**Note**: Consider the following before starting the assignment:

* A **static field** declared inside a class is called a **class-level variable**. To access this variable, use the class name and the dot operator (e.g., Integer.MAX\_VALUE).
* A **static method** defined inside a class is called a **class-level method**. To access this method, use the class name and the dot operator (e.g., Integer.parseInt()).
* When accessing static members within the same class, you do not need to use the class name.

#### ****1. Working with**** java.lang.Boolean

**a.** Explore the [Java API documentation for java.lang.Boolean](https://docs.oracle.com/javase/8/docs/api/java/lang/Boolean.html) and observe its modifiers and super types.

**b.** Declare a method-local variable status of type boolean with the value true and convert it to a String using the toString method. (Hint: Use Boolean.toString(Boolean) ).

**CODE:**

package org.programs;

public class Ques1b {

public static void main(String[] args) {

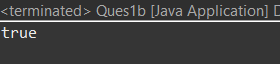
boolean status = true;

String statusString = Boolean.*toString*(status);

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

}

}



**c.** Declare a method-local variable strStatus of type String with the value "true" and convert it to a boolean using the parseBoolean method. (Hint: Use Boolean.parseBoolean(String)).

**CODE:**

package org.programs;

public class Ques1c {

public static void main(String[] args) {

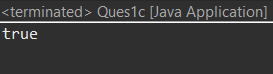
String strstatus = "true";

boolean status = Boolean.*parseBoolean*(strstatus);

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

}

}



**d.** Declare a method-local variable strStatus of type String with the value "1" or "0" and attempt to convert it to a boolean. (Hint: parseBoolean method will not work as expected with "1" or "0").

**CODE:**

package org.programs;

public class Ques1d {

public static void main(String[] args) {

String strstatus = "0";

boolean status = Boolean.*parseBoolean*(strstatus);

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

}

}



AND FOR “1” THE OUTPUT IS:



**e.** Declare a method-local variable status of type boolean with the value true and convert it to the corresponding wrapper class using Boolean.valueOf(). (Hint: Use Boolean.valueOf(boolean)).

**CODE:**

package org.programs;

public class Ques1e {

public static void main(String[] args) {

boolean status = true;

boolean wstatus = Boolean.*valueOf*(status);

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

}

}



**f.** Declare a method-local variable strStatus of type String with the value "true" and convert it to the corresponding wrapper class using Boolean.valueOf(). (Hint: Use Boolean.valueOf(String)).

**CODE:**

package org.programs;

public class Ques1g {

public static void main(String[] args) {

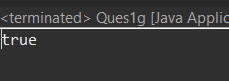
String strstatus = "true";

boolean status = Boolean.*valueOf*(strstatus);

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

}

}



**g.** Experiment with converting a boolean value into other primitive types or vice versa and observe the results.

package org.programs; //BOOLEAN TO INT

public class Ques1g {

public static void main(String[] args) {

boolean status = true;

int intValue = status ? 1 : 0;

System.***out***.println("Boolean true as int: " + intValue);

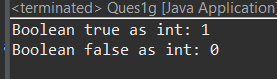
status = false;

intValue = status ? 1 : 0;

System.***out***.println("Boolean false as int: " + intValue);

}

}



//INT TO BOOLEAN

package org.programs;

public class Ques1gr {

public static void main(String[] args) {

int intValue = 1;

boolean status = intValue != 0;

System.***out***.println("Integer 1 as boolean: " + status);

intValue = 0;

status = intValue != 0;

System.***out***.println("Integer 0 as boolean: " + status);

}

}



**//BOOLEAN TO CHAR**

package org.programs;

public class Ques1gr2 {

public static void main(String[] args) {

boolean status = true;

char charValue = status ? 'T' : 'F';

System.***out***.println("Boolean true as char: " + charValue);

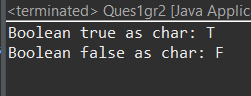
status = false;

charValue = status ? 'T' : 'F';

System.***out***.println("Boolean false as char: " + charValue);

}

}



**//CHAR TO BOOLEAN**

package org.programs;

public class Ques1gbc2 {

public static void main(String[] args) {

char charValue = 'T';

boolean status = charValue == 'T' || charValue == 't';

System.***out***.println("Character 'T' as boolean: " + status);

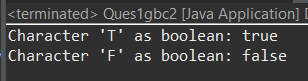
charValue = 'F';

status = charValue == 'T' || charValue == 't';

System.***out***.println("Character 'F' as boolean: " + status);

}

}



#### ****2. Working with**** java.lang.Byte

**a.** Explore the [Java API documentation for java.lang.Byte](https://docs.oracle.com/javase/8/docs/api/java/lang/Byte.html) and observe its modifiers and super types.

**b.** Write a program to test how many bytes are used to represent a byte value using the BYTES field. (Hint: Use Byte.BYTES).

package org.programs;

public class Ques2b {

public static void main(String[] args) {

System.***out***.println("Number of bytes used to represent a byte value: " + Byte.***BYTES***);

}

}



**c.** Write a program to find the minimum and maximum values of byte using the MIN\_VALUE and MAX\_VALUE fields. (Hint: Use Byte.MIN\_VALUE and Byte.MAX\_VALUE).

package org.programs;

public class Ques2c {

public static void main(String[] args) {

System.***out***.println("Minimum value of byte: " + Byte.***MIN\_VALUE***);

System.***out***.println("Maximum value of byte: " + Byte.***MAX\_VALUE***);

}

}



**d.** Declare a method-local variable number of type byte with some value and convert it to a String using the toString method. (Hint: Use Byte.toString(byte)).

package org.programs;

public class Ques2d {

public static void main(String[] args) {

byte number = 20;

String strnumber = Byte.*toString*(number);

System.***out***.println("String representation of byte value: " + strnumber);

}

}



**e.** Declare a method-local variable strNumber of type String with some value and convert it to a byte value using the parseByte method. (Hint: Use Byte.parseByte(String)).

package org.programs;

public class Ques2e {

public static void main(String[] args) {

String strnumber = "123";

byte number = Byte.*parseByte*(strnumber);

System.***out***.println("Byte value of string '" + strnumber + "': " + number);

}

}



**f.** Declare a method-local variable strNumber of type String with the value "Ab12Cd3" and attempt to convert it to a byte value. (Hint: parseByte method will throw a NumberFormatException).

package org.programs;

public class Ques2f {

public static void main(String[] args) {

String strNumber = "Ab12Cd3";

try {

byte number = Byte.*parseByte*(strNumber);

System.***out***.println("Byte value of string '" + strNumber + "': " + number);

} catch (NumberFormatException e) {

System.***out***.println("NumberFormatException occurred: " + e.getMessage());

}

}

}



**g.** Declare a method-local variable number of type byte with some value and convert it to the corresponding wrapper class using Byte.valueOf(). (Hint: Use Byte.valueOf(byte)).

package org.programs;

public class Ques2g {

public static void main(String[] args) {

byte number = 20;

Byte byteObj = Byte.*valueOf*(number);

System.***out***.println("Byte wrapper object: " + byteObj);

}

}



**h.** Declare a method-local variable strNumber of type String with some byte value and convert it to the corresponding wrapper class using Byte.valueOf(). (Hint: Use Byte.valueOf(String)).

package org.programs;

public class Ques2h {

public static void main(String[] args) {

String strNumber = "100";

Byte byteObj = Byte.*valueOf*(strNumber);

System.***out***.println("Byte wrapper object from string '" + strNumber + "': " + byteObj);

}

}



1. Experiment with converting a byte value into other primitive types or vice versa and observe the results.

package org.programs;

public class Ques2i {

public static void main(String[] args) {

byte byteValue = 100;

short shortValue = byteValue;

int intValue = byteValue;

long longValue = byteValue;

float floatValue = byteValue;

double doubleValue = byteValue;

System.out.println("byte to short: " + shortValue);

System.out.println("byte to int: " + intValue);

System.out.println("byte to long: " + longValue);

System.out.println("byte to float: " + floatValue);

System.out.println("byte to double: " + doubleValue);

short s = 50;

byte byteFromShort = (byte) s;

int i = 200;

byte byteFromInt = (byte) i;

long l = 300L;

byte byteFromLong = (byte) l;

float floatNum = 123.45f;

byte byteFromFloat = (byte) floatNum;

double doubleNum = 255.99;

byte byteFromDouble = (byte) doubleNum;

System.out.println("short to byte: " + byteFromShort);

System.out.println("int to byte: " + byteFromInt);

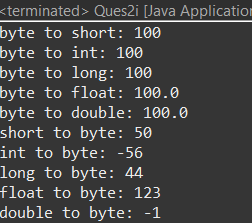
System.out.println("long to byte: " + byteFromLong);

System.out.println("float to byte: " + byteFromFloat);

System.out.println("double to byte: " + byteFromDouble);

}

}



#### ****3. Working with**** java.lang.Short

**a.** Explore the [Java API documentation for java.lang.Short](https://docs.oracle.com/javase/8/docs/api/java/lang/Short.html) and observe its modifiers and super types.

**b.** Write a program to test how many bytes are used to represent a short value using the BYTES field. (Hint: Use Short.BYTES).

package org.programs;

public class Ques3b {

public static void main(String[] args) {

System.***out***.println("Number of bytes used to represent a short value: " + Short.***BYTES***);

}

}



**c.** Write a program to find the minimum and maximum values of short using the MIN\_VALUE and MAX\_VALUE fields. (Hint: Use Short.MIN\_VALUE and Short.MAX\_VALUE).

package org.programs;

public class Ques3c {

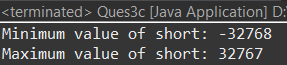
public static void main(String[] args) {

System.***out***.println("Minimum value of short: " + Short.***MIN\_VALUE***);

System.***out***.println("Maximum value of short: " + Short.***MAX\_VALUE***);

}

}



**d.** Declare a method-local variable number of type short with some value and convert it to a String using the toString method. (Hint: Use Short.toString(short)).

package org.programs;

public class Ques3d {

public static void main(String[] args) {

short number = 12345;

String strNumber = Short.*toString*(number);

System.***out***.println("String representation of short value: " + strNumber);

}

}



**e.** Declare a method-local variable strNumber of type String with some value and convert it to a short value using the parseShort method. (Hint: Use Short.parseShort(String)).

package org.programs;

public class Ques3e {

public static void main(String[] args) {

String strNumber = "12345";

short number = Short.*parseShort*(strNumber);

System.***out***.println("Short value of string '" + strNumber + "': " + number);

}

}



**f.** Declare a method-local variable strNumber of type String with the value "Ab12Cd3" and attempt to convert it to a short value. (Hint: parseShort method will throw a NumberFormatException).

package org.programs;

public class Ques3f {

public static void main(String[] args) {

String strNumber = "Ab12Cd3";

try {

short number = Short.*parseShort*(strNumber);

System.***out***.println("Short value of string '" + strNumber + "': " + number);

} catch (NumberFormatException e) {

System.***out***.println("NumberFormatException occurred: " + e.getMessage());

}

}

}



**g.** Declare a method-local variable number of type short with some value and convert it to the corresponding wrapper class using Short.valueOf(). (Hint: Use Short.valueOf(short)).

package org.programs;

public class Ques3g {

public static void main(String[] args) {

short number = 12345;

Short shortObj = Short.*valueOf*(number);

System.***out***.println("Short wrapper object: " + shortObj);

}

}



**h.** Declare a method-local variable strNumber of type String with some short value and convert it to the corresponding wrapper class using Short.valueOf(). (Hint: Use Short.valueOf(String)).

package org.programs;

public class Ques3h {

public static void main(String[] args) {

String strNumber = "12345";

Short shortObj = Short.*valueOf*(strNumber);

System.***out***.println("Short wrapper object from string '" + strNumber + "': " + shortObj);

}

}



1. Experiment with converting a short value into other primitive types or vice versa and observe the results.

package org.programs;

public class Ques3i {

public static void main(String[] args) {

short shortValue = 1000;

int intValue = shortValue;

long longValue = shortValue;

float floatValue = shortValue;

double doubleValue = shortValue;

System.out.println("short to int: " + intValue);

System.out.println("short to long: " + longValue);

System.out.println("short to float: " + floatValue);

System.out.println("short to double: " + doubleValue);

int i = 2000;

short shortFromInt = (short) i;

long l = 3000L;

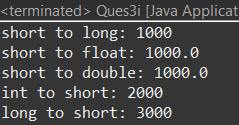
short shortFromLong = (short) l;

System.out.println("int to short: " + shortFromInt);

System.out.println("long to short: " + shortFromLong);

}

}



#### ****4. Working with**** java.lang.Integer

**a.** Explore the [Java API documentation for java.lang.Integer](https://docs.oracle.com/javase/8/docs/api/java/lang/Integer.html) and observe its modifiers and super types.

**b.** Write a program to test how many bytes are used to represent an int value using the BYTES field. (Hint: Use Integer.BYTES).

package org.programs;

public class Ques4b {

public static void main(String[] args) {

System.***out***.println("Number of bytes used to represent an int value: " + Integer.***BYTES***);

}

}



**c.** Write a program to find the minimum and maximum values of int using the MIN\_VALUE and MAX\_VALUE fields. (Hint: Use Integer.MIN\_VALUE and Integer.MAX\_VALUE).

package org.programs;

public class Ques4c {

public static void main(String[] args) {

System.***out***.println("Minimum value of int: " + Integer.***MIN\_VALUE***);

System.***out***.println("Maximum value of int: " + Integer.***MAX\_VALUE***);

}

}



**d.** Declare a method-local variable number of type int with some value and convert it to a String using the toString method. (Hint: Use Integer.toString(int)).

package org.programs;

public class Ques4d {

public static void main(String[] args) {

int number = 123456;

String numberStr = Integer.*toString*(number);

System.***out***.println("String representation of int value: " + numberStr);

}

}



**e.** Declare a method-local variable strNumber of type String with some value and convert it to an int value using the parseInt method. (Hint: Use Integer.parseInt(String)).

package org.programs;

public class Ques4e {

public static void main(String[] args) {

String strNumber = "98765";

int number = Integer.*parseInt*(strNumber);

System.***out***.println("Int value of string '" + strNumber + "': " + number);

}

}



**f.** Declare a method-local variable strNumber of type String with the value "Ab12Cd3" and attempt to convert it to an int value. (Hint: parseInt method will throw a NumberFormatException).

package org.programs;

public class Ques4f {

public static void main(String[] args) {

String strNumber = "Ab12Cd3";

try {

int number = Integer.*parseInt*(strNumber);

System.***out***.println("Int value of string '" + strNumber + "': " + number);

} catch (NumberFormatException e) {

System.***out***.println("NumberFormatException occurred: " + e.getMessage());

}

}

}



**g.** Declare a method-local variable number of type int with some value and convert it to the corresponding wrapper class using Integer.valueOf(). (Hint: Use Integer.valueOf(int)).

package org.programs;

public class Ques4g {

public static void main(String[] args) {

int number = 123456;

Integer integerObj = Integer.*valueOf*(number);

System.***out***.println("Integer wrapper object: " + integerObj);

}

}



**h.** Declare a method-local variable strNumber of type String with some integer value and convert it to the corresponding wrapper class using Integer.valueOf(). (Hint: Use Integer.valueOf(String)).

package org.programs;

public class Ques4h {

public static void main(String[] args) {

String strNumber = "123456";

Integer integerObj = Integer.*valueOf*(strNumber);

System.***out***.println("Integer wrapper object from String '" + strNumber + "': " + integerObj);

}

}



1. Declare two integer variables with values 10 and 20, and add them using a method from the Integer class. (Hint: Use Integer.sum(int, int)).

package org.programs;

public class Ques4i {

public static void main(String[] args) {

int a = 10;

int b = 20;

int sum = Integer.sum(a, b);

System.out.println("Sum of " + a + " and " + b + ": " + sum);

}

}



**j.** Declare two integer variables with values 10 and 20, and find the minimum and maximum values using the Integer class. (Hint: Use Integer.min(int, int) and Integer.max(int, int)).

package org.programs;

public class Ques4j {

public static void main(String[] args) {

int a = 10;

int b = 20;

int minValue = Integer.*min*(a, b);

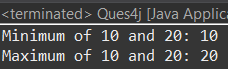
int maxValue = Integer.*max*(a, b);

System.***out***.println("Minimum of " + a + " and " + b + ": " + minValue);

System.***out***.println("Maximum of " + a + " and " + b + ": " + maxValue);

}

}



**k.** Declare an integer variable with the value 7. Convert it to binary, octal, and hexadecimal strings using methods from the Integer class. (Hint: Use Integer.toBinaryString(int), Integer.toOctalString(int), and Integer.toHexString(int)).

package org.programs;

public class Ques4k {

public static void main(String[] args) {

int number = 7;

String binaryString = Integer.*toBinaryString*(number); // Convert int to octal string

String octalString = Integer.*toOctalString*(number);

String hexString = Integer.*toHexString*(number);

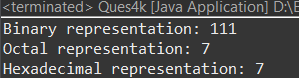
System.***out***.println("Binary representation: " + binaryString);

System.***out***.println("Octal representation: " + octalString);

System.***out***.println("Hexadecimal representation: " + hexString);

}

}



**l.** Experiment with converting an int value into other primitive types or vice versa and observe the results.

package org.programs;

public class Ques4l {

public static void main(String[] args) {

int intValue = 1000;

byte byteValue = (byte) intValue;

short shortValue = (short) intValue;

long longValue = intValue;

float floatValue = intValue;

double doubleValue = intValue;

System.out.println("int to byte: " + byteValue);

System.out.println("int to short: " + shortValue);

System.out.println("int to long: " + longValue);

System.out.println("int to float: " + floatValue);

System.out.println("int to double: " + doubleValue);

byte b = 50;

int intFromByte = b;

short s = 3000;

int intFromShort = s;

long l = 5000L;

int intFromLong = (int) l;

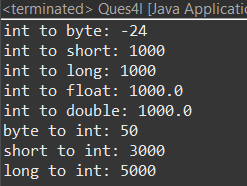
System.out.println("byte to int: " + intFromByte);

System.out.println("short to int: " + intFromShort);

System.out.println("long to int: " + intFromLong);

}

}



#### ****5. Working with**** java.lang.Long

**a.** Explore the [Java API documentation for java.lang.Long](https://docs.oracle.com/javase/8/docs/api/java/lang/Long.html) and observe its modifiers and super types.

**b.** Write a program to test how many bytes are used to represent a long value using the BYTES field. (Hint: Use Long.BYTES).

package org.programs;

public class Ques5b {

public static void main(String[] args) {

System.***out***.println("Number of bytes used to represent a long value: " + Long.***BYTES***);

}

}



**c.** Write a program to find the minimum and maximum values of long using the MIN\_VALUE and MAX\_VALUE fields. (Hint: Use Long.MIN\_VALUE and Long.MAX\_VALUE).

package org.programs;

public class Ques5c {

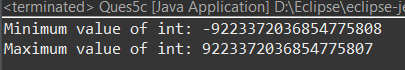
public static void main(String[] args) {

System.***out***.println("Minimum value of int: " + Long.***MIN\_VALUE***);

System.***out***.println("Maximum value of int: " + Long.***MAX\_VALUE***);

}

}



**d.** Declare a method-local variable number of type long with some value and convert it to a String using the toString method. (Hint: Use Long.toString(long)).

package org.programs;

public class Ques5d {

public static void main(String[] args) {

Long number = 123456789l;

String numberStr = Long.*toString*(number);

System.***out***.println("String representation of Long value: " + numberStr);

}

}



**e.** Declare a method-local variable strNumber of type String with some value and convert it to a long value using the parseLong method. (Hint: Use Long.parseLong(String)).

package org.programs;

public class Ques5e {

public static void main(String[] args) {

String strNumber = "9876543210";

long number = Long.*parseLong*(strNumber);

System.***out***.println("Long value of string '" + strNumber + "': " + number);

}

}



**f.** Declare a method-local variable strNumber of type String with the value "Ab12Cd3" and attempt to convert it to a long value. (Hint: parseLong method will throw a NumberFormatException).

package org.programs;

public class Ques5f {

public static void main(String[] args) {

String strNumber = "Ab12Cd3";

try {

long number = Long.*parseLong*(strNumber);

System.***out***.println("Long value of string '" + strNumber + "': " + number);

} catch (NumberFormatException e) {

System.***out***.println("NumberFormatException occurred: " + e.getMessage());

}

}

}



**g.** Declare a method-local variable number of type long with some value and convert it to the corresponding wrapper class using Long.valueOf(). (Hint: Use Long.valueOf(long)).

package org.programs;

public class Ques5g {

public static void main(String[] args) {

long number = 123456789L;

Long longObj = Long.*valueOf*(number);

System.***out***.println("Long wrapper object: " + longObj);

}

}



**h.** Declare a method-local variable strNumber of type String with some long value and convert it to the corresponding wrapper class using Long.valueOf(). (Hint: Use Long.valueOf(String)).

package org.programs;

public class Ques5h {

public static void main(String[] args) {

String strNumber = "1234567890";

Long longObj = Long.*valueOf*(strNumber);

System.***out***.println("Long wrapper object from string '" + strNumber + "': " + longObj);

}

}



**i.** Declare two long variables with values 1123 and 9845, and add them using a method from the Long class. (Hint: Use Long.sum(long, long)).

package org.programs;

public class Ques5i {

public static void main(String[] args) {

long a = 1123L;

long b = 9845L;

long sum = Long.*sum*(a, b);

System.***out***.println("Sum of " + a + " and " + b + ": " + sum);

}

}

****

**j.** Declare two long variables with values 1122 and 5566, and find the minimum and maximum values using the Long class. (Hint: Use Long.min(long, long) and Long.max(long, long)).

package org.programs;

public class Ques5j {

public static void main(String[] args) {

long a = 1122L;

long b = 5566L;

long minValue = Long.*min*(a, b);

long maxValue = Long.*max*(a, b);

System.***out***.println("Minimum of " + a + " and " + b + ": " + minValue);

System.***out***.println("Maximum of " + a + " and " + b + ": " + maxValue);

}

}



**k.** Declare a long variable with the value 7. Convert it to binary, octal, and hexadecimal strings using methods from the Long class. (Hint: Use Long.toBinaryString(long), Long.toOctalString(long), and Long.toHexString(long)).

package org.programs;

public class Ques5k {

public static void main(String[] args) {

long number = 7L;

String binaryString = Long.*toBinaryString*(number);

String octalString = Long.*toOctalString*(number);

String hexString = Long.*toHexString*(number);

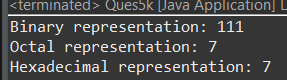
System.***out***.println("Binary representation: " + binaryString);

System.***out***.println("Octal representation: " + octalString);

System.***out***.println("Hexadecimal representation: " + hexString);

}

}



**l.** Experiment with converting a long value into other primitive types or vice versa and observe the results.

package org.programs;

public class Ques5l {

public static void main(String[] args) {

long longValue = 100000L;

int intValue = (int) longValue;

short shortValue = (short) longValue;

byte byteValue = (byte) longValue;

float floatValue = longValue;

double doubleValue = longValue;

System.***out***.println("long to int: " + intValue);

System.***out***.println("long to short: " + shortValue);

System.***out***.println("long to byte: " + byteValue);

System.***out***.println("long to float: " + floatValue);

System.***out***.println("long to double: " + doubleValue);

int i = 2000;

long longFromInt = i;

short s = 3000;

long longFromShort = s;

byte b = 100;

long longFromByte = b;

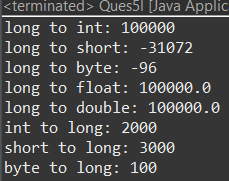
System.***out***.println("int to long: " + longFromInt);

System.***out***.println("short to long: " + longFromShort);

System.***out***.println("byte to long: " + longFromByte);

}

}



#### ****6. Working with**** java.lang.Float

**a.** Explore the [Java API documentation for java.lang.Float](https://docs.oracle.com/javase/8/docs/api/java/lang/Float.html) and observe its modifiers and super types.

**b.** Write a program to test how many bytes are used to represent a float value using the BYTES field. (Hint: Use Float.BYTES).

package org.programs;

public class Ques6b {

public static void main(String[] args) {

System.***out***.println("Number of bytes used to represent a float value: " + Float.***BYTES***);

}

}



**c.** Write a program to find the minimum and maximum values of float using the MIN\_VALUE and MAX\_VALUE fields. (Hint: Use Float.MIN\_VALUE and Float.MAX\_VALUE).

package org.programs;

public class Ques6c {

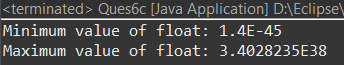
public static void main(String[] args) {

System.***out***.println("Minimum value of int: " + Float.***MIN\_VALUE***);

System.***out***.println("Maximum value of int: " + Float.***MAX\_VALUE***);

}

}



**d.** Declare a method-local variable number of type float with some value and convert it to a String using the toString method. (Hint: Use Float.toString(float)).

package org.programs;

public class Ques6d {

public static void main(String[] args) {

float number = 123.456f;

String numberStr = Float.*toString*(number);

System.***out***.println("String representation of float value: " + numberStr);

}

}



**e.** Declare a method-local variable strNumber of type String with some value and convert it to a float value using the parseFloat method. (Hint: Use Float.parseFloat(String)).

package org.programs;

public class Ques6e {

public static void main(String[] args) {

String strNumber = "987.65";

float number = Float.*parseFloat*(strNumber);

System.***out***.println("Float value of string '" + strNumber + "': " + number);

}

}



**f.** Declare a method-local variable strNumber of type String with the value "Ab12Cd3" and attempt to convert it to a float value. (Hint: parseFloat method will throw a NumberFormatException).

package org.programs;

public class Ques6f {

public static void main(String[] args) {

String strNumber = "Ab12Cd3";

try {

float number = Float.*parseFloat*(strNumber);

System.***out***.println("Float value of string '" + strNumber + "': " + number);

} catch (NumberFormatException e) {

System.***out***.println("NumberFormatException occurred: " + e.getMessage());

}

}

}



**g.** Declare a method-local variable number of type float with some value and convert it to the corresponding wrapper class using Float.valueOf(). (Hint: Use Float.valueOf(float)).

package org.programs;

public class Ques6g {

public static void main(String[] args) {

float number = 123.456f;

Float floatObj = Float.*valueOf*(number);

System.***out***.println("Float wrapper object: " + floatObj);

}

}



**h.** Declare a method-local variable strNumber of type String with some float value and convert it to the corresponding wrapper class using Float.valueOf(). (Hint: Use Float.valueOf(String)).

package org.programs;

public class Ques6h {

public static void main(String[] args) {

String strNumber = "123.456";

Float floatObj = Float.*valueOf*(strNumber);

System.***out***.println("Float wrapper object from string '" + strNumber + "': " + floatObj);

}

}



**i.** Declare two float variables with values 112.3 and 984.5, and add them using a method from the Float class. (Hint: Use Float.sum(float, float)).

package org.programs;

public class Ques6i {

public static void main(String[] args) {

float a = 112.3f;

float b = 984.5f;

float sum = Float.*sum*(a, b);

System.***out***.println("Sum of " + a + " and " + b + ": " + sum);

}

}

****

**j.** Declare two float variables with values 112.2 and 556.6, and find the minimum and maximum values using the Float class. (Hint: Use Float.min(float, float) and Float.max(float, float)).

package org.programs;

public class Ques6j {

public static void main(String[] args) {

float a = 112.2f;

float b = 556.6f;

float minValue = Float.*min*(a, b);

float maxValue = Float.*max*(a, b);

System.***out***.println("Minimum of " + a + " and " + b + ": " + minValue);

System.***out***.println("Maximum of " + a + " and " + b + ": " + maxValue);

}

}



**k.** Declare a float variable with the value -25.0f. Find the square root of this value. (Hint: Use Math.sqrt() method).

package org.programs;

public class Ques6k {

public static void main(String[] args) {

float number = -25.0f;

// Math.sqrt() only works with positive values; for negative values, it returns NaN (Not a Number)

double sqrt = Math.*sqrt*(number);

System.***out***.println("Square root of " + number + ": " + sqrt);

}

}



**l.** Declare two float variables with the same value, 0.0f, and divide them. (Hint: Observe the result and any special floating-point behavior).

package org.programs;

public class Ques6l {

public static void main(String[] args) {

float a = 0.0f;

float b = 0.0f;

float result = a / b;

System.***out***.println("Result of dividing " + a + " by " + b + ": " + result);

}

}



**m.** Experiment with converting a float value into other primitive types or vice versa and observe the results.

package org.programs;

public class Ques6m {

public static void main(String[] args) {

float floatValue = 1234.56f;

int intValue = (int) floatValue;

long longValue = (long) floatValue;

double doubleValue = floatValue;

System.***out***.println("float to int: " + intValue);

System.***out***.println("float to long: " + longValue);

System.***out***.println("float to double: " + doubleValue);

int i = 2000;

float floatFromInt = i;

long l = 3000L;

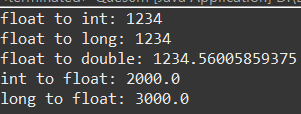
float floatFromLong = (float) l;

System.***out***.println("int to float: " + floatFromInt);

System.***out***.println("long to float: " + floatFromLong);

}

}



#### ****7. Working with**** java.lang.Double

**a.** Explore the [Java API documentation for java.lang.Double](https://docs.oracle.com/javase/8/docs/api/java/lang/Double.html) and observe its modifiers and super types.

**b.** Write a program to test how many bytes are used to represent a double value using the BYTES field. (Hint: Use Double.BYTES).

package org.programs;

public class Ques7b {

public static void main(String[] args) {

System.***out***.println("Number of bytes used to represent a float value: " + Double.***BYTES***);

}

}



**c.** Write a program to find the minimum and maximum values of double using the MIN\_VALUE and MAX\_VALUE fields. (Hint: Use Double.MIN\_VALUE and Double.MAX\_VALUE).

package org.programs;

public class Ques7c {

public static void main(String[] args) {

System.***out***.println("Minimum value of double: " + Double.***MIN\_VALUE***);

System.***out***.println("Maximum value of double: " + Double.***MAX\_VALUE***);

}

}



**d.** Declare a method-local variable number of type double with some value and convert it to a String using the toString method. (Hint: Use Double.toString(double)).

package org.programs;

public class Ques7d {

public static void main(String[] args) {

double number = 123.456;

String numberStr = Double.*toString*(number);

System.***out***.println("String representation of double value: " + numberStr);

}

}



**e.** Declare a method-local variable strNumber of type String with some value and convert it to a double value using the parseDouble method. (Hint: Use Double.parseDouble(String)).

package org.programs;

public class Ques7e {

public static void main(String[] args) {

String strNumber = "987.654";

double number = Double.*parseDouble*(strNumber);

System.***out***.println("Double value of string '" + strNumber + "': " + number);

}

}



**f.** Declare a method-local variable strNumber of type String with the value "Ab12Cd3" and attempt to convert it to a double value. (Hint: parseDouble method will throw a NumberFormatException).

package org.programs;

public class Ques7f {

public static void main(String[] args) {

String strNumber = "Ab12Cd3";

try {

double number = Double.*parseDouble*(strNumber);

System.***out***.println("Double value of string '" + strNumber + "': " + number);

} catch (NumberFormatException e) {

System.***out***.println("NumberFormatException occurred: " + e.getMessage());

}

}

}



**g.** Declare a method-local variable number of type double with some value and convert it to the corresponding wrapper class using Double.valueOf(). (Hint: Use Double.valueOf(double)).

package org.programs;

public class Ques7g {

public static void main(String[] args) {

double number = 123.456;

Double doubleObj = Double.*valueOf*(number);

System.***out***.println("Double wrapper object: " + doubleObj);

}

}



**h.** Declare a method-local variable strNumber of type String with some double value and convert it to the corresponding wrapper class using Double.valueOf(). (Hint: Use Double.valueOf(String)).

package org.programs;

public class Ques7h {

public static void main(String[] args) {

String strNumber = "123.456";

Double doubleObj = Double.*valueOf*(strNumber);

System.***out***.println("Double wrapper object from string '" + strNumber + "': " + doubleObj);

}

}



**i.** Declare two double variables with values 112.3 and 984.5, and add them using a method from the Double class. (Hint: Use Double.sum(double, double)).

package org.programs;

public class Ques7i {

public static void main(String[] args) {

double a = 112.3;

double b = 984.5;

double sum = Double.*sum*(a, b);

System.***out***.println("Sum of " + a + " and " + b + ": " + sum);

}

}

****

**j.** Declare two double variables with values 112.2 and 556.6, and find the minimum and maximum values using the Double class. (Hint: Use Double.min(double, double) and Double.max(double, double)).

package org.programs;

public class Ques7j {

public static void main(String[] args) {

double a = 112.2;

double b = 556.6;

double minValue = Double.*min*(a, b);

double maxValue = Double.*max*(a, b);

System.***out***.println("Minimum of " + a + " and " + b + ": " + minValue);

System.***out***.println("Maximum of " + a + " and " + b + ": " + maxValue);

}

}



**k.** Declare a double variable with the value -25.0. Find the square root of this value. (Hint: Use Math.sqrt() method).

package org.programs;

public class Ques7k {

public static void main(String[] args) {

double number = -25.0;

// Math.sqrt() only works with positive values; for negative values, it returns NaN (Not a Number)

double sqrt = Math.*sqrt*(number);

System.***out***.println("Square root of " + number + ": " + sqrt);

}

}



**l.** Declare two double variables with the same value, 0.0, and divide them. (Hint: Observe the result and any special floating-point behavior).

package org.programs;

public class Ques7l {

public static void main(String[] args) {

double a = 0.0;

double b = 0.0;

double result = a / b;

System.***out***.println("Result of dividing " + a + " by " + b + ": " + result);

}

}



**m.** Experiment with converting a double value into other primitive types or vice versa and observe the results.

package org.programs;

public class Ques7m {

public static void main(String[] args) {

double doubleValue = 1234.5678;

int intValue = (int) doubleValue;

long longValue = (long) doubleValue;

float floatValue = (float) doubleValue;

System.***out***.println("double to int: " + intValue);

System.***out***.println("double to long: " + longValue);

System.***out***.println("double to float: " + floatValue);

int i = 2000;

double doubleFromInt = i;

long l = 3000L;

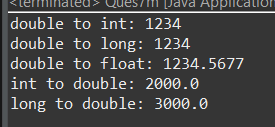
double doubleFromLong = l;

System.***out***.println("int to double: " + doubleFromInt);

System.***out***.println("long to double: " + doubleFromLong);

}

}



#### ****8. Conversion between Primitive Types and Strings****

Initialize a variable of each primitive type with a user-defined value and convert it into String:

* + First, use the toString method of the corresponding wrapper class. (e.g., Integer.toString()).
  + Then, use the valueOf method of the String class. (e.g., String.valueOf()).

package org.programs;

public class Ques8 {

public static void main(String[] args) {

int intValue = 42;

double doubleValue = 3.14;

boolean booleanValue = true;

char charValue = 'A';

long longValue = 123456789L;

float floatValue = 2.718f;

short shortValue = 100;

byte byteValue = 10;

// using toString method of wrapper classes

String intString = Integer.toString(intValue);

String doubleString = Double.toString(doubleValue);

String booleanString = Boolean.toString(booleanValue);

String charString = Character.toString(charValue);

String longString = Long.toString(longValue);

String floatString = Float.toString(floatValue);

String shortString = Short.toString(shortValue);

String byteString = Byte.toString(byteValue);

// using valueOf method of String class

String intStringValueOf = String.valueOf(intValue);

String doubleStringValueOf = String.valueOf(doubleValue);

String booleanStringValueOf = String.valueOf(booleanValue);

String charStringValueOf = String.valueOf(charValue);

String longStringValueOf = String.valueOf(longValue);

String floatStringValueOf = String.valueOf(floatValue);

String shortStringValueOf = String.valueOf(shortValue);

String byteStringValueOf = String.valueOf(byteValue);

System.out.println("Using toString method:");

System.out.println("int: " + intString);

System.out.println("double: " + doubleString);

System.out.println("boolean: " + booleanString);

System.out.println("char: " + charString);

System.out.println("long: " + longString);

System.out.println("float: " + floatString);

System.out.println("short: " + shortString);

System.out.println("byte: " + byteString);

System.out.println("\nUsing valueOf method:");

System.out.println("int: " + intStringValueOf);

System.out.println("double: " + doubleStringValueOf);

System.out.println("boolean: " + booleanStringValueOf);

System.out.println("char: " + charStringValueOf);

System.out.println("long: " + longStringValueOf);

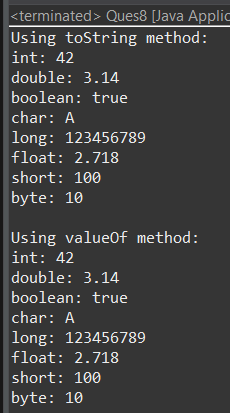
System.out.println("float: " + floatStringValueOf);

System.out.println("short: " + shortStringValueOf);

System.out.println("byte: " + byteStringValueOf);

}

}



#### ****9. Default Values of Primitive Types****

Declare variables of each primitive type as fields of a class and check their default values. (Note: Default values depend on whether the variables are instance variables or static variables).

package org.programs;

public class DefaultValues {

// Instance variables (fields)

private int intInstance;

private double doubleInstance;

private boolean booleanInstance;

private char charInstance;

private long longInstance;

private float floatInstance;

private short shortInstance;

private byte byteInstance;

// Static variables

private static int *intStatic*;

private static double *doubleStatic*;

private static boolean *booleanStatic*;

private static char *charStatic*;

private static long *longStatic*;

private static float *floatStatic*;

private static short *shortStatic*;

private static byte *byteStatic*;

public static void main(String[] args) {

DefaultValues obj = new DefaultValues();

// Print default values for instance variables

System.***out***.println("Instance variable default values:");

System.***out***.println("int: " + obj.intInstance);

System.***out***.println("double: " + obj.doubleInstance);

System.***out***.println("boolean: " + obj.booleanInstance);

System.***out***.println("char: '" + obj.charInstance + "'");

System.***out***.println("long: " + obj.longInstance);

System.***out***.println("float: " + obj.floatInstance);

System.***out***.println("short: " + obj.shortInstance);

System.***out***.println("byte: " + obj.byteInstance);

// Print default values for static variables

System.***out***.println("\nStatic variable default values:");

System.***out***.println("int: " + *intStatic*);

System.***out***.println("double: " + *doubleStatic*);

System.***out***.println("boolean: " + *booleanStatic*);

System.***out***.println("char: '" + *charStatic* + "'");

System.***out***.println("long: " + *longStatic*);

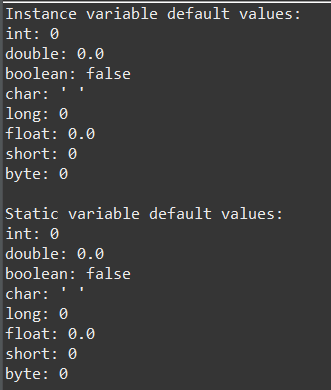
System.***out***.println("float: " + *floatStatic*);

System.***out***.println("short: " + *shortStatic*);

System.***out***.println("byte: " + *byteStatic*);

}

}



#### ****10. Arithmetic Operations with Command Line Input****

Write a program that accepts two integers and an arithmetic operator (+, -, \*, /) from the command line. Perform the specified arithmetic operation based on the operator provided. (Hint: Use switch-case for operations).

package org.programs;

public class ArithmeticOperations {

public static void main(String[] args) {

// Print the number of arguments for debugging

System.***out***.println("Number of arguments: " + args.length);

// Check if the correct number of arguments are provided

if (args.length != 3) {

System.***out***.println("Usage: java ArithmeticOperations <num1> <num2> <operator>");

System.***out***.println("Example: java ArithmeticOperations 10 5 +");

return;

}

try {

// Parse the integers from command line arguments

int num1 = Integer.*parseInt*(args[0]);

int num2 = Integer.*parseInt*(args[1]);

String operator = args[2];

// Variable to store the result of the operation

double result;

// Perform the operation based on the operator

switch (operator) {

case "+":

result = num1 + num2;

System.***out***.println("Result: " + result);

break;

case "-":

result = num1 - num2;

System.***out***.println("Result: " + result);

break;

case "\*":

result = num1 \* num2;

System.***out***.println("Result: " + result);

break;

case "/":

// Check for division by zero

if (num2 == 0) {

System.***out***.println("Error: Division by zero");

} else {

result = (double) num1 / num2;

System.***out***.println("Result: " + result);

}

break;

default:

System.***out***.println("Error: Invalid operator. Use +, -, \*, or /");

break;

}

} catch (NumberFormatException e) {

System.***out***.println("Error: Invalid number format");

}

}

}

