Core Java

UNIT 1

Take Home Task Solution

Write a Java Code to read an integer from user and find Quotient and Remainder.

```
public class QuotientRemainder {
    public static void main(String[] args) {
        int n1,n2;
        Scanner sc =new Scanner(System.in);
        System.out.println("Enter 2 numbers");
        n1=sc.nextInt();
        n2=sc.nextInt();
        System.out.println("Quotient="+n1/n2);
        System.out.println("Remainder="+n1%n2);
    }
}
```

Write a Java code to read any character from user and print its ASCII Value.

```
public class AsciiValue {
    public static void main(String[] args) {
        char ch;
        Scanner sc =new Scanner(System.in);
        System.out.println("Enter a character");
        ch=sc.next().charAt(0);
        int a=ch;
        System.out.println("Ascii Value of "
                                   +ch+" is "+a);
    }
}
```

Learning Outcomes

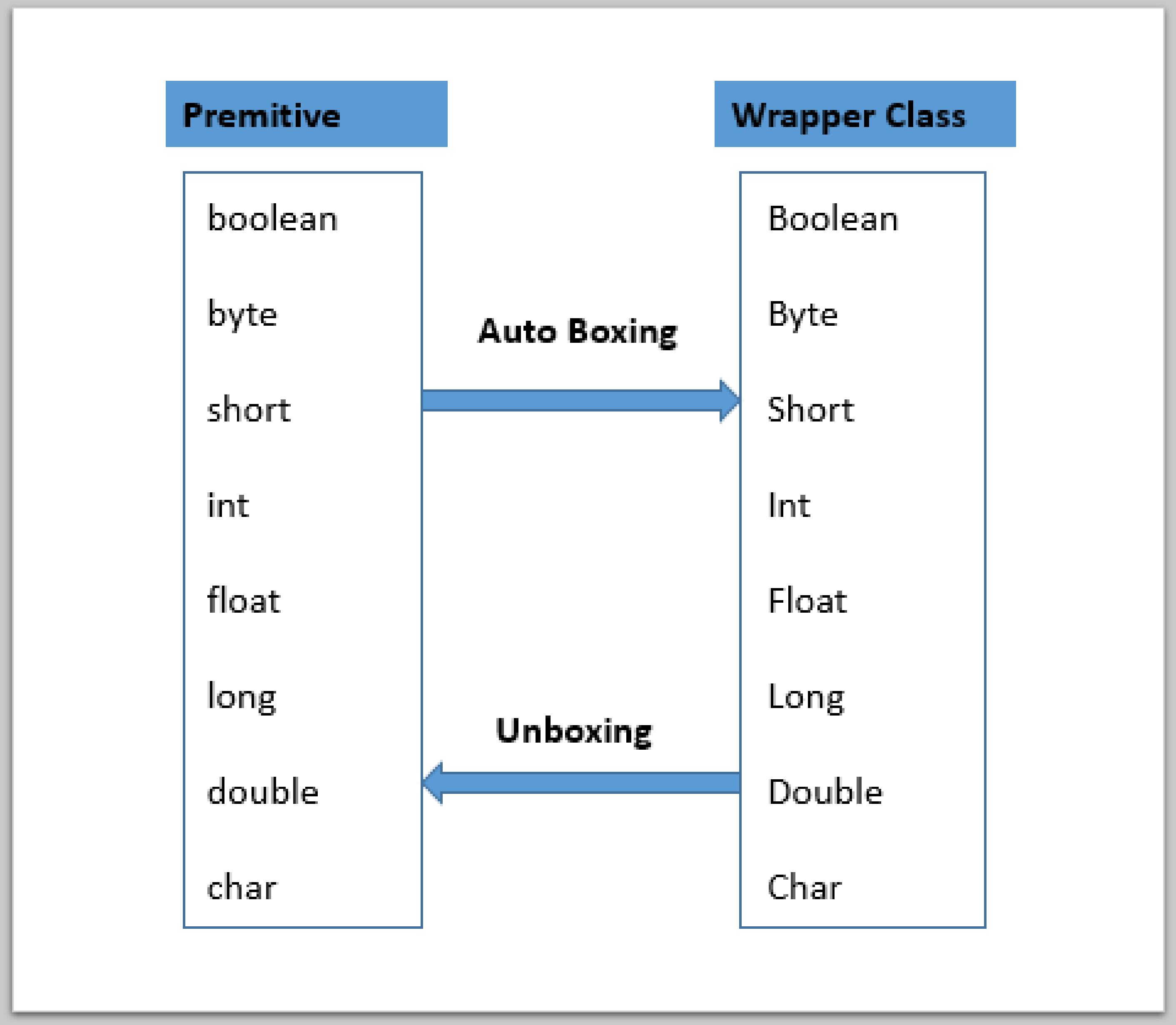
- Autoboxing and Unboxing
- Java Operators

Wrapper Classes

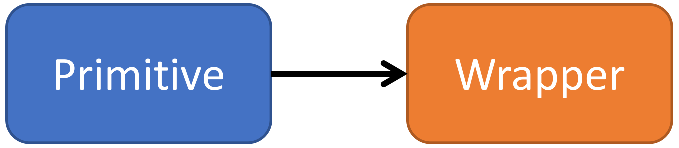
- A Wrapper class is a class whose **object** wraps or contains primitive data types.

- Example -

  Integer I = new Integer(25);



Autoboxing

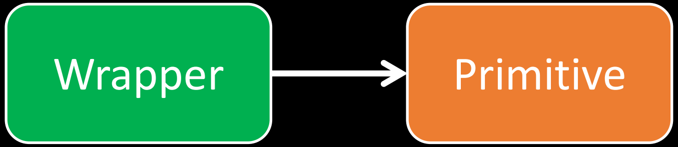


- The automatic conversion of primitive data types into its equivalent Wrapper type is known as **Autoboxing** (boxing).

```
class Autoboxing{
 public static void main(String[] args){

 int a=50;

 Integer a2=new Integer(a);  //Boxing

 Integer a3=5;    //Boxing

 System.out.println(a2+" "+a3);
}
}
```

```
Output:-
50  5
```

Unboxing



- The automatic conversion of Wrapper type into its equivalent Primitive type is known as **Unboxing.**

```
class Unboxing {
 public static void main(String args[]){
 Integer i=new Integer(50);
 int a=i;
 System.out.println(a);
}
}
```

Output:-

```
50
```

Java Operators

- Arithmetic
- Relational
- Logical
- Bitwise
- Conditional

Arithmetic Operators

Operator	Result
+	Addition
-	Subtraction
*	Multiplication
/	Division
%	Modulus
++	Increment
--	Decrement
+=	Addition Assignment
-=	Subtraction Assignment
*=	Multiplication Assignment
/=	Division Assignment
%=	Modulus Assignment

Relational Operators

Operator	Result
==	Equal to
>	Greater than
<	Less than
<=	Less than equal to
>=	Greater than equal to
!=	Not Equal to

Logical Operators

Operator	Result
&&	Logical AND
||	Logical OR
!	Logical Not

Bitwise Operators

Operator	Result
~	Bitwise unary NOT
&	Bitwise AND
|	Bitwise OR
^	Bitwise exclusive OR
>>	Shift Right
>>>	Shift right zero fill
<<	Shift left
&=	Bitwise AND assignment
!=	Bitwise OR assignment
^=	Bitwise exclusive OR assignment
>>=	Shift right assignment
>>>=	Shift right zero fill assignment
<<=	Shift left assignment

Conditional Operator (?:)

- Also known as Ternary Operator
- General Form: **exp1 ? exp2 : exp3**

Quiz

Take a home task

- ✓ List down String class Methods and their functionality
- Revise Entire Unit - 1

Thank You