ARRAY:

1.what is array?

It is an user defined data structure which contains same type of data element in contigious memory places.

int a[]=new int[10];//declaration can contains 10 integer value

int []a=new int[10];//also correct

double a[]={2,4,7.8,9.6,5}//declaration with assigning value

double a[]=new double[]{2,4,7.8,9.6,5};//also correct

2)what is length?

It is a property. It finds number of elements in the array.

int a[]=new int [15]; int k=a.length; sopln(k);

o/p= 15

3)

Linear search	Binary search
i)array may or may not be sortd	i)array must be sorted
ii)slow process	ii)fast process
iii)searching through out array	iii)searching in either half
iv)sequential searching	iv)searching using key value/mid value

4)advantage of binary search over linear search?

Faster process

5)disadvantage of binary search over linear search?

In binary search array must be sorted.

6)advantage of array?

i)More than one variable possible to store under single variable name

ii)elements in contagious places, hence accessing becomes first.

7)disadvantage of array?

i)size have to specify before use

ii)only same type of element possible to store

8)why is array called subscripted variable?

Array is accessed by its subscript position, hence it is called subscripted variable.

9. find size of array using primitive data.

Primitive data	Size(in byte)	Size(in bits)	default value
byte	1	8	0
boolean	1	8	false
char	2	16	'\u0000'
short	2	16	0
int	4	32	0
float	4	32	0.0f
long	8	64	0L
double	8	64	0.0d

10. an array A contains 10 double values. Print size in bits and byte and print default value of each cell.

Size in bits 10*64=640 bites

Size in byte 10*8=80 byte

default value: 0.0d

11. syntax , size(in bits) for boolean array of 10 elements

Syntax: boolean b[]=new boolean[10];

Size in bytes: 10*1=10 byte

Size in bits:10*8=80 bits.

12)find output:

```
int a[]={3,12,6,16,9,20};

int s=0;

Sopln("length="+a.length);

Sopln(Math.ceil(Math.sqrt(a[3]))

for(int i=0;i<a.length;i++)

{if(a[i]%4==0)

s=s+a[i];

}

Sop("sum ="+s);
```

o/p

length=6

4.0

sum=48

13)int a[]=new int [8];

```
for(int i=0;i<8;i++)

{if(i%2==0)

a[i]=a[i]+3*i;

}
```

for(int i=0;i<6;i++)

Sop(a[i]);

o/p….0 0 6 0 12 0

14)all methods of String class in java.lang package.

15)

length	length()
Used to find number of element in the array	Used to find number of number of character in array
It is property	It is method
int a[]=new int[10]; Sopln(a.length)-> 10	String st="Kolkata Sopln(st.length() 7

16)array ,String are composite/reference/user defined data type

Array ,String passed in method is known as pass by reference

17)Any change in array must be reflected in actual parameter

Static/class/global	Non static
Created only once whatever may be number of object	Created repeatedly with every object
Can't use non static method or member	Can use static method or member

String	String Buffer
Immutable	Mutable
For every changes new object is ceated	Change made on same object

Oop special:

Encapsulation: binding or wrapping up of data and method together, abstraction possible for Encapsulation

Abstraction: hiding details, secured, implemented through encapsulation

Inheritance: subclass, superclass relation, accessing data of one class by another

Polymorphism: same external feature but different internal interface, method overloading(static binding) method overriding(dynamic binding)

Class: blue print of object, object factory

Object: unique entity of class,

Super class/base class/parent class(from where data member inherited)

Sub class/child class/derived class(which inherits)

ACCESS SPECIFIER:

private can be accessed by that class only and cannot be inherited

default/friendly can be accessed by all class of that package

protected accessed by all classes of that package and first subclasss in other package

public accessed by all classes of that package as well as other package

Static :

i)not bounded to object,

ii) bounded to class,

iii)change in any object effect on other

iv)static method can not use non static for that variable at a time

v)all object of a class contains same value

Non static/ instance variable

i)bounded to object

ii)variables may have different values for different object of the same class

Local Variable: Scope only the method block where they are declared

Searching	sorting
i)search an element in the array	i)arrange all elements in the array in proper order
ii)two type search: linear , binary	ii)two type :bubble sort, selection sort

Conditional statement or if-else

1.What is sequential statement?

This type of statements are being executed sequentially. It requires default flow of statement.

2.what selection or conditional statement?

It allows to choose a set of instruction for execution depending upon expressions true or false.

e.g if-else, switch

3.What is compound statement?

One or more than one statements under a pair of curly bracket is known as compound statement.

e.g

if(a>b){

s=s+a+b;

sop(s);

}

3.difference between if else switch

If-else/if/if-else-if	switch
It can check any relational expression	It can check only equality
It can perform test on floating point also	It cannot work on floating point
If-else	**switch**
It is bi-directional	It is multiple branching statement
if	**switch**
It is unidirectional	It is multiple branching statement
If-else-if	**switch**
It is slower	It is faster

4. What is fall through?

If 'break' is not applied after each case in switch statement then all cases will be executed until it gets break. It is called fall through.

e.g switch(ch)

{case 1:

Sop("one");

case 2:

Sop("two");

break;

case 3:

sop("three");

break;

default:

sop("invalid");

}

I/P ch=1..O/P one two

I/Pch=2...   O/P two

5.what is default statement?

In a switch case if no other case constant matches with switch value then default is executed.

e.gswitch(ch)

{case 1:

Sop("one");

break;

case 2:

Sop("two");

break;

case 3:

sop("three");

break;

default:

sop("invalid");

}

I/P ch=5 O/P invalid

6.What is use of 'break' statement in java

'break' is used to avoid fall through in a switch case. Otherwise all cases will be executed from matching case until it gets break.

e.g switch(ch)

{case 1:

Sop("one");

case 2:

Sop("two");

break;

case 3:

sop("three");

break;

default:

sop("invalid");

}

I/P ch=1..O/P one two

I/Pch=2...   O/P two

7. What are different type of case constant?

char, int

8.What is error? What are different kind of error?

Any problem occurs at compile time or runtime of a java program is known as error. Three type of error.

i)Syntax error/compile error- This error due to grammatical mistake at compile time.

e.g missing semicolon, bracket mismatch etc

ii)logical error-> This is due to error in program logic development. Output must comes but not desired.

e.g Incorrect formula.

iii)Runtime error: This occurs during execution of a java program due to violation of java rules and limitation in java execution environment. e.g Division by zero.

9.What is scope of variable?

It indicates the program region through which a variable is accessible in a program.

e.g void sum(int a, int b) {

int s=0;

```
for(int i=1;i<=10;i++)

{s=s+i;

}

}
```

Here scope of s,a,b are through sum() method and scope of 'i' through loop

Write down name of following error:

i)semicolon not given at the end of statement:

ii)addition operation in exchange of multiplication

iii(switch(a>2)

iv)sop(56/0)

v)int a=5;b=7;

vi)square root of negative number:

answer:

i)syntax error

ii)logical error

iii)syntax

iv)runtime

v)syntax

vi)runtime

1. What is constructor? Or characteristics of constructor.

It is a member method of a class having same name as that of the class. It has no return type.

It initializes data member at the time of creation of object and it is called automatically at the time of creation of object.

2.What are different kind of constructor?

Two type.

i)Non parameterized constructor: It has no parameter. It initializes data member by default value.

ii)Parameterized constructor: It has parameter and it initializes data member by parameterized value

3.What is constructor overloading? Give example.

When a class contains more than one constructor with different parameter list then they are said to be overloaded.

Class Employee

{String name;

double sal;

int age;

Employee()//non parameterized constructor

{name="";

Sal=0.0d;

age=0;

}

Employee (String n, double s, int a)//parameterized constructor

{name=n;

Sal=s;

age=a;

}

}

constructor	Method
i)It has same name as that of the class	i)It has any valid name
ii)It has no return type	ii)it must has a return type
Iii)It initializes data member	iii)It has any purpose
iv)It is implicitly called	iv)It is explicitly called

4.difference between constructor and method

5.**what is default constructor?**

If no other constructor provided by programmer then the constructor provided by compiler is known as default constructor. Here data member initized by default value

Myclass ob=new Myclass();//default constructor

6.

User defined data/composite data/nonprimitive data	Primitive data
They are dependent on primitive data	They are independent
Length depends on user	Length is fixed
They are identified by particular block where they are defined	They are identified by through out the program
e.g..class, array	e.g int,long,float etc

What is this keyword?

i)It removes name confliction

ii)It refers current object

class Abc

{int a;

```
Abc(int a)

{this.a=a;

}

}
```

1. Call by value	Call by reference
i)Here method is called using primitive data	i)here method is called using reference data
ii)Any change in formal parameter does not reflect actual parameter	ii)Any change in formal parameter reflects the actual parameter
iii)Here value is copied from actual to formal parameter	iii)here address is copied from actual to formal parameter

e.g   void calc(int x,int y)   {
x++;y++;
}
	e.g void calc(Myclass ob)
	{ ob.x++;
	Ob.y++;
	}

2. Actual Parameter	Formal Parameter
i)It is used in method calling statement	i)It is used in method signature
ii)Data type are used with formal parameter	ii)Data type are used for call side parameter
iii)It takes value from outside	iii)It takes value from actual parameter

e.g
void main(){
int a=5,b=2;
sum(a,b);//actual parameter/argument/method calling statement
}
	e.g void sum(int x,int y){->formal parameter/parameter/method signature/method definition
	
	……
	}

Impure method	Pure method
i)It changes state of object	ii)It does not change the state of object
ii)It may or may not return value	ii)It returns a value

e.g void setdata(int x){
this.x=x;
}
	e.g int getdata(){
	return x;
	}

Static method	Non static method
i)It is not associated with object	i)It is associated with object
ii)It can use only static variables or method	ii)It can use static as well as non static variable and method
e.g	e.g
Static variable/global/class variale	**Non static variable/instance variable**
i)It is not associated with object	i)It is associated with object

ii)It is used by static as well as non static method	ii)It can be used by non static method only
e.g	e.g

Local variable	Instance variable
i)It is not associated with object	i)It is associated with object
ii)It can be accessed by the block where it is declared	ii)It can be accessed by all methods of the class

e.g class abc{
int x;
void sum(int a){->'a' local variable
…..}
}
	Class abc{
	Int x;//->'x' instance variable
	Void set(){…}
	}

Local variable	Global variable/static/class
It is accessed by the block where it is declared	It is accessed by all methods of the class
It is stored in stack memory	It is stored in heap memory

e.g class abc{
static int x;
void sum(int a){->'a' local variable
…..}
}
	Class abc{
	static int x;//->'x' global variable
	Void set(){…}
	}

Access specifier: It indicates the program region through which a variable or method is accessiable.

Four type:

i)private: Accessed by that class only. It is preceded by private keyword. It is most restricted.

ii)default: Accessed by all class of that package. No keyword is used here

iii)protected: Accessed by all class of that package and next subclass of other package. It is preceded by protected keyword

iv)public: Accessed by all classes of that package as well as all classes of other package. It is preceded by public keyword. It is least restricted

What are different way of invoking method?

Two.

i)call by value/pass by value

ii)call by reference /pass by reference

what does return statement do?

i)It returns value to calling method

ii)It returns control to calling method

What do u mean by void?

'void' means the method returns no value

What is method?

It is sequence of some declarative and executable statement. It is also known as subroutine or sub program and is called from different part of program

Functionality of method.

i)To cope with complexity- It breaks a program into sub program to cope with complexity

ii)Hiding details- private members are used from methods ,which are not accessible from outside

iii)Reusability: Once it is written can be used from different part

What is method overloading?

When a class contains more than one method having same name but different parameter list then the methods are said to be overloaded

e.g void sum(int a, int b)

{......

}

Void sum(double a,int b,int c)

{....

}

Void sum()

{.....

}

What is method prototype?

It is the first line of a method which contains return type , method name and parameter list.

Int sum(int a,int b,double c)-> method prototype

What is method signature?

The parameter list (number of parameter and data type of parameter) in a method is known as method signature.

Int sum(int a,int b, double c);

Here method signature ..int a,int b,double c

e.g of Static variable: and nonstatic /instance variable and static, non static method

class abc

{

int a;//instance/non static variable variable

static int b;//static variable

void change()

```
{a++;b++;// non static method can use both static and non static variable

}

static void change1(){

b++;//static method can only use static variable

}

}
```

History Development of java(chapter 2)

1.Feature of Bluej

i)It is windows based

ii)It facilitates us with sample program

iii)Debuging and correcting error is easily done

iv)Both compiled and interpreted

2.what is comment line? What are different kind of comment line?Give example

Comment lines are used to make the code understandable to other . It is not compiled.

There are three type of comment line

i)Single Line comment(//)..when programmer needs to comment on single line

ii)Multiline comment(/*……*/)when programmer needs to comment more than one line

iii)Documenting comment(/**…..*/)when programmer needs to include some text document which is some description about the program

3.what is byte code?

After java compilation an intermediate code is created called byte code. It is platform independent. It makes java platform independent

4.what is JVM?

It is java virtual machine. It is an interpreter. It helps to run byte code from machine to machine.

5. what are different kind of java program?

Two type of java program.

i)java stand alone program(It is used for personal purpose in personal machine)

ii)java applet program(it is used for web browser)

6.what is package? What is default package in java?

A package is a group of co related classes which is included in a program by import keyword such that the user can use its implicite facility in a program.

e.g java.util, java.io,java.lang

default package in java   java.lang

7.what is import keyword?

'import' is keyword which is used to include a package with its classes in a program.

e.g import java.util.*; import java.io.*;

8)what is java API?

Java Application Programming Interface. It contains all prewritten packages, classes, methods etc.

e.g javaAPI package java.util, java.io

javaAPI class Scanner, Math

java API method Math.pow, Math.sqrt

9)what is final keyword?

'final', is a keyword which makes a variable constant through out the program.

e.g final int a=10;

10)what is 'new' keyword?

'new' is a keyword which is used to create memory space dynamically for an object.

e.g Scanner sc=new Scanner(System.in);

11)what is Math function?

All function helps in mathematical calculation is known as Math function. These all function are of java.Math class

e.g Math.sqrt(), Math.pow etc

12. difference between compiler and interpreter

compiler	interpreter
Converts whole source code into object code at a time	Converts whole source code into object code line by line
It is faster	It is slower
Displays error of whole program together	Displays error of whole program line by line

13.difference between sop() and sopln()

System.out.print()	System.out.println()
i)after printing cursor remains on the same line	i)after printing cursor moves to nextline
ii)it must contain parameter	It may or may not contain parameter

14.difference between souce code and object code

Source code	Object code
i)it is written by programmer	i)It is converted from source code by compiler

ii)it is human readable	ii)it is machine readable
iii)it is high level language	iii)it is low level language

15.difference between normal compilation and java compilation

Normal compilation	Java compilation
i)here source code is compiled only	i)Here source code is compiled as well as interpreted
ii)it creates native code	ii)it creates byte code
e.g c,c++	e.g java.python

16.difference between next() and nextLine()

next()	nextLine()
i)it takes input till space	i)it takes input including space
ii)after taking input cursor remains on the same line	ii)after taking input cursor moves to nextLine.

What is Library package? Give example.

The packages already written in java compiler is known as library package.

E.g java.util, java.io etc

What is Library class? Give example.

The classes already written in java compiler is known as library class.

E.g Scanner ,Math etc

What is Library method? Give example.

The methods already written in java compiler is known as library method.

E.g Math.pow(), nextInt() etc

Four type of access of specifier

i)private:accessed by that class only

ii)default :accessed by all classes of that package

iii)protected : accessed by all classes of that package and next subclass in other package

iv)public: accessed by all classes of that package and all classes of other package

2)i)new operator: creating memory space for object dynamically

ii)dot(.)operator: it is used to call package, classes, methods etc.

3)static variable: created only once what ever may be number of object .

All object commonly share the static variable

It is related with class not associated with object

4)non static variable: It is associated with object

It is created separately for all object

5)static method: it is related with class . not associated with object.

It can use only static variable and method

6)non static method: it is associated with object

It can use both static and non static member and method

Oop feature:

i)encapsulation (wrapping up of data member and member method)

ii)abstraction(hiding data)

iii)inheritance: capability of one class to inherit data member and member method of another class

iv)polymorphism: it is method overloading and method overriding(same external feature and different internal interface)

class: blue print of object, object fctory

object: instance of class

class ABC

{int a;

static int b;

void input()

{...input

}

```
void main()

{

ABC ob=new ABC();

ABC ob1=new ABC();

ob.input();

ob1.input();

}

}
```

Four type of access of specifier

i)private:accessed by that class only

ii)default :accessed by all classes of that package

iii)protected : accessed by all classes of that package and next subclass in other package

iv)public: accessed by all classes of that package and all classes of other package

2)i)new operator: creating memory space for object dynamically

ii)dot(.)operator: it is used to call package, classes, methods etc.

3)static variable: created only once what ever may be number of object .

All object commonly share the static variable

It is related with class not associated with object

4)non static variable: It is associated with object

It is created separately for all object

5)static method: it is related with class . not associated with object.

It can use only static variable and method

6)non static method: it is associated with object

It can use both static and non static member and method

Oop feature:

i)encapsulation (wrapping up of data member and member method)

ii)abstraction(hiding data)

iii)inheritance: capability of one class to inherit data member and member method of another class

iv)polymorphism: it is method overloading and method overriding(same external feature and different internal interface)

class: blue print of object, object fctory

object: instance of class

class ABC

{int a;

static int b;

void input()

{...input

}

```
void main()

{

ABC ob=new ABC();

ABC ob1=new ABC();

ob.input();

ob1.input();

}

}
```

.   **LOOP**

1.What is loop? Name three types of loop.

It is a iterative structure which performs same work repeatedly based on condition

> i)   for loop, ii) while loop iii)do while loop

2. what is null loop?

The loop which has no body is known as null loop. If we put semicolon after loop then it becomes null loop.

int i;

for(int i=1;i<=10;i++);

s=s+i;

sopln("sum="+s);

3.Define infinite loop

The loop which never ends due to improper condition or updation of loop counter then it is   known infinite loop.

e.g for(int i=1; i>=1;i++)

sop(i);

4. What is delay loop?

The loop which executes only for consuming timed.. no purposeful work is there. Is known as delay loop.

e.g

int i=1;

while(i<=10){

i++;

}

5. Name of two jump statement.

break and continue

break	continue
It terminates inner most loop immediately	It skips the rest part of current iteration and moves the cursor to next iteration

e.g
for(int i=1;i<=5;i++)
{
If(i==3)
break;
sop(i);
}
o/p 1 2
	e.g
	for(int i=1;i<=5;i++)
	{
	If(i==3)
	continue
	sop(i);
	}
	o/p 1 2 4 5

while	**do while**
i)it is entry control loop	i)it is exit control loop
ii)If condition is not satisfied then will not execute	ii)If condition is not satisfied it will execute at least once

e.g
int i=5;
while(i>6)
{
Sop(i);
i++;
}
o/p no output
	e.g
	int i=5;
	do
	{
	Sop(i);
	i++;
	}while(i>6);
	o/p 5

for	while
Here initialization, checking and updation are at same place	Here initialization is above, checking is at the starting of the loop, updation is inside the loop
Number of iteration is known in advance	Number of iteration is not known in advance

e.g for(int i=1;i<=10;I++)
s=s+i
	Int i=1;
	While(I<=10)
	{
	S=s+i;
	i=i+1;
	}

Math.max(34,67.3)=67.3

Math.max(7.8,9)=9.0

Math.min(8,9)=8

Math.min(5,7.8)=5.0

Math.sqrt(25)=5.0

Math.pow(4,2)=16.0

Math.ceil(4.5)=5.0

Math.ceil(6.7)=7.0

Math.ceil(-7.8)=-7.0

Math.floor(5.61)=5.0

Math.floor(-6.7)=-7.0

Math.abs(-6.7)=6.7

Math.abs(5.6)=5.6

Math.abs(-7)=7

Math.cbrt(8)=2.0

Math.round(3.4)=3

Math.round(3.5)=4

Math.round(3.7)=4

//Math.round(-4.3)=-4

//Math.round(-4.5)=-4

//Math.round(-4.8)=-5

Math.rint(3.4)=3.0

Math.rint(3.5)=3.0

Math.rint(3.51)=4.0

Math.rint(3.7)=4.0

Math.random()//it gives any value in between 0 to 1

Q)Math.sqrt(Math.floor(16.5))

Math.sqrt(16.0)=4.0

Q)Math.abs(Math.ceil(-4.6))

Math.abs(-4.0)

=4.0

Return type double->

ceil,floor,rint,pow,sqrt,random,cbrt

int->round

any kind->max, min, abs,

Math.ceil(5)=5.0

Math.floor(-6)=-6.0

c.w copy

1)Math.ceil(-9.6)=-9.0

2)Math.sqrt(25)=5.0

3)Math.pow(3,4)=3^4=81.0

4)Math.round(6.7)=7

5)Math.rint(6.5)=6.0

6)Math.abs(-4)=4

7)Math.ceil(Math.abs(-4))

Math.ceil(4)

8)Math.sqrt(Math.abs(Math.floor(-15.7)))

chapter 1(oop object oriented program)

1.Difference between oop and pop(procedure oriented program)

oop gives more stress on data rather than method hence any change within the data the method will be automatically changed.

But in case of pop it gives more stress on method rather than data .Here method have to change manually when data is changed.

2.Feature of OOP

i)Emphasis on data

ii)divides the program into some modules

iii)data is hidden and is more secured

3.Basic elements or principal of oop

i)Encapsulation(wrapping up of data member and method together)

ii)Abstraction(represents essential feature without including background details)

iii)Inheritance(reusability)

iv)polymorphism(method overloading)

v)class

vi)object

4.what is class?

A class represents a set of objects having similar characteristics and behaviour where characteristics are represented through data and behaviour are represented through method

5. what is object?

An object is an instance of class having same characteristics and behaviour where characteristics are represented through data and behaviour are represented through method.It is unique entity of a class.

e.g leaving half page

6. what is encapsulation?

It is an oop feature which encapsulates data member and member method together into a single unit called class.

7.what is abstraction?

It is an oop feature which represents essential feature without including background details.

e.g leaving half page

8.what is inheritance?

It is the capability of one class to inherit data member and member method of another class according access specifier. Here the class from which properties are inherited is known as super class and the class which inherits properties are known as subclass.

leaving half page

9. what is polymorphism?

It is a process of using a fuction for more than one purposes. It allows the use of different internal structure by keeping same external interface.

e.g method overloading, method overriding

10.why is class called object factory?

A class is a blue print of objects. It can represent a set of objects having similar characteristics and behavior where characteristics are represented through data and behavior are represented through method. Like a factory can create so many things of same type class also can represent so many object of same type.That is why class is called object factory.

11.Why is object called instance of class?

An object is a unique entity of class . It has same characteristics and behavior of the the class where characteristics are represented through data and behavior are represented through method. That is why object is called instance of class

Leaving half page

12.why is class called user defined data type?

Primitive data are not sufficient to represent every real object. Hence user needs to create data type with the help of class as per his requirement .hence class is called user defined data type.

Leaving half page

13.why is class called user composite data type?

A class can contain primitive as well as non primitive data as its data member. Hence class is called composite data

Leaving half page

14.Difference between class and object

Class	object
Blue print of object	it is an unique entity of class having same characteristics and behavior
It is an object factory	It is an instance of class
No memory space is created for class	Memory space is created for object

Test 1: 4/7/2020 10*2=20

1.write features of java

2.write 4 elements of oop

3.define encapsulation

4.define abstraction

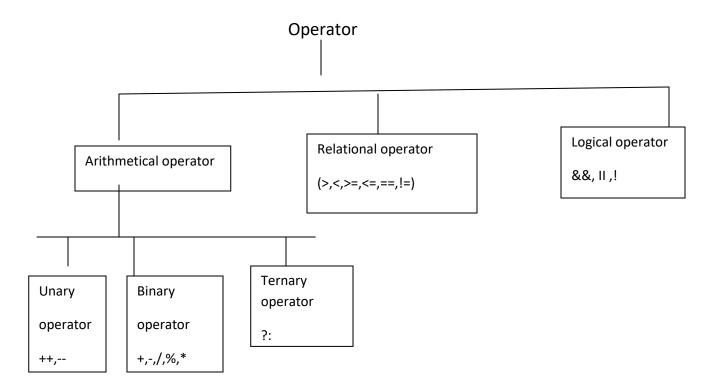5.what is inheritance? Define super class and subclass

6.difference between class and object

7.why is class called object factory?

8.why is object called instance of class?

9.why is class called composite data type?

10 example of polymorphism

Operator

```
                              Operator
                                 |
        +------------------------+------------------------+
        |                        |                        |
  Arithmetical operator    Relational operator      Logical operator
                           (>,<,>=,<=,==,!=)          &&, II ,!
        |
   +----+--------+-----------+
   |             |           |
  Unary        Binary      Ternary
  operator     operator    operator
  ++,--        +,-,/,%,*    ?:
```

1.Define operator?

Operator is basically a symbol or token which performs arithmetical and logical operations and gives a meaningful result

2.Define operand?

The values which are involved in the operation are termed as operand;

e.g..z=a+b*c

a,b,c are operand here

3. what is expression?

It is the combination of operator , operand and constant values to yield a meaningful result.

int z=a+2*b+6;

4.what is conditional operator?

It is ternary operator(?:) which works on three operand. A relational operator works here if it is true then the value after '?' returns otherwise the value after' :'

returns.

Syntax:value=(condition)?true value:false value

e.g max=(a>b)?a:b

5.what is precedence of operator?

It is the order by which operators are evaluated in an expression . If more than one has same precedence then they are executed as per their associativity.

operator
[],()
++ , -- , !
new (type)
*,/,%
+, -
>,<,>=,<=,==,!=
&&
?:
=, +=,-=

Unary	binary
It works on single operand	It works on two operand
It can increase or decrease value only	It can do any arithmetical operation
e.g ++,--	e.g +,-,*,/,%

Prefix operator	**Postfix operator**
First change the value and use it	First use the value and change it
e.g int a=5; System.out.println(++a); o/p->6	e.g int a=5; System.out.println(a++); o/p-> 5

Increment operator	**Decrement operator**
i)It increases the value of variable by one .	i)It decreases the value of variable by one
ii)It is of two type. Preincrement, postincrement	ii)it is of two type. Predecrement, postdecrement
e.g int a=7; sop(a++);sop(++a);	e.g int a=7; sop(a--);sop(--a);
Relational operator	Logical Operator
It checks the relation between two values	It works between the results yield by two relational operator
e.g >,<,>=,<=,==,!=	e.g !, &&,||
/	**%**
It gives quotient	It gives remainder
It can works on integer as well as floating point	It can works on only integer
e.g 5/2=2, 7/2=3	e.g 5%3=2,17%4=1

=	==
It ia assignment operator	It is relational operator
It assigns a value to a variable	It checks equality between two values
e.g int a=10;	e.g int a=10,b=20 if(a=b){....}
!	!=
i)It is logical operator	i)It is relational operator
ii)It reverses the value yield by relational operator	ii) it checks inequality between two primitive values
e.g int a=4,b=4 if(!(a==b))	e.g int a=4,b=4; if(a!=b)

STRING

Character class

Method name	Return type	Purpose	example
1.Character.isUpperCase(char)	boolean	It checks whether the given character is in upper case or not	Character.isUpperCase('A') true Character.isUpperCase('a') false Character.isUpperCase('?') false
2.Character.isLowerCase(char)	boolean	It checks whether the given character is in lower case or not	Character.isLowerCase('A') false Character.isLowerCase('a') true Character.isLowerCase('?') false
3.Character.toUpperCase(char)	char	It converts the given character to uppercase	Character.toUpperCase('A') A Character.toUpperCase('a') A Character.toUpperCase('?') ?
4.Character.toLowerCase(char)	char	It converts the given character to lowercase	Character.toLowerCase('A') a Character.toLowerCase('a') a Character.toLowerCase('?') ?
5.Character.isLetter(char)	boolean	It checks whether the given character is a letter or not	Chacter.isLetter('a')true Character.isLetter('1')false Character.isLetter('?')false
6.Character.isDigit(char)	boolean	It checks whether the given character is a digit or not	Chacter.isDigit('a')false Character.isDigit('1')true Character.isDigit('!')false
7.Character.isLetterOrDigit(char)	boolean	It checks whether the given character is a digit or letter	Chacter.isLetterOrDigit('a')true Character.isLetterOrDigit('1')true
8.Character.isWhiteSpace(char)	boolean	It checks whether the given character is a space or not	Character.isWhiteSpace(' ')true Character.isWhiteSpace('A')false

Return type	Method name	Purpose	example
9.int	length()	It returns the number of character in the string	"KOLKATA".length()->7 "BLUE SKY".length()->8
10.char	charAt(int)	It returns the character in the specified index	"KOLKATA".charAt(2)->L
11.int	indexOf(char)	It returns the index (first occurance)of specified character	"KOLKATA".indexOf('K')->0 "KOLKATA".indexOf('a')->-1 "KOLKATA".indexOf('p')->-1
12.int	lastIndexOf(char)	It returns the index (last occurance) of specified character	"KOLKATA".lastIndexOf('K')->3 "KOLKATA".lastIndexOf('p')-> -1
13.int	indexOf(char,int)	It returns the index(first occurance)of specified character starting from specified index	"MALAYALAM".indexOf('A',2)->3 "MALAYALAM".indexOf('A',5)->5
14.boolean	equals(string)	It checks the equality between two string values considering case	"KOLKATA".equals("KOLKATA")->true "KOLKATA".equals("Kolkata")->false "KOLKATA".equals("PUNE")->false
15.boolean	equalsIgnoreCase(String)	It checks equality between two string ignoring case	"KOLKATA".equalsIgnoreCase("KOLKATA")->true "KOLKATA".equalsIgnoreCase("Kolkata")->true "KOLKATA".equalsIgnoreCase("PUNE")->false
16.String	replace(char,char)	It replaces first specified character by the last specified character in the string	"MALAYALAM".replace('M','P')-> PALAYALAP
17.String	replace(String,String)	It replaces first specified String by the last specified String in the string	"green parrot on a green tree". replace("green", "red")->red parrot on a red tree
18.int	compareTo(String)	It compares two String lexicographically	"RAM".compareTo("RAHIM")-> 5 "RAHIM".compareTo("RAM")-> -5 "SUN".compareTo("SUNNY")-> -2 "SUNNY".compareTo("SUN")-> 2 "SUN".compareTo("SUN")-> 0 "well'.compareTo("Welcome")-> 32 "HAT".compareTo("bat")-> -26
19.int	compareToIgnoreCase(String)	It compares two String lexicographically ignoring case	RAM".compareToIgnoreCase("RAHIM")-> 5 "rahim".compareToIgnoreCase("RAM")-> -5 "SUN".compareToIgnoreCase("SUNNY")->-2 "SUNNY".compareToIgnoreCase("SUN")->2 "HAT".compareToIgnoreCase("bat")=6

			"SUN".compareToIgnoreCase("SUN")->0 "well'.compareToIgnoreCase("Welcome")->9 "HAT".compareToIgnoreCase("bat")->6

Return type	Method name	purpose	example
20.String	toUpperCase(String)	Converts the total string to uppercase	"Apple".toUpperCase()->APPLE "APPLE".toUpperCase()->APPLE
21.String	toLowerCase(String)	Converts the total string to lowercase	Apple".toLowerCase()->apple "APPLE".toLowerCase()->apple
22.String	concat(String)	It concats two string	"Kolkata".concat("pune")->Kolkatapune
23.String	substring(int)	It returns group of characters from the specified index to the end of the string	"Kolkata".substring(3)->kata
24.String	substring(int,int)	It returns group of characters from the first specified index before the last specified index of the string	"Kolkata".substring(2,5)->lka
25.int	Integer.parseInt(string) Or Integer.valueOf(String)	Converts string to primitive int	Integer.parseInt("234")->234
26.double	Double.parseDouble(String) Or Double.valueOf(String)	Converts string to primitive double	String st="67.8"; Double.parseDouble(st);->67.8
27.String	Integer.toString(int)	It converts int to String	String st=Integer.toString(45) Sop(st)->45
28.String	Double.toString(double)	It converts double to String	double d=7.8 String s=Double.toString(d); Sop(s);->7.8
29.String	String.valueOf(char)	Here character is converted to String	String.valueOf('a')="a"
30.String	trim()	It removes the leading and trailing spaces	" Kolkata city " Kolkata city
31.boolean	startsWith(String)	It checks the given string starts with the parameterized string or not	"KOLKATA".startsWith("KOL")->true "KOLKATA".startsWith("K")->true "KOLKATA".startsWith("kol")->false
32.boolean	endsWith(String)	It checks the given string ends with the parameterized string or not	"KOLKATA".endsWith("KATA")->true "KOLKATA"endsWith("A")->true "KOLKATA"endsWith("kol")->false

VALUES AND DATATYPE

1.what is token?

Each smallest individual unit used in java program is known as token. E.g

Literal,operator, separator,punctuator ,data type etc

2.what is literal?

Each constant value used in java program is known as literal.

e.g

int literal(5,7,8,3)

double literal(5.6,7.8)

char literal ('a','d')

3.what is variable?

A variable is a named memory location which contains a value.

e.g int a=10;

'a' is a variable

4.what is identifier?

It is an user given name to class array method etc.

e.g Scanner sc=new Scanner(System.in); where 'sc' is a Scanner object

5.what is data type?

Data type is the type or kind of data and associated operation to handle them.

Two type:

Primitive data   and non primitive data

6. Dfference between primitive and non primitive data

Primitive data	Non primitive data
i)fundamental data	i) derived data
ii)independent data	ii)indirectly dependent on primitive data

Iii) they have fixed length	iii)their length depends on user
e.g int , float, char etc	e.g class, array etc

7.what type conversion and what are different kind of type conversion?

In a mixed expression when data are converted to a single data type then it is known as type conversion.

Two type of conversion.

i)implicite type conversion ii)explicite type conversion

8.Difference between implicite and explicite type conversion

Implicite type conversion/coercion	Explicite conversion/casting
i)Here data is converted to higher type	i) Here data is converted to desired type
ii)User intervension is not needed here	ii)User intervension is needed here
iii)data value will not loss here	iii)data value may or may not loss here

e.g int a=8; double d=7.8
d=a;
	e.g int a=8; double d=7.8
	a=(int)d;
	d=(double)a;

9.what is escape sequence?

There are some non graphic character which are used as command to direct the cursor while printing.

e.g \t horizoantal tab

\n   new line

10.what is punctuator?

There are some punctuation sign used as special character in java. E.g '?', ':' ,'.'

11.what is separator?

Separator are used to separate same kind of variable . ',' is used as separator.

12. Write down naming rules for variable

i)A variable may have any number of character.

ii)It may contain alphabets, digits and underscore

iii)Variable name should be meaningful

iv)special character '_' can be applied in between any two characters

13. Reasons for a variable to be invalid?

i) A variable name should not starts with digit.

ii)A variable name should not include any space in between two characters

iii)It should not be any keyword

14. What is arithmetic expression and what are different kind of arithmetic expression ?

Give definition and example

An expression which contains variable constants and arithmetic operators is termed as arithmetic expression

Two type:

i)Pure Arithmatic Expression: In this expression it contains same type of data items(variable and constant) e.g int a=10; int x=a*2+20;

ii)Impure Arithmatic Expression: In this expression it contains different type of data items(variable and constant) e.g double a=5.6; int b=8; double x=a+b*2;

15.Description about primitive data

Data type	Size	Default value	range
byte	1 byte(8 bits)	0	-128 to +127
boolean	1 byte	false	true, false
char	2 byte(16 bits)	'\u0000'	0 to 65535
short	2 byte(16 bits)	0	-32768 to 32767
int	4 byte(32 bits)	0	-2^{31} to $2^{31}-1$
float	4 byte(32 bits)	0.0f	
long	8 byte(64 bits)	0 L	-2^{63} to $2^{63}-1$
double	8 byte(64 bits)	0.0 d	

16.List of escape sequence

\t->horizontal tab	\'->single quote
\v->vertical tab	\''->double quote
\\->back slash	\b->back space

\0->null	\n->new line

Here's the full list:

- \t - tab.
- \b - backspace (a step backward in the text or deletion of a single character).
- \n - new line.
- \r - carriage return. ()
- \f - form feed.
- \' single quote.
- \" double quote.
- \\ **backslash**.

Satic initialization	Dianamic initialization
Value to the variable is given at compile time	Value to the variable is given at run time
e.g int a=10;	e.g int a=sc.nextInt(); where sc ia a Scanner object

Char	String
Take one character	Takes group of charactres
Enclosed within single quote	Enclosed within double quote
e.g char ch='a'	String st="a", st1="Kolkata"
float	double
It has memory space 4 byte	It has memory space 8 byte
It takes decimal place upto 4 place	It take decimal places upto 8 place
Dedfault value 0.0 f	Default value 0.0d

Int	float
Takes only integer value	Takes integer as well as floating value
Default value 0	Default value 0.0f

Wrapper class: it is a member of java library java.lang which converts primitive data in terms of object. it also provides conversion

Primitivedata value	wrapper class	size(byte)	default
byte	Byte	1	0
boolean	Boolean	1	false
short	Short	2	0
char	Character	2	'\u0000'
int	Integer	4	0
float	Float	4	0.0f/0.0F
long	Long	8	0L
double	Double	8	0.0d

Wrapper class method description:

1)

a)Integer ob=new Integer(5);//autoboxing(converting primitive value to object)

Integer obj=5;//autoboxing

b)int a=ob;//unboxing(converting object to primitive value)

OR

Int a=ob.intValue()//(converting object to primitive value)

2)

a)int a=Integer.parseInt("5");(conversion from string to int)

OR

int a=Integer.valueOf("5");(converting string to int)

b)String s=Integer.toString(100);(conversion from int to string)

c)double d=Double.parseDouble("8.9")-> (converting String to double)

OR

double d=Double.valueOf("8.9") -> (converting String to double)

d)Double ob=new Double(7.8);//primitive data to object

3)

static methods is called directly with class name

all methods(e.g parseDouble(), toString(), valueOf())of wrapper

4)e.g

String a="20",b="30";

Sopln(a+b);

int c=Integer.parseInt(a);

```
int d=Integer.parseInt(b);

Sopln(c+d);

o/p

2030

50
```

//package contains group of classes

Package java.util for Scanner class

Package java.io for BufferedReader class

anything else->package java.lang

default package(java.lang)