only double Values → d_1 97.0 / 86

Program:-

```
class Demo
{
  Public static void main (String [ ] args)
  {
    int a = 'a' ; //
    System.out.println (a) ; //97.

    double d = 35; //d --> 35.00
    System.out.println (d); //35.00

    double d1='□' ; //48.0
    System.out.println (d1); //48.0.

  // from JDK 1.5 auto-widening.
    double d2 = 34; //-- javac --> double d2 =(double) 34;
    // double d2 = (double) 34; -- java c --> double d2 =(double) 34;
                System.out.println (d2);
  // from JDK
        // int i= true; CTE
        float f1 = (float) 99.99 ; // explicit-narrowing.
        // long l = 35.56f ; // here float is larger dt than
            that of // long hence CTE
        char ch=63; // even through it is narrowing we dont get CTE
            System.out.println (ch); // 'a'
  } [float f1 = (float) 99.99; // explicit-narrowing
  } //long l= 35.56f ; // here float is larger]
```