cbse cs n ip

Programming Fundamentals

Type A: Very Short Answer Questions

1	**Name the character set supported by Java.**
Ans	Java uses the Unicode character set.
2	**What is meant by token? Name the tokens available in Java.**
Ans	The smallest individual unit in a program is known as a **Token.** Following are the tokens available in Java. 1. Keywords 2. Identifiers 3. Literals 4. Punctuators 5. Operators
3	**What are Keywords? Can keywords be used as identifiers?**
Ans	**Keywords** are the words that convey a special meaning to the language compiler. These are reserved for special purpose. Keywords cannot be used as identifiers.
4	**What is an identifier? What is the identifier forming rule of Java?**
Ans	**Identifiers** are user define names for different parts of program. **Identifier forming rules of java state the following:** 1. Identifiers can have alphabets, digits, and underscore and dollar sign characters. 2. They must not be a keyword or Boolean literal or null literal. 3. They must not begin with a digit. 4. They can be of any length. 5. Java is case sensitive i.e., upper-case letters and lower case letters are treated differently.
5	**Is Java case sensitive? What is meant by the term 'case sensitive'?**
Ans	Java is case sensitive. **Case sensitive** meant upper-case letters and lower case letters are treated differently.
6	**Which of the following are valid identifiers and why/why not?** **Data_rec, _data, 1 data, data 1, my.file, asm, switch, goto, break**
Ans	**1. Data_rec:** Valid identifiers **2. _data:** Valid identifiers **3. 1 data:** Not valid because they must not begin with a digit **4. data 1:** Not valid because space is not allowed. **5. my.file:** Not valid because special character not allowed. **6. Asm:** Valid identifiers **7. Switch:** Not valid because reserved keyword is not allowed. **8. goto:** Not valid because reserved keyword is not allowed. **9. break:** Not valid because reserved keyword is not allowed.
7	**What are literals? How many types of integer literals are available in java?**
Ans	**Literals** are data items that never change their value during a program run. integer-literal (decimal, octal, hexadecimal)
8	**What is an integer constant? Write integer forming rule of Java.**
Ans	A constant that is a whole number with no decimal point. It can have a leading sign and is interpreted as a decimal number. An integer constant must have at least one digit and must not contain any decimal point.
9	**How many types of integer constant are allowed in java: How are they written?**
Ans	There are three types of integer constant are allowed in java. 1. Decimal: 17 2. Octal: 011 3. Hexadecimal: 0XBC1
10	**What kind of program elements are the following: 13, 'a', 4.38925, "a", main ()?**
Ans	**13 –**integer literal

	'a' –character literal **4.38925 –**floating literal **"a" –**string literal **main() –**in-built method
11	**What kind of constant are the following: 14, 011, 0X2A, 17, 014, 0XBC1?**
Ans	**1. Decimal:** 14, 17 **2. Octal:** 011, 014 **3. Hexadecimal:** 0XBC1, 0X2A
12	**What is a character constant in Java? How are no graphic characters represented in Java?**
Ans	A **character constant** is formed by enclosing a single character from the representable character set within single quotation marks (' '). The no graphic characters can be represented by using escape sequence.
13	**Why are characters \, ', ", and ? typed using escape sequences?**
Ans	These characters can be typed from the keyboard but when used without escape sequence, these carry a special meaning and have a special purpose, however, if these are to be typed as it is, then escape sequences should be used.
14	**Which escape sequences represent the newline character & null character?**
Ans	**\n –**newline character **\0 –**null character.
15	**What is floating constant in java? How many ways can a floating constant be represented into?**
Ans	A **floating constant** is a decimal number that represents a signed real number. The representation of a signed real number includes an integer portion, a fractional portion, and an exponent.
16	**Write the following real constants into exponent form: 23.197, 7.214, 0.00005, and 0.319.**
Ans	**23.197 ->** 2.3197×10^2 =2.3197E02 **7.214 ->** 0.7214 x 10^1 = 0.7214E01 **0.00005 ->** 0.000005 x 10^1 =0.000005E01 **0.319 ->** 0.0319 x 10^1 = 0.0319E01
17	**Write the following real constants into fractional form: 0.13E04, 0.417E-04, 0.4E-05, 0.123E02**
Ans	**0.13E04 –>** 1300.0 **0.417E-04 –>** 0.000041 **0.4E-05 –>** 0.000004 **0.123E02 –>** 12.3 **Note:** **Explanation for better understanding** 0.123E02 To convert 0.123 × 1002 into regular notation: To convert 0.123 into regular format, move the decimal point to the right 2 places. Why? Because that's what × 102 meAns multiply by a 1 with 2 zero(s) after it. And, with each zero in a multiplication, the decimal moves to the right 1 place. So 0.123 × 1002 written in regular notation is 12.3
18	**What are the binary operators? Give examples of arithmetic binary operators.**
Ans	They are operators that require two operands to operate upon. The examples are: Addition operator(+), Subtraction operator(-), Multiplication operator(*), Division operator(/), and Modulus operator(%). **Example:** 1. 4+20 results in 24 2. 4-3 evaluates to 11 3. 3*4 evaluates to 12 4. 100/5 evaluates to 20 5. 19%6 evaluates to 1
19	**What will be the result of a=5/3 if a is** **(i) float (ii) int?**

Ans	(i) float-> 1.0 (ii) int->1
20	**The expression 8%3 evaluates to _____.**
Ans	2
21	**Assuming that res starts with the value 25, what will the following code fragment print out?** **System.out.println (res--);** **System.out.println (++res);**
Ans	25 25
22	**What will be the value of j=--k+2*k+l++ if k is 20 initially?**
Ans	57
23	**What will be the value of p=p*++j where j is 22 and p=3 initially?**
Ans	69
24	**What will be the value of following, if j=5 initially?** **(i) (5*++j)%6** **(ii) (5*j++)%6**
Ans	(i) 0 (ii) 1
25	**What will be the result of following expression if** **(i) age=25** **(ii) age=65** **(iii) age=85?** **age > 65 ? 350: 100.**
Ans	(i) 100 (ii) 100 (iii) 350
26	**What will be the result of the following expression if** **(i) ans=700, val=300** **(ii) ans=800, val=700** **ans – val < 500 ? 150 : 50**
Ans	(i) 150 (ii) 150
27	**Write equivalent java expression for the following expressions:** **(i)** $ut + \frac{1}{2} ft^2$ **(ii)** **|a|+b>=|b|+a** **(iii)** $\left(\frac{3x+5y}{5x+3y} - \frac{8xy}{2yx}\right)$ **(iv)** $e^{\lvert 2x^2-4x \rvert}$
Ans	(i) x=u*t+(1/2)*f*t*t; (ii) a+b>=b+a; (iii) x=((3*x+5*y)/(5*x+3*y)-(8*x*y)/(2*y*x)) (iv) Math.pow(e,(2*x*x-4*x))
28	**What is meant by implicit and explicit type conversion?**
Ans	An **implicit type conversion** is a conversion performed by the compiler without programmer's intervention. An **explicit type conversion** is user-defined that forces an expression to be of specific type.
29	**What do you mean by type casting? What is type cast operator?**
Ans	The explicit conversion of an operand to a specific type is called **Type Casting.** **Type cast operator** is used for Casts or converts a value to the specified type.
30	**What will be the resultant type of the following expressions if bh represents a byte variable, i is an int variable, fl is a float variable and db is a double variable?** **(i) Bh-i+db/fl-i*fl+db/i.** **(ii) (int) (fl+db)**

Ans	(iii) Double (iv) Double
31	**Which class is used for using different mathematical methods in Java program?**
Ans	Math class is used for using different mathematical methods in java program.
32	**What are instance variables? What are class variables?**
Ans	**Instance Variable –**A data member that is created for every object of the class. For example, if there are 10 objects of a class type, there would be 10 copies of instance variables, one each for an object. **Class Variable –**A data member that is declared once for a class. All objects of the class type, share these data members, as there is single copy of them available in memory. The class variables are declared by adding keyword static in front of a variable declaration.
33	**The modulas operator (%) can be used only with integer operands. True/False?**
Ans	False
34	**The range of values for the long type data is** **(a) -2^{31} to $2^{31}-1$** **(b) -2^{64} to 2^{64}** **(c) -2^{63} to $2^{63}-1$** **(d) -2^{32} to $2^{32}-1$**
Ans	(c) -2^{63} to $2^{63}-1$
35	**Which of the following represent(s) a hexadecimal number?** **(a) 570** **(b) (hex) 5** **(c) 0X9G** **(d) 0X5**
Ans	(d)0X5
36	**Which of the following assignments are invalid?** **(a) float x=123.4** **(b) long m=023** **(c) int n=(int) false;** **(d) double y=0X756.**
Ans	(a) and (c) are invalid.
37	**The default value of char type variable is** **(a) '\u0020'** **(b) '\u00ff** **(c) " "** **(d) '\u0000'**
Ans	(d)'\u0000'
38	**What will be the result of expression 13 & 25? (13_{10} =00001101_2, 25_{10} =00011001_2)** **(a) 38** **(b) 25** **(c) 9** **(d) 12**
Ans	(c) 9
39	**What will be result of the expression 9|9?** **(a) 1** **(b) 18** **(c) 9** **(d) None of the above**
Ans	(c) 9
40	**Which of the following will produce a value of 22 if x=22.9?** **(a) Math.ceil(x)** **(b) Math.round(x)** **(c) Math.abs(x)**

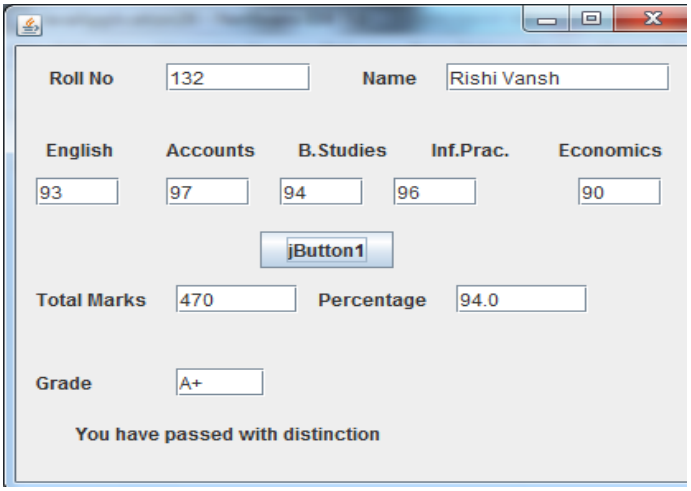
	(d) Math.floor(x)
Ans	(a) Math.floor(x)
41	**Which of the following will produce a value of 10 if x=9.7?** **(a) Math.floor(x)** **(b) Math.abs(x)** **(c) Math.round(x)** **(d) Math.ceil(x)**
Ans	(c) Math.round(x)
42	**Given the declarations** **boolean b;** **short x1=100, x2=200, x3=300;** **Which of the following statements are evaluated to true?** **(a) b=x1*2==x2;** **(b) b=x1+x2!=3*x1;** **(c) b=(x3-2*x2<0||(x3=400)<2**x2);** **(d) b(x3-2*x2>0||(x3=400)2*x2);**
Ans	(a) b=x1*2==x2;

Type B: Short Answer Questions

1	**How are Keywords different from identifiers?**
Ans	**Keywords** ✓ Keywords are reserved words used for special purpose and must not be used as normal identifier names. ✓ Example : do, if, catch, final **Identifiers** ✓ Identifiers are fundamental building blocks of a program and are used as the general terminology for the names given to different parts of the program. ✓ Example: Myfile, _ds, date_7_9
2	**What are literals in Java? How many types of literals are allowed in Java?**
Ans	**Literals** are also referred to as constants. They never change their value during program run. There are several kinds of literals available in Java. These are: integer-literals, character-literals, floating literals, boolean literals, string literals, the null literals.
3	**Can nongraphic characters be used and processed in Java? How? Give examples.**
Ans	Yes, non graphic characters can be used in Java. These characters cannot be typed directly from keyboard like backspace, tab, newline etc. These characters can be represented by using escape sequences. An escape sequence is represented by backslash (\) followed by one or more characters. Examples : \b is used to get backspace, \t is used to get tab
4	**Determine the data type of the expression** **(i)** $\left(\frac{100(1-pq)}{(q+r)}\right) - \left(\frac{(p+r)/s}{(long)(s+p)}\right)$ **(ii)** $\left(\frac{2x+3y}{5w+6z} + \frac{8t}{5u}\right)^4$ **If p, x is an int, r, w is a float, q, y is a long and s, z is double, t is short and u is long double.**
Ans	(i) double (ii) double
5	**Given that:** **int x, m =2000; short y;** **byte b1 = -40, b2; long n;** **Which of the following assignment statements will evaluate correctly? Also determine the data type of resultant variable.** **(a) x = m*b1;** **(b) y = m*b1;** **(c) n = m*3L;** **(d) x = m*3L;**

Ans	(a) x = m*b1 -> data type of x is int (c) n = m*3L -> data type of n is long

Type C: Practical/Lab Questions

1	**Develop an application to prepare students result on the basic of marks entered. A sample screenshot is being shown below:**  ➢ **Decide what controls are required, on your own.** ➢ **Decide about the number of variables required and their types on your owm.**

Ans	`private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {`
	`int m1=Integer.parseInt(jTextField4.getText());`
	`int m2=Integer.parseInt(jTextField5.getText());`
	`int m3=Integer.parseInt(jTextField6.getText());`
	`int m4=Integer.parseInt(jTextField7.getText());`
	`int m5=Integer.parseInt(jTextField8.getText());`
	`int total=m1+m2+m3+m4+m5;`
	`float per=total/5;`
	`jTextField10.setText(Integer.toString(total));`
	`jTextField11.setText(Float.toString(per));`
	`float p=Float.parseFloat(jTextField11.getText());`
	`if(p>=90)`
	`{`
	`jTextField12.setText("A+");`
	`}`
	`else if(p>=80 && p<90)`
	`{`
	`jTextField12.setText("A");`
	`}`
	`else if(p>=70 && p<80)`
	`{`
	`jTextField12.setText("B");`
	`}`
	`else if(p>=60 && p<70)`
	`{`
	`jTextField12.setText("C");`
	`}`
	`else if(p>=50 && p<60)`
	`{`
	`jTextField12.setText("D");`
	`}`
	`else`

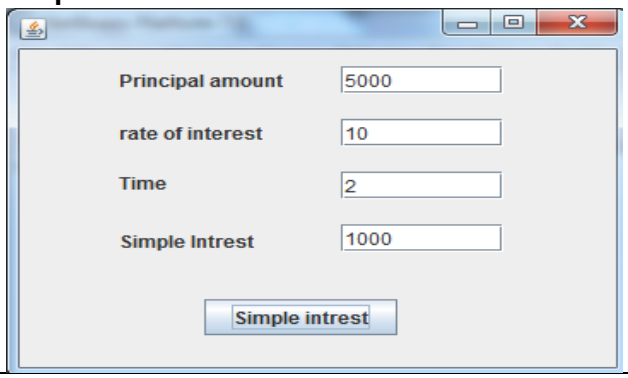
```
        {
            jTextField12.setText("E");
        }
        if(p>=70)
        {
            jLabel13.setText("You have passed with distinction");
        }
       else if(p>=60 && p<70)
        {
            jLabel13.setText("You have passed with First class");
        }
       else if(p>=50 && p<60)
        {
            jLabel13.setText("You have passed with second class");
        }
       else
        {
            jLabel13.setText("You have passed with third class");
        }
    }
```

2 **Develop an application that accepts principal amount, rate of interest and time from the user. Then it computes the simple interest and displays it.**
Decided about the controls and variables required for this application, on your own.

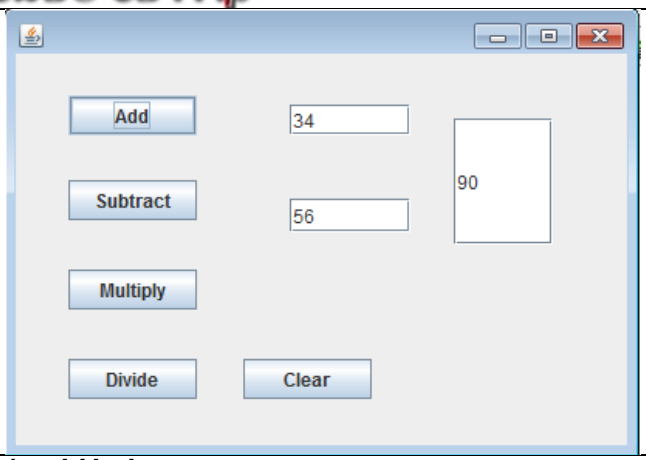
Ans

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
 {
         int p=Integer.parseInt(jTextField1.getText());
         int r=Integer.parseInt(jTextField2.getText());
         int n=Integer.parseInt(jTextField3.getText());
         int s=p*r*n/100;
         jTextField4.setText(Integer.toString(s))
 }
```

Output:



3 **Design an application that performs arithmetic operations (+, -, *, and /). The sample screenshot is being show below.**




```
// Addition
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
 {
    int num1=Integer.parseInt(jTextField1.getText());
    int num2=Integer.parseInt(jTextField2.getText());
jTextField3.setText(Integer.toString(Integer.parseInt
(jTextField1.getText())+Integer.parseInt(jTextField2.getText())));
}
// Substraction
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt)
{
    int num1=Integer.parseInt(jTextField1.getText());
    int num2=Integer.parseInt(jTextField2.getText());
jTextField3.setText(Integer.toString(Integer.parseInt
(jTextField1.getText())-Integer.parseInt(jTextField2.getText())));
}
// Multiply
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt)
{
     int num1=Integer.parseInt(jTextField1.getText());
     int num2=Integer.parseInt(jTextField2.getText());
jTextField3.setText(Integer.toString(Integer.parseInt
(jTextField1.getText())*Integer.parseInt(jTextField2.getText())));
}
// Division
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt)
{
     int num1=Integer.parseInt(jTextField1.getText());
     int num2=Integer.parseInt(jTextField2.getText());
jTextField3.setText(Integer.toString(Integer.parseInt
(jTextField1.getText())/Integer.parseInt(jTextField2.getText())));
}
//Clear
jTextField1.setText(null);
jTextField2.setText(null);
jTextField3.setText(null);
```