

# CLASS AS THE BASIS OF ALL COMPUTATION

Objects encapsulates state and behavior-numerous example, member variables, attributes or features. Variable define states, Member functions/operation/methods/messages defines behaviors. Class as abstractions for set of objects, class as an object factory, concept of type, primitive data types, composite data types. Variable declaration for both types, difference between the two types. Objects as instance of a class. Modeling by composition.

Q. What are keywords? can keywords be used as a identifiers?

Ans: Keywords are the words that convey a special meaning to the language

compiler. No, keywords can never be used as identifiers.

Q. What is an identifier? What is the identifier formatting rule of Java? OR  
What are the rules for naming a variable?

Ans: Identifiers are names given to different parts of a program e.g. variables, functions, classes etc. The identifiers in Java.

- (i) Can contains alphabets, digits, dollar sign and underscore.
- (ii) Must not start with a digit.
- (iii) Can not be a Java keywords.
- (iv) Can have any length and are case-sensitive.

Q. Why keyword is different from identifiers?

Ans: Keywords are predefine sets of words that have a special meaning for the Java compiler. Identifiers on the other hand are created by Java programmers in order to

give names to variables, function, classes etc.

**Q.** State the difference between Token and Identifier.

**Ans:** The smallest individual unit of a program is known as Token. The following Tokens are available in Java: Keywords, Identifiers, Literals, Punctuations, Operators.

Identifiers are names given to different parts of a program e.g. variables, functions, classes etc. The identifiers in Java.

**Q.** What are literals? How many types of integer literals are available in Java?

**Ans:** A literal is sequence of characters used in a program to represent a constant value. For example ‘A’ is a literal that represents the value A of type char, and 17L is a literal that represents the number 17 as value of type long. Different types of literals available in Java, they are: Integer

literal, Float literal, Boolean literal, Character literal, String literal and null literal.

**Q.** What is an integer constant? Write integer forming rule of Java.

**Ans:** Integer constants are whole numbers without any decimal part. The rule for forming an integer constants is: An integer constant must have at least one digit and cannot contain a decimal point. It may contains + or – sign. A number with no sign is interpreted to be positive.

**Q.** What do you mean by Escape sequence and name few escape sequences in Java?

**Ans:** Java have certain nongraphic characters (nongraphic characters are those characters that can not be typed directly from keyboard e.g. backspace, tab, carriage return etc.). Those nongraphic character can be represented by escape sequence. An escape sequence is represented by backslash followed by one

or more character. The few escape sequence characters are: \n for new line, \t for Horizontal Tab, \v for Vertical Tab, \b for Backspace, \" for Double Quotes etc.

**Q.** How many integer constants are allowed in Java? How are they written?

**Ans:** Java allows three types of integer constants: Octal (base 8), Decimal (base 10), and Hexadecimal (base 16). An Octal integer must be started with a zero ‘0’, a Hexadecimal integer starts with a ‘0X’, all others are treated as decimal integer constant.

**Q.** What is meant by a floating constant in Java? How many ways can a floating constant be represented into?

**Ans:** Floating constants are real numbers. A floating constant can either be a fractional or in exponent form.

**Q.** Differentiate between Integer and Floating type constants.

**Ans:** Integer constants are the whole

numbers (without decimal points). e.g. 1231. Floating point constants are fractional numbers (number with decimal points). e.g. 14.2356

Q. Write the following real constants into fractional form: 0.113E04, 0.417E-04, 0.4E-05, 0.123E02

Ans: 0.113E04 becomes 1130, 0.417E-04 becomes .0000417, 0.4E-05 becomes .000004, 0.123E02 becomes 12.3

Q. What is a type or ‘Data Type’? How this term is related to programming?

Ans: A type or datatype represents a set of possible values. When we specify that a variable has certain type, we are saying what values the expression can have. For example to say that a variable is of type int says that integer values in a certain range can be stored in that variable.

Q. What is primitive data type? Name its different types.

Ans: Primitive data types are those that are

not composed of other data types. Numeric Integral, Fractional, character and boolean are different primitive data types.

Q. State the two kind of data types?

Ans: The two types of data types are:

Primitive and non-

primitive/composite/user define data types.

The primitive data types are: byte, short, int, long, float, double, char and Boolean.

The non-primitive/reference data types are: class, array and interface.

Q. Write down the names of three primitive and three non-primitive/reference data types in Java/BlueJ.

Ans: The primitive data types are: byte, short, int, long, float, double, char and Boolean. The non-primitive/reference data types are: class, array and interface.

Q. How many bytes occupied by the following data types: byte, short, int, long, float, double, char, boolean.

Ans: char-2 byte, byte-1 byte, short-2

bytes, int-4 bytes, long-8 bytes, float-4 bytes, double-8 bytes, boolean-Java reserve 8 bits but only use 1 bit.

Q. What is the range of the following data types: byte, short, int, long, float, double, char, boolean.

Ans: byte -> -128 to 127

short -> -32768 to 32767

int -> -2<sup>31</sup> to 2<sup>31</sup>-1

long ->-2<sup>63</sup> to 2<sup>63</sup>-1

float -> -3.4×10<sup>38</sup> to 3.4×10<sup>38</sup>

double -> -1.7×10<sup>308</sup> to 1.7×10<sup>308</sup>

char -> 0 to 65536

boolean – > true or false

Q. What is the largest and smallest value for floating point primitive data types float?

Ans: The smallest value is -3.4E+38 and largest values is 3.4E+38 of floating point data type.

Q. What is Token? What are the tokens available in Java?

**Ans:** The smallest individual unit of a program is known as Token. The following Tokens are available in Java:- Keywords, Identifiers, Literals, Punctuations, Operators.

**Q.** What do you mean by variables?

**Ans:** A variable is a named memory location, which holds a data value of a particular data types. E.g. double p;

**Q.** What do you mean by variables? What do you mean by dynamic initialization of a variable?

**Ans:** A variable is a named memory location, which holds a data value of a particular data types. When a method or functions is called and the return value is initialise to a variable is called dynamic initialisation. example double  
p=Math.pow(2,3);

**Q.** What is the function of an operator?

**Ans:** Operators are special symbols that

represent operations that can be carried out on variables, constants or expressions.

Q. What do you mean by operator and write the name of all operators given in your textbook.

Ans: The operations are represented by operators and the object of the operations are referred to as operands. The types of Operators available in Java are: 1. Arithmetic 2. Increment/Decrement 3. Relational 4. Logical 5. Shift 6. Bitwise 7. Assignment 8. Conditional 9. [] operator 10. new operator 11. (type) cast Operator 12. () operator. 13. dot operator.

Q. What are arithmetic operators?

Ans: Arithmetical operators are used for various mathematical calculations. The result of an arithmetical expression is a numerical values. Arithmetical operators are of following types: Unary and Binary operators.

**Q.** Write major difference between the unary and binary operators?

**Ans:** The operators that acts on one operand are referred to as Unary Operator. There are two Unary operators Unary + operator and Unary – operator. The operators that acts upon two operands are referred to as Binary Operator. The Binary Operators are Addition(+), Subtraction (-), Multiplication (\*), Division (/) and Modulus (%).

**Q.** What is increment operator? What are postfix and prefix increment operators?

**Ans:** The ‘++’ operator is called increment operator. The increment operators add 1 to its operand. These are two types (i) Prefix and (ii) Postfix The prefix version comes before the operand for e.g. `++a`, whereas postfix comes after the operand e.g. `a++`

**Q.** Find the value of x after evaluating `x += x++ + -x + 4` where `x=3` before the evaluation. Explain your answer.

Ans: Result is 13, because  $x++$  is 3,  $-x$  is 2 + 4 the answer is 9 add this with  $x$  that is 3 it becomes 12 and due to pre increment of  $x++$  the result becomes 13.

Q. What do you mean by Relational Operators.

Ans: Relational operators are used to determine the relationship between different operands. These are used in work of compression also. The relational expression (condition) returns 0 if the relation is false and return 1 if the relation is true.  $<$  (less than),  $>$  (greater than),  $\leq$  (less than equals to),  $\geq$  (greater than equals to),  $\equiv$  (equals to),  $\neq$  (not equals to).

Q. What is Logical operators?

Ans: The logical operators combine the result of or more than two expressions. The mode of connecting relationship in these expressions refers as logical and the expressions are called logical expression.

The logical expression returns 1 if the result is true otherwise 0 returns. The logical operators provided by Java are && Logical AND, || Logical OR, ! Logical NOT.

Q. What do you mean by Assignment Statement or Assignment Operator?

Ans: Assignment operator is represented by symbol ‘=’. It takes the value on the right and stores it in the variable on the left side. for example  $x = y + 30$

Q. What do you mean by Shift operators?  
OR Differentiate between Shift LEFT and Shift RIGHT operators.

Ans: A Shift operators performs bit manipulation on data by shifting the bits of its first operand right to left. The shift operators available in Java are:

- (1)  $>>$  shift bits of right by distance.  
(signed shifting)
- (2)  $<<$  shift bits of left by distance. (signed shifting)

(3) >>> shift bits of right by distance (unsigned shifting)

Q. Differentiate between Shift LEFT and Shift RIGHT operators.

Ans: Shift LEFT (<<) operator shifts the bit pattern of the operand towards left by defined number of bits. Shift RIGHT (>>) operator shifts the bit pattern of the operand towards right by defined number of bits.

e.g.  $13 \gg 2$  is 3

binary value of 13 is  $1101 \gg 2$  is 0011 is equivalent to 3. Similarly LEFT shift (<<) operator is also work.

Q. What do you mean by Bitwise operators?

Ans: The Bitwise operations are performed by Bitwise operator. The Bitwise operations calculate each bit of their result by comparing the corresponding bits of the two operands.

(a) The AND operator &

- (b) The OR operator |
- (c) The XOR operator ^
- (d) The compliment operator ~

Q. Illustrate ‘?’ operator with an example?

Ans: It is a conditional operator, that stores a value depending upon a condition. This operator is also known as ternary operator.

The syntax for this operator is

expression1?expression2:expression3

the example

is bonus=sales>15000?250:50;

Q. What is the purpose of new operator?

Ans: We can use new operator to create a new objects or new array.

Ex. myClass obj = new myClass();

int arr[] = new int[5];

Q. What do you mean by precedence?

Illustrate with the help of example.

Ans: Precedence is the order in which a program evaluates the operations in a formula or expression. All operators have precedence value. An operator with higher

precedence value is evaluated first then the operator having lower precedence value. consider the following example

$x = 5 + 4 * 6;$

The value of this expression is 29 not 54 or 34. Multiplication has been performed first in this expression.

Q. What is operands?

Ans: An operator acts on different data items/entities called operands.

Q. What do you mean by constant? How you declare a variable as constant variables.

Ans: The memory variables/locations whose values can not be changed within the program is called constants. The keyword final makes a variable as constants.

Q. Which class is used for using different mathematical function in Java program?

Ans: The class used for different

mathematical functions in Java is  
java.lang.Math

Q. Write down the equivalent expression  
for the mathematical expression (a)  $(\cos x/\tan^{-1} x)+x$  (b)  $|e^x - x|$

Ans:  $(\text{Math.cos}(x)/\text{Math.atan}(x)) + x$  and  $\text{Math.abs}(\text{Math.exp}(x)-x)$

Q. What is the difference between these  
two function Math.ceil() and Math.rint(),  
explain with example.

Ans: Math.ceil() this function returns the  
smallest whole number greater than or  
equal to the given number. e.g.

$\text{Math.ceil}(12.85)$  gives output 13 and  
 $\text{Math.ceil}(12.35)$  also gives output 13.

Where as the Math.rint() returns the  
roundup nearest integer value. e.g.

$\text{Math.rint}(12.85)$  gives output 13 but  
 $\text{Math.rint}(12.35)$  gives output 12.

Q. What do you mean by type conversion?  
What is the difference between implicit  
and explicit type conversion explain with

example.

Ans: The process of converting one predefined type into another is called Type Conversion.

A implicit type conversion is a conversion performed by the compiler. The Java compiler converts all operands up to the type of the largest operand. This is also known as type promotion. e.g. 'c'-32 is converted to int type. Where as an explicit type conversion is user defined that forces an expression to be of specific type, this also known as type casting. e.g.  
 $(\text{float})(x+y/2)$

Q. What is coercion? How it is implemented?

Ans: Implicit type conversion of an expression is termed as coercion. A implicit type conversion is a conversion performed by the compiler. The Java compiler converts all operands up to the type of the largest operand. This is default type conversion.

Q. What do you mean by type casting?

What is the type cast operator?

Ans: The explicit conversion of an operand to a specific type is called type casting.

The operator that converts its operand to a specified type is called the typecast operator. The typecast operator is ( ) in Java and is used as (type-to-be-converted-in)

Q. Explain the methods print() and println()?

Ans: A computer program is written to manipulate a given set of data and to display or print the results. Java supports two output methods that can be used to send the results to the screen. print() method println() method.

The print() method sends information into a buffer. This buffer is not flushed until a new line (or end-of-line) character is sent. As a result print() method prints output on one line.

The println() method by contrast takes the

information provided and displays it on a line followed by a line feed.

Q. What is an Expression? Explain its different types.

Ans: An Expression is any statement which is composed of one or more operands and return a value. It may be combination of operators, variables and constants. There are three different types of expressions.

(1) Constant Expressions:  $8 * 12 / 2$

(2) Integral Expressions: formed by connecting integer constants  $x = (a + b)/2$

(3) Logical Expressions:  $a > b$  or  $a != b$

Q. Mention two different styles of expressing a comment in a program.

Ans: The two ways of inserting comments in a program are:

(i) using // single line  
comments

(ii) using /\* \*/ multiple line  
comments

**Q.** Differentiate between operator and expression.

**Ans:** The operations are represented by operators and the object of the operations are referred to as operands. The expression is any valid combination of operators, constant and variables.

**Q.** What is a compound Statement? Give an Example.

**Ans:** It is a block of code containing more than one executable statement. In Java the { } is called block and the statements written under { } is called compound statements or block statement. The { } opening and closing braces indicates the start and end of a compound statement.

```
for(int i=1;i<=5;i++)  
{  
    System.out.println("Hello");  
    System.out.println("How");  
    System.out.println("are you?");  
}
```

