CDAC Mumbai PG-DAC August 24

Assignment No- 4

1) Write a program that demonstrates widening conversion from int to double and prints the result.

CODE:

```
package org.example;

public class Program1 {
	public static void main(String[] args) {
		int num = 10;

		double num1 = num;

		System.out.println("Int to Double Number: "+num1);
	}
}
```

OUTPUT:

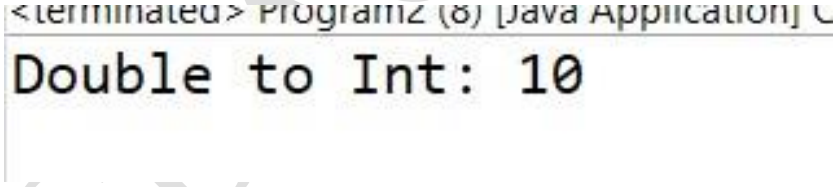
```
Int to Double Number: 10.0
```

2) Create a program that demonstrates narrowing conversion from double to int and prints the result.

CODE:

```
package org.example;

public class Program2 {
    public static void main(String[] args) {
        double num = 10;

        int num1 = (int) num;

        System.out.println("Double to Int: "+num1);
    }

}
```

OUTPUT:



3) Write a program that performs arithmetic operations involving different data types (int, double, float) and observes how Java handles widening conversions automatically.

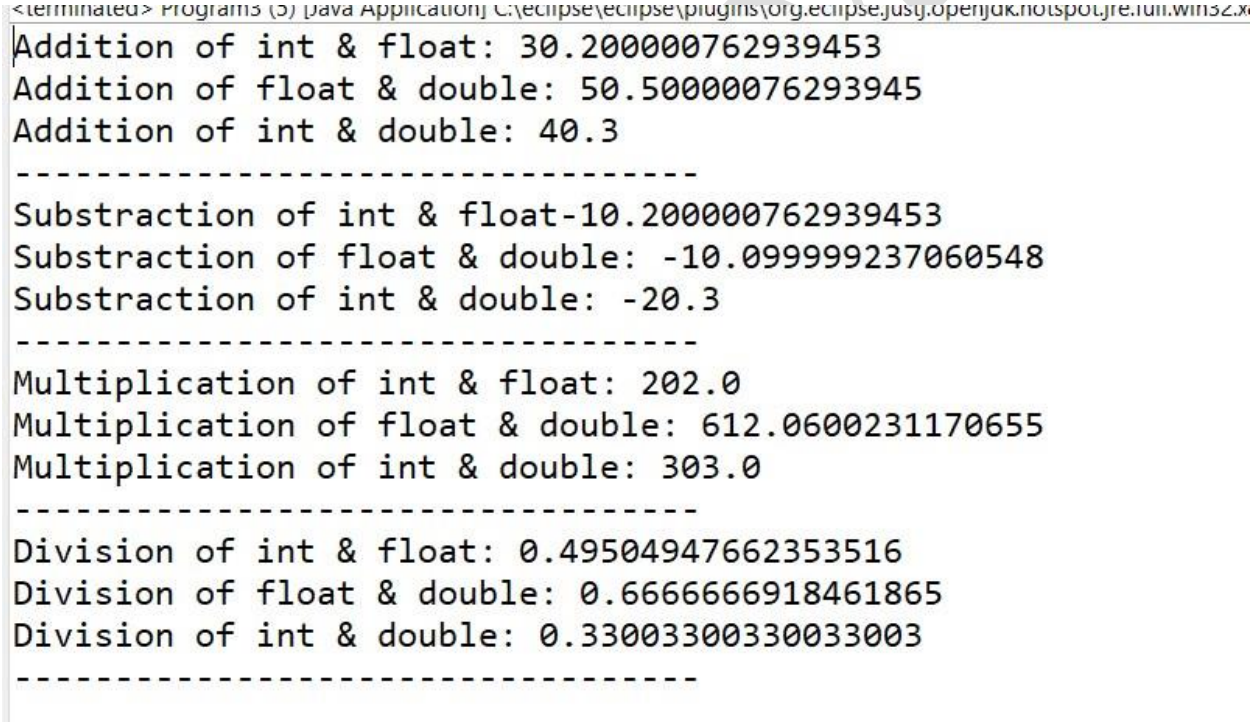
CODE:

```
package org.example;

public class Program3 {
	public static void main(String[] args) {
		int num1 = 10;
		float num2 = 20.2f;
		double num3 = 30.3;

		//Addition
		double sum = num1 + num2;
		System.out.println("Addition of int & float: "+ sum);

		double sum1 = num2 + num3;
		System.out.println("Addition of float & double: "+
sum1);

		double sum2 = num1 + num3;
		System.out.println("Addition of int & double: "+
sum2);
		System.out.println("---------------------------------");

		//Substraction
		double sub = num1 - num2;
		System.out.println("Substraction of int & float"+ sub);
```

```
        double sub1 = num2 - num3;
        System.out.println("Substraction of float & double: "+
sub1);

        double sub2 = num1 - num3;
        System.out.println("Substraction of int & double: "+
sub2);
        System.out.println("--------------------------------");

        //Multiplication
        double mul = num1 * num2;
        System.out.println("Multiplication of int & float: "+
mul);

        double mul1 = num2 * num3;
        System.out.println("Multiplication of float & double:
"+ mul1);

        double mul2 = num1 * num3;
        System.out.println("Multiplication of int & double: "+
mul2);
        System.out.println("--------------------------------");


        //Division
        double div = num1 / num2;
        System.out.println("Division of int & float: "+ div);

        double div1 = num2 / num3;
        System.out.println("Division of float & double: "+
```

```
div1);

            double div2 = num1 / num3;
            System.out.println("Division of int & double: "+
div2);
            System.out.println("---------------------------------");

      }
}
```

OUTPUT:

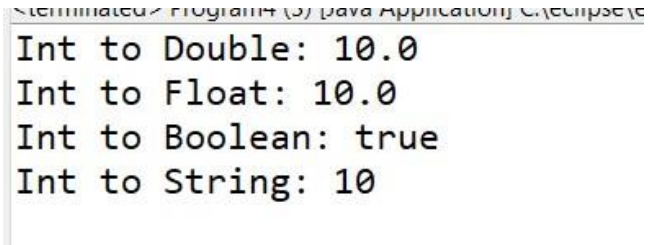


4) Write a Program that demonstrates widening conversion from int to (double,float, boolean, string) and prints the result.

CODE:

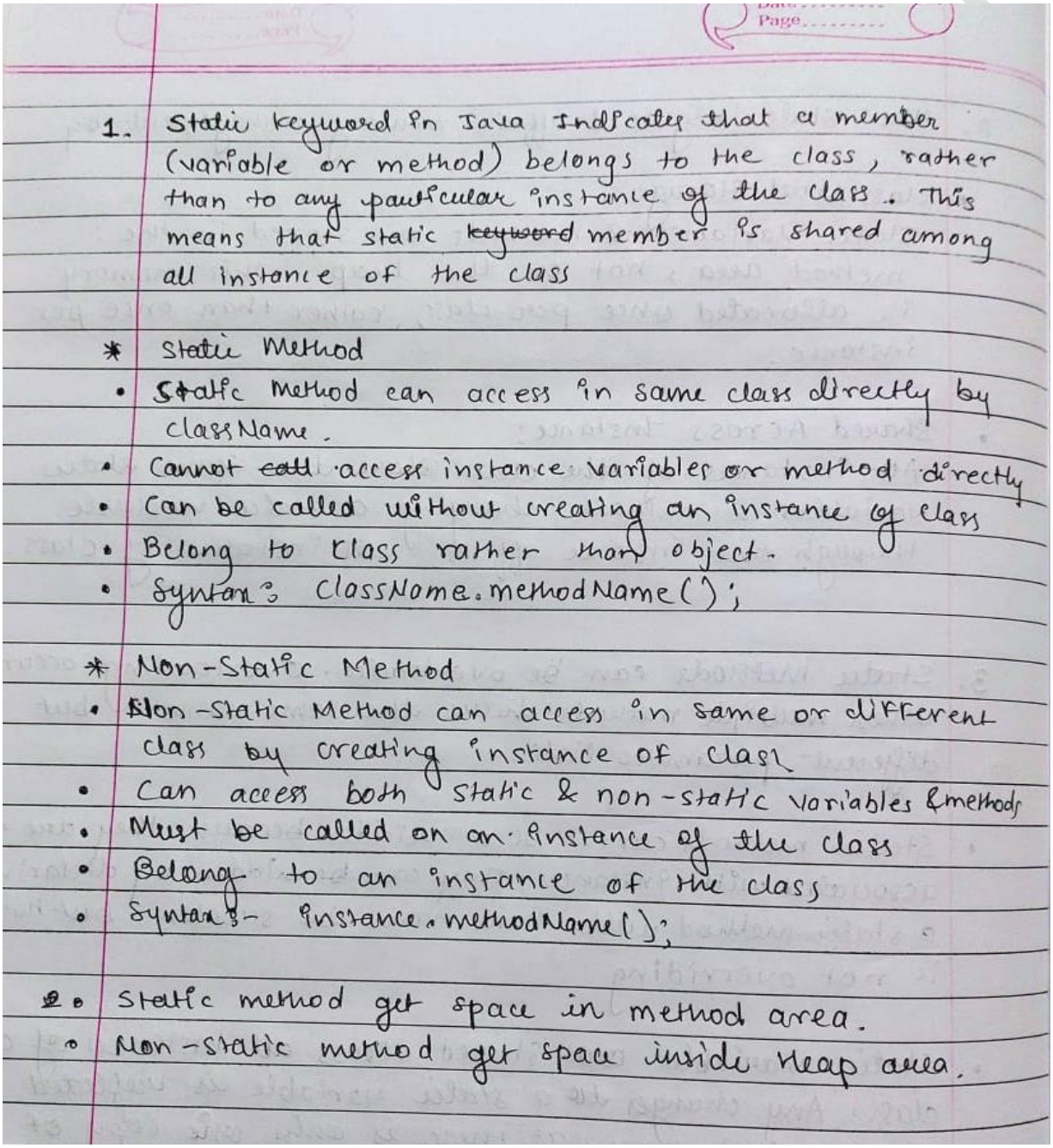
```
package org.example;
public class Program4 {
	public static void main(String[] args) {
		int num = 10;

		double num1 = (int) num;
		System.out.println("Int to Double: "+num1);

		float num2 = (int)num;
		System.out.println("Int to Float: "+num2);

		boolean num3 = (num != 0);
		System.out.println("Int to Boolean: "+num3);

		String str = Integer.toString(num);
		System.out.println("Int to String: "+num);
	}
}
```

OUTPUT:



INTERVIEW QUESTIONS

1. What does the static keyword mean in Java? Explain the difference between static and non-static methods.



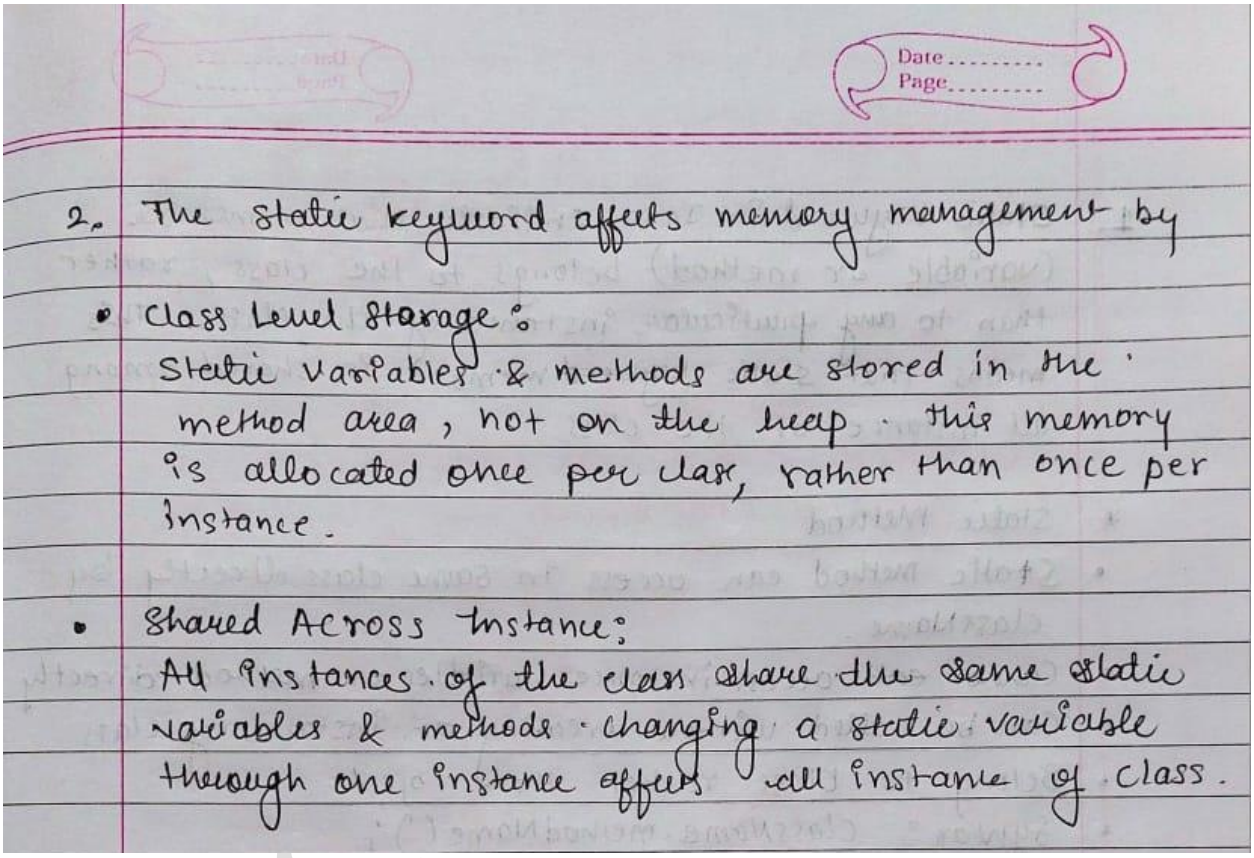
1. Static keyword in Java indicates that a member (variable or method) belongs to the class, rather than to any particular instance of the class. This means that static ~~keyword~~ member is shared among all instance of the class

* Static Method
- Static method can access in same class directly by className.
- Cannot ~~call~~ access instance variables or method directly
- Can be called without creating an instance of class
- Belong to class rather than object.
- Syntax: ClassName.methodName();

* Non-Static Method
- Non-Static Method can access in same or different class by creating instance of class
- Can access both static & non-static variables & methods
- Must be called on an instance of the class
- Belong to an instance of the class
- Syntax:- instance.methodName();

2. Static method get space in method area.
- Non-static method get space inside Heap area.

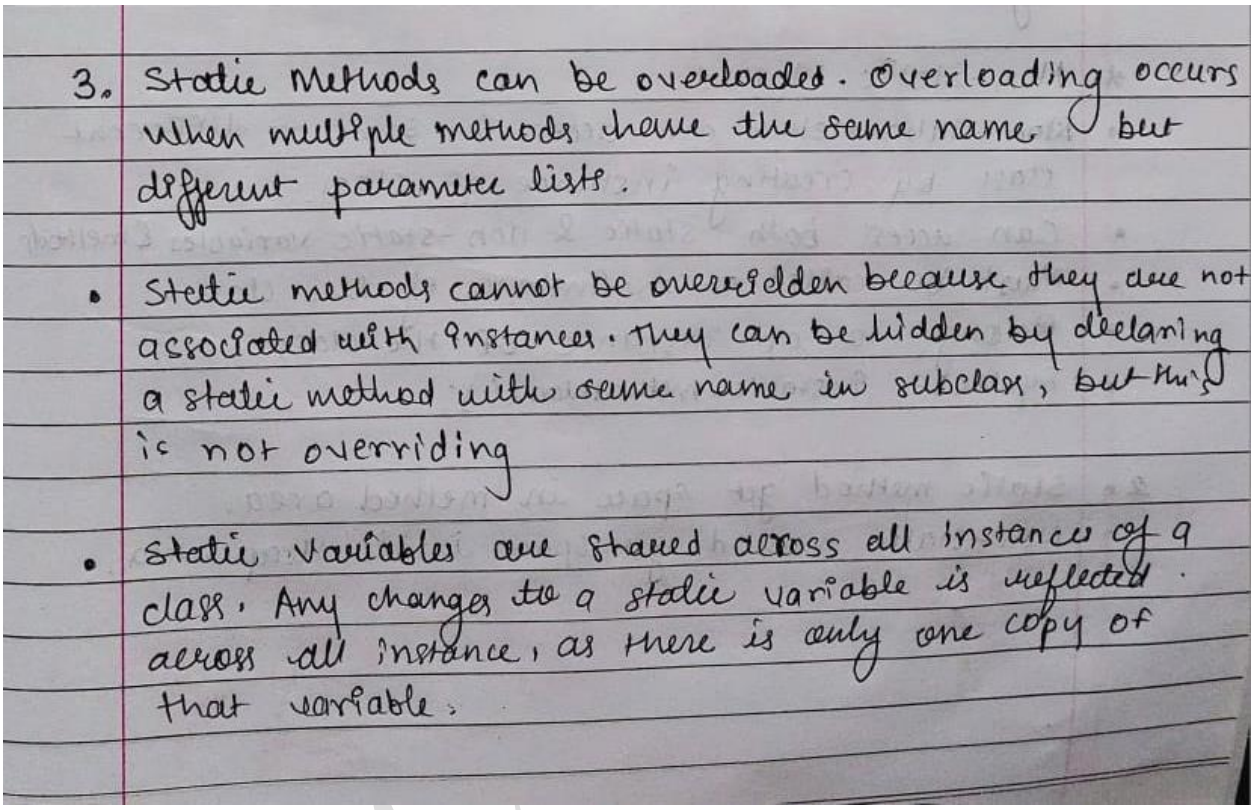
2. What is the role of the static keyword in the context of memory management.



2. The static keyword affects memory management by

- Class Level Storage:
  Static variables & methods are stored in the method area, not on the heap. This memory is allocated once per class, rather than once per instance.
- Shared Across Instance:
  All instances of the class share the same static variables & methods. Changing a static variable through one instance affects all instance of class.

3. Can static methods be overloaded and overridden in Java?Howstatic variables shared across multiple instances of a class?

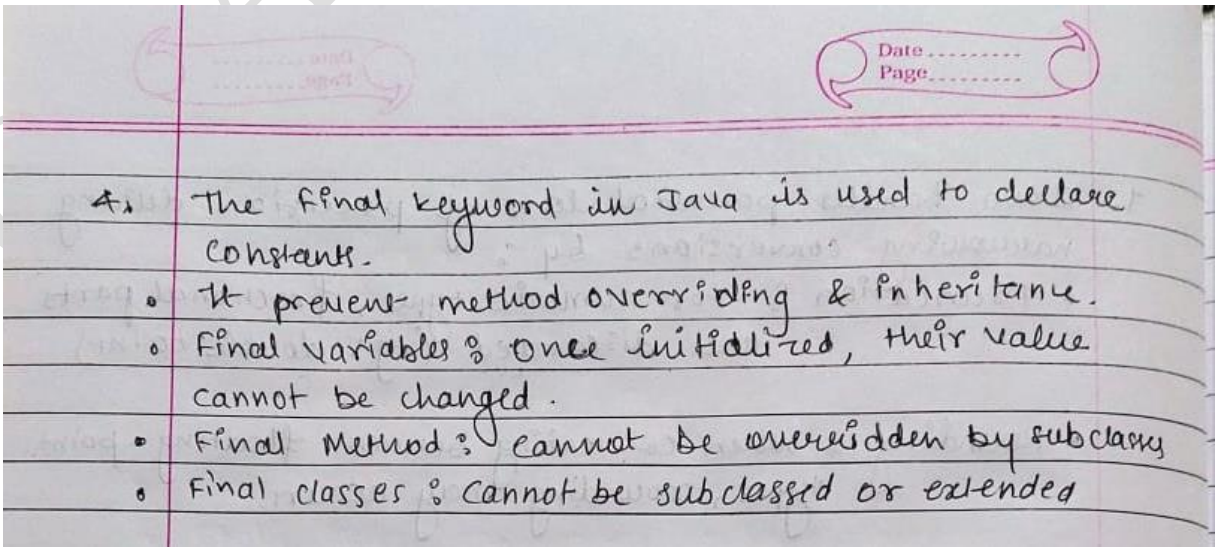


3. Static Methods can be overloaded. Overloading occurs when multiple methods have the same name but different parameter lists.

- Static methods cannot be overridden because they are not associated with instances. They can be hidden by declaring a static method with same name in subclass, but this is not overriding

- Static variables are shared across all instances of a class. Any changes to a static variable is reflected across all instance, as there is only one copy of that variable.

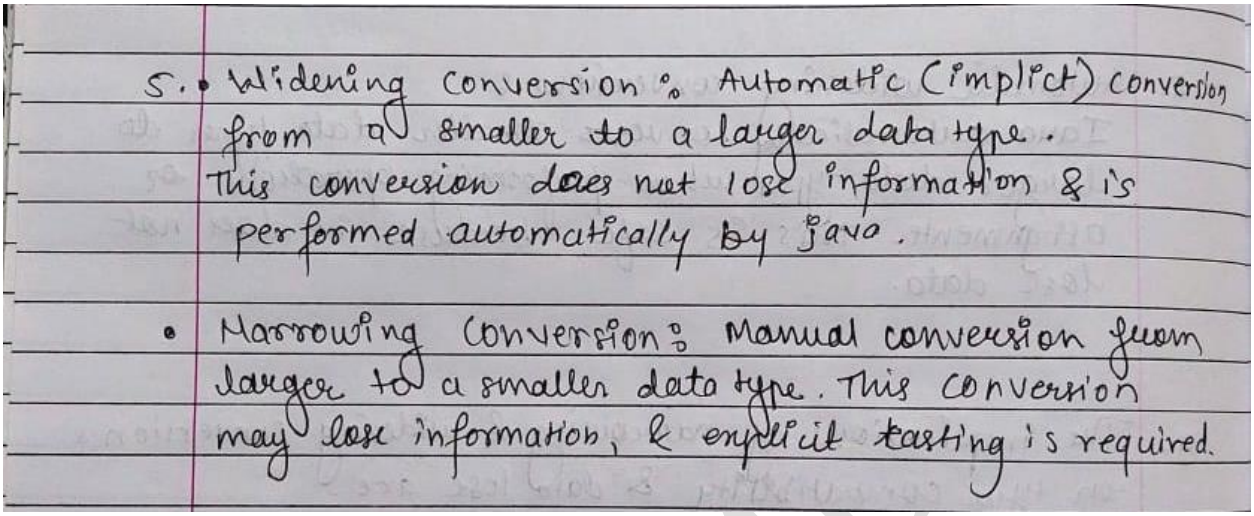
4. What is the significance of the final keyword in Java?



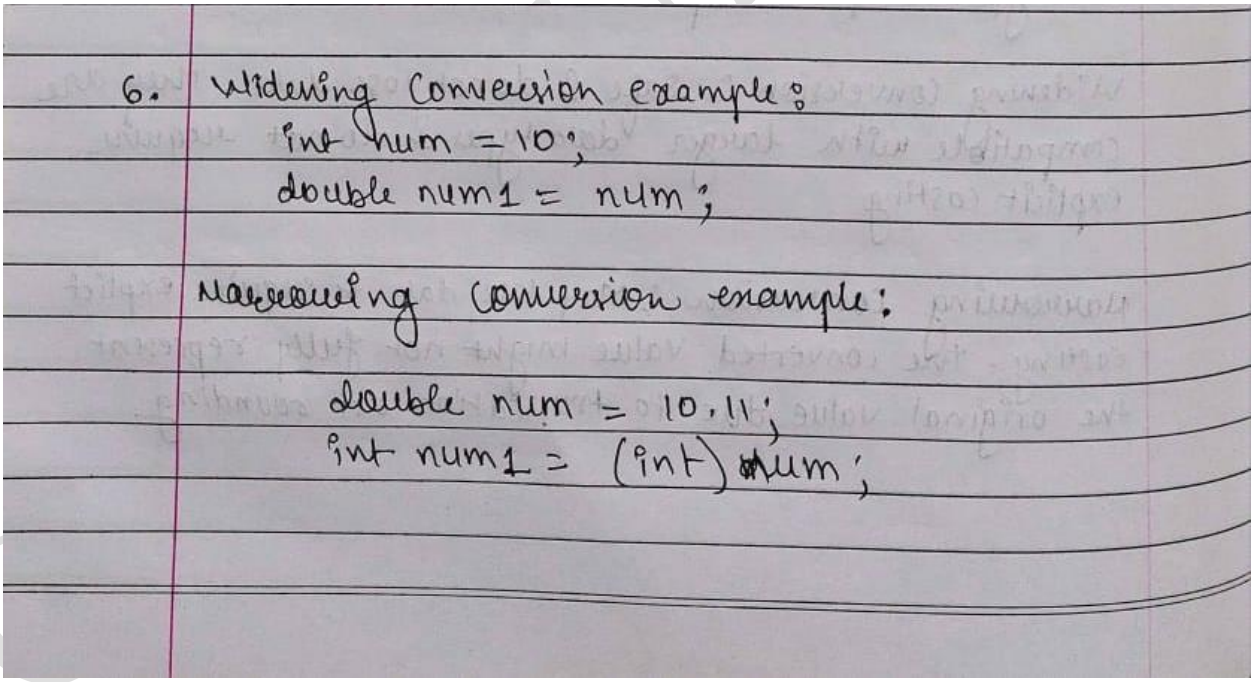
Date
Page

4. The final keyword in Java is used to declare constants.
- It prevent method overriding & inheritance.
- Final variables : Once initialized, their value cannot be changed.
- Final Method : cannot be overridden by subclass
- Final classes : Cannot be subclassed or extended

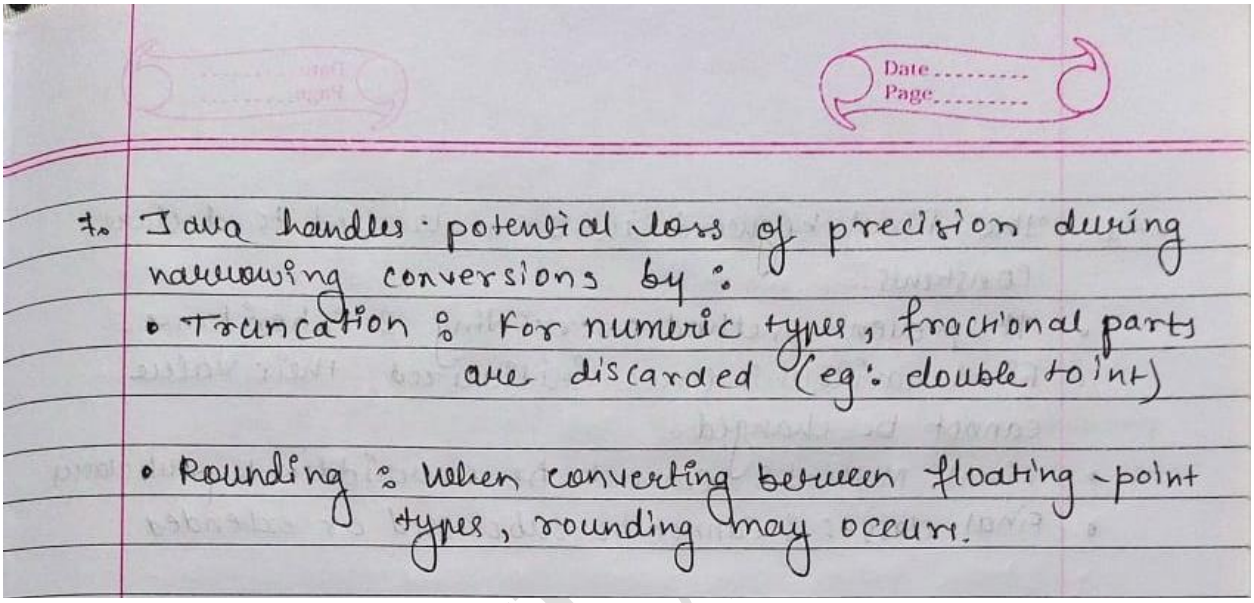
5. What are narrowing and widening conversions in Java?



6. Provide examples of narrowing and widening conversions between primitive data types.

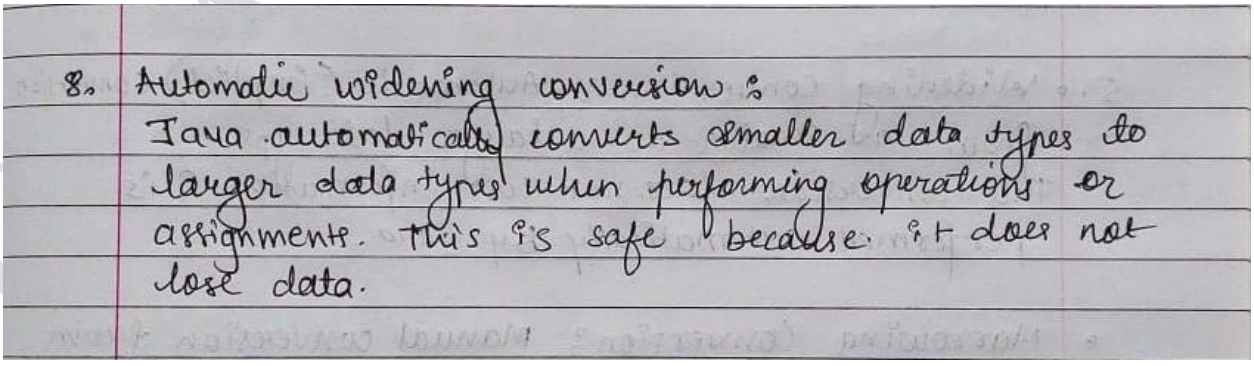


7. How does Java handle potential loss of precision during narrowing conversions?



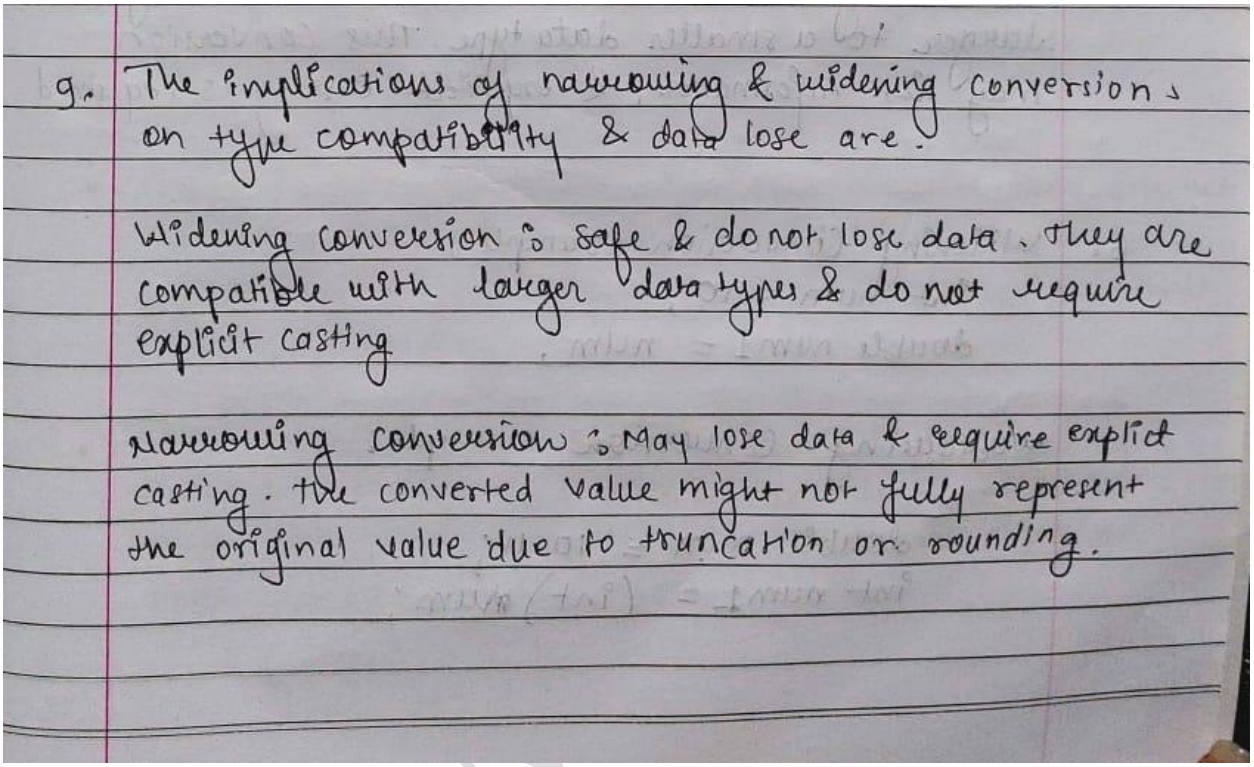
7. Java handles potential loss of precision during narrowing conversions by :
- Truncation : For numeric types, fractional parts are discarded (eg: double to int)
- Rounding : When converting between floating-point types, rounding may occurs.

8. Explain the concept of automatic widening conversion in Java.



8. Automatic widening conversion :
Java automatically converts smaller data types to larger data types when performing operations or assignments. This is safe because it does not lose data.

9. What are the implications of narrowing and widening conversions on type compatibility and data loss?



9. The implications of narrowing & widening conversions on type compatibility & data lose are:

Widening conversion : safe & do not lose data. They are compatible with larger data types & do not require explicit casting

Narrowing conversion : May lose data & require explict casting. the converted value might not fully represent the original value due to truncation or rounding.