

JAVA BASICS - SYNTAX AND PROGRAMS (J2SDK 5.0)

1. COMMENTS

```
package com.java.syntax;
/*
 * Class Name :Comments.java
 * Date :Mar 27, 2007
 * Author :Antony
 * Here is the Block Comment
 */

public class Comments
{
    /*
     * In java basically two types of comments are available
     * i) Single line comment
     * ii) Block comment
     */
    public static void main(String[] args)
    {
        //This is the Single line Comment
        System.out.println("Java is a free form " +
                           "Language and case sensitive");
    }
}
```

2. DATA TYPES

```
package com.java.syntax;
/*
 * Class Name :DataType.java
 * Date :Mar 27, 2007
 * Author :Antony
 */

public class DataType
{
    /*
     * There are eight primitive data types in java
     * Four Integer datatypes
     * Two Floating-point datatypes
     * One Character type
     * One Booleantype
     */
    public static void main(String[] args)
    {

        //Java Integer Datatypes
        byte b =20; // used to store upto 1 byte (-127 to 127)
        int i =1090; // used to store upto 4 bytes(-2,147,483,648
                    // to 2,147,483, 647 (just over 2 billion)
        short s =1234; // used to store upto 2 bytes (-32,768 to
```

```

        // 32,767)
    long l =12345678; // used to store upto 8 bytes
    //(-9,223,372,036,854,775,808L to 9,223,372,036,854,775,807L)

    // Java Floating-Point DataTypes
    float f =20.5f; // used to store upto 4 bytes
    // (approximately 3.40282347E+38F)
    double d =123.23456; // used to store upto 8 bytes
    // (approximately 1.79769313486231570E+308

    //Character type
    char c ='a'; //2 bytes of single character

    //Boolean Type
    boolean flag =true; // one byte 0 or 1

    System.out.println("The Byte Value is :"+b);
    System.out.println("The Integer Value is :"+i);
    System.out.println("The Character Value is :"+c);
    System.out.println("The Float Value is :"+f);
    System.out.println("The Double Value is :"+d);
    System.out.println("The Long Value is :"+l);
    System.out.println("The Boolean Value is :"+flag);
    System.out.println("The Short Value is :"+s);

}
}

```

3. VARIABLES

```

package com.java.syntax;
/*
 * Class Name :Variables.java
 * Date :Mar 27, 2007
 * Author :Antony
 */

public class Variables
{

    /*
    * In Java every variables has type.You can declare or initialize
    * variable in a java program.There are two types of variable in
    * java
    * i) Primitive Type Variables
    * ii) Object Reference Variables
    */
    public void display()
    {
        System.out.println("Be a compiler when you develop a program");
    }

    /*

```

```

* Convention for naming the variable
* i) A variable name must begin with a letter.
* ii) It must be a sequence of letters or digits.
* iii)It can be any of the lenth
*/

public static void main(String[] args)
{
//Primitive Type variable Declaration
    int sampleInt;
    double sampleDouble1;

//Primitive Type variable initialization
    double d =10.12324;
    char sample_char ='a';
    int s =10;

//Object Reference Variable-> v
    Variables var = new Variables();
    var.display();

}
}

```

4. CONSTANTS

```

package com.java.syntax;
/*
 * Class Name :Contants.java
 * Date :Mar 27, 2007
 * Author :Antony
 */

public class Contants
{

    /*
     * Constant is one type of variable that value never be changed
     * in a full java program.using final key word we can declare
     * a constant variable
     */
    public static void main(String[] args)
    {
        final double PI =3.14; // float constant
        System.out.println(PI);
    }

}
}

```

5. OPERATORS

```
package com.java.syntax;
/*
 * Class Name :Operators.java
 * Date :Mar 27, 2007
 * Author :Antony
 */

public class Operators
{
    /*
    * The usual arithmetic operators + * / are used in Java for
    * addition, subtraction, multiplication, and division. The /
    * operator denotes integer division if both arguments are
    * integers, and floating-point division otherwise. Integer
    * remainder (that is, the mod function) is denoted by %. For
    * example, 15 / 2 is 7, 15 % 2 is 1, and 15.0 / 2 is 7.5.
    */
    public void arithmeticOperator(int x,int y)
    {
        int addition = x + y;
        int subtraction = x - y;
        int multiplication = x * y;
        int division = x / y;
        int modulo = x % y;

        System.out.println("The Addition is :"+addition);
        System.out.println("The Subtraction is :"+subtraction);
        System.out.println("The Multiplication is
                            :"+multiplication);
        System.out.println("The Divison is :"+division);
        System.out.println("The Reminder is :"+modulo);

    }

    /*
    * The Shorthand assignment operator used to reduce the
    * expression size. for example
    * x =x+10 is equal to x+=10
    */
    public void shortHandOperator(int x,int y)
    {
        x += 5;
        y += 10;
        x *= 2;
        y /= 2;

        System.out.println("The result of x value is :"+x);
        System.out.println("The result of y value is :"+y);

    }

    /*
    * The value will be incremented or decremented by 1 using
    * Increment and Decrement Operators,There are two types
    */
}
```

```

*   i) Post / Pre Increment
*   ii) Post / Pre Decrement
*/
public void incrAndDecrOperator(int x,int y)
{
    x++; // Post increment
    --x; // Pre decrement
    y++;
    --y;

    System.out.println("The result of x value is :"+x);
    System.out.println("The result of y value is :"+y);
}

/*
* Relational Operator is used for checking the condition in
* the control statements .This operators are always
* return boolean value(true or false)
*/
public void relationalOperator(int x,int y)
{
    if(x==y)
        System.out.println("x and y are equal");
    else if(x>y)
        System.out.println("x is greater than y");
    else
        System.out.println("x is less than y");
}

/*
*Logical Operator is used for checking one or more conditions in
*the control statements .This operators are always
*return boolean value(true or false)
*/
public void logicalOperator(int x,int y)
{
    if(x==10 && y==5)
        System.out.println("And Condition true");
    else if(x==10 || y==5)
        System.out.println("Or Condition true");
    else if(!(x==10))
        System.out.println("Not Condition true");
}

public static void main(String[] args)
{
    Operators op = new Operators();
    op.arithmeticOperator(20,10);
    op.incrAndDecrOperator(20,30);
    op.logicalOperator(10,5);
    op.relationalOperator(35,40);
    op.shortHandOperator(5,2);
}
}

```

6. MATHEMATICAL FUNCTIONS

```
package com.java.syntax;
/*
 * Class Name :MathFunctions.java
 * Date :Mar 27, 2007
 * Author :Antony
 */
import static java.lang.Math.*;
public class MathFunctions
{
    /*
    * java.lang.Math has lot of in built mathematical functions
    * Math class is static so you can directly access all functions
    * using its class name or you can directly access method using
    * static import (since jdk1.5)
    */
    public static void main(String[] args)
    {
        double x =100;

        // Returns the correctly rounded positive square root of a
        // double value
        System.out.println("The Square Root Value is :"+sqrt(x));

        // Returns the base 10 logarithm of a double value
        System.out.println("The Log Value is :"+log10(x));

        /*
        * Returns the value of the first argument raised
        * to the power of the second argument
        */
        System.out.println("The Power Value is :"+Math.pow(x, 2));
    }
}
```

7. READING INPUT

```
package com.java.syntax;
/*
 * Class Name :ReadingInput.java
 * Date :Mar 27, 2007
 * Author :Antony
 */
import java.util.Scanner;
import javax.swing.JOptionPane;
public class ReadingInput
{
    /*
    * In java two types of Input methos are available
    * i) Using Scanner (Since jdk1.5)
    * ii) Using showInputDialog()
    */
    public static void main(String[] args)
    {
        //This code allow user to read String from a System.in
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter message:");
        String msg = sc.nextLine();
        System.out.println("Your message is :"+msg);

        //Shows a question-message dialog requesting input from the user.

        String msg2 =JOptionPane.showInputDialog("Enter Message");
        JOptionPane.showMessageDialog(null,"Your message is
                                           :"+msg2);

    }
}
```

8. STRING CLASS

```
package com.java.syntax;
/*
 * Class Name :StringClass.java
 * Date :Mar 27, 2007
 * Author :Antony
 */
public class StringClass
{
    /*
    * Strings are sequences of characters, such as "Hello".
    * Java does not have a built-in string type. Instead,
    * the standard Java library contains a predefined class called,
    * naturally enough, String.
    */
}
```

```

public static void main(String[] args)
{
//String Initialization
String str1 ="Java is a";
String str2 = " Fully Object Oriented";

/*
 * Compares this string to the specified object.
 * The result is true if and only if the argument is not
 * null and is a String object that represents the same
 * sequence of characters as this object.
 */
if(str1.equals(str2))
{
System.out.println("Two strings are equal");
}

//Concatenates the specified string to the end of this string.
System.out.println("The Contenation is :"+str1.concat(str2));
}
}

```

9. CONDITIONAL STATEMENTS

i) If Condition

```

package com.java.syntax;

import java.util.Scanner;
/*
 * Class Name :IfCondition.java
 * Date :Mar 27, 2007
 * Author :Antony
 */

public class IfCondition
{

public static void main(String[] args)
{
Scanner sc = new Scanner(System.in);
System.out.println("Enter the Number:");
int num = sc.nextInt();
if(num >10)
{
System.out.println("The Number is Greater than 10");
}
}
}
}

```

ii) if – else condition

```
package com.java.syntax;

import java.util.Scanner;
/*
 * Class Name :IfElseCondition.java
 * Date :Mar 27, 2007
 * Author :Antony
 */

public class IfElseCondition
{

    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the Number:");
        int num = sc.nextInt();

        if(num >10)
        {
            System.out.println("The Number is Greater than 10");
        }
        else
        {
            System.out.println("The Number is less than 10");
        }
    }
}
```

iii) Nested If-else Condition

```
package com.java.syntax;

import java.util.Scanner;
/*
 * Class Name :NestedIfElse.java
 * Date :Mar 27, 2007
 * Author :Antony
 */

public class NestedIfElse
{

    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the Mark:");
        int num = sc.nextInt();

        if(num >=90)
```

```

        {
            System.out.println("S Grade");
        }
        else if(num >=80)
        {
            System.out.println("A Grade");
        }
        else if(num >=70)
        {
            System.out.println("B Grade");
        }
        else if(num >=60)
        {
            System.out.println("C Grade");
        }
        else if(num >=55 && num <=59)
        {
            System.out.println("D Grade");
        }
        else if(num >=52 && num <=54)
        {
            System.out.println("E Grade");
        }
        else
        {
            System.out.println("Fail");
        }
    }
}

```

iv) Switch Statement

```

package com.java.syntax;

import java.util.Scanner;
/*
 * Class Name :SwithCase.java
 * Date :Mar 27, 2007
 * Author :Antony
 */

public class SwithCase
{

    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("1.Coke \n2.Fanta \n3.Miranda
                            \n4.Pepsi");
        System.out.println("Enter Your Choice:");
        int num = sc.nextInt();

        switch(num)
        {

```

```

        case 1:
            System.out.println("You have selected Coke");
            break;

        case 2:
            System.out.println("You have selected Fanta");
            break;

        case 3:
            System.out.println("You have selected Miranda");
            break;

        case 4:
            System.out.println("You have selected Pepsi");
            break;

        default :
            System.out.println("Select Valid Item");

    }

}
}

```

10 . BRABCHING AND LOOPING

i) While Loop

```

package com.java.syntax;
/*
 * Class Name :WhileLoop.java
 * Date :Mar 27, 2007
 * Author :Antony
 */

public class WhileLoop
{

    public static void main(String[] args)
    {
        int i=1;
        while(i<=10)
        {
            System.out.println("Count is :"+i);
            i++;
        }

    }

}
}

```

ii) do-while loop

```
package com.java.syntax;
/*
 * Class Name :doWhileLoop.java
 * Date :Mar 27, 2007
 * Author :Antony
 */

public class DoWhileLoop
{
    public static void main(String[] args)
    {
        int i=1;
        do
        {
            System.out.println("Count is :"+i);
            i++;
        }while(i<=10);
    }
}
```

iii) for loop

```
package com.java.syntax;
/*
 * Class Name :ForLoop.java
 * Date :Mar 27, 2007
 * Author :Antony
 */

public class ForLoop
{
    public static void main(String[] args)
    {
        for(int i=0;i<=10;i++)
        {
            System.out.println("Count is :"+i);
        }
    }
}
```

iv) Enhanced for loop(Since jdk1.5)

```
package com.java.syntax;
/*
 * Class Name :EnhancedForLoop.java
 * Date :Mar 27, 2007
 * Author :Antony
 */

public class EnhancedForLoop
{

    public static void main(String[] args)
    {
        String str[] = {"antony","raj","ligory","evanjelin"};
        int a[] = {10,11,12,13,14,15};

        System.out.println("The String Array values are:");
        for(String x : str)
        {
            System.out.println(x);
        }

        System.out.println("The Integer Array values are:");
        for(int y : a)
        {
            System.out.println(y);
        }
    }
}
```

OUTPUT

```
The String Array values are:
    antony
    raj
    ligory
    evanjelin
The Integer Array values are:
    10
    11
    12
    13
    14
    15
```

11. CLASS AND OBJECTS

```
package com.java.syntax;
/*
 * Class Name :SimpleClass.java
 * Date :Mar 27, 2007
 * Author :Antony
 */

public class SimpleClass
{

    //Variable Declarations
    int x;
    String str;

    //Method Declaration
    public void display(int x, String s)
    {
        System.out.println("The Integer Value is "+x);
        System.out.println("The String Value is "+s);
    }

    // Main Method
    public static void main(String[] args)
    {
        SimpleClass s =new SimpleClass();
        s.x =10;
        s.str ="Java";
        s.display(s.x, s.str);
    }
}
```

12 .INNER CLASS

```
package com.java.syntax;
/*
 * Class Name :InnerClass.java
 * Date :Mar 27, 2007
 * Author :Antony
 */

public class InnerClass
{
    class Sample
    {
        void display()
        {
            System.out.println("This is Inner Class Methos");
        }
    }

    public static void main(String[] args)
    {
```

```

        InnerClass i = new InnerClass();
        i.new Sample().display();
    }
}

```

13 .STATIC INNER CLASS

```

package com.java.syntax;

/*
 * Class Name :InnerClass.java
 * Date :Mar 27, 2007
 * Author :Antony
 */

public class StaticInnerClass
{
    static class Sample
    {
        static void display()
        {
            System.out.println("This is Inner Class Methos");
        }
    }

    public static void main(String[] args)
    {
        Sample.display();
    }
}

```

14 .INTERFACE

```

package com.java.syntax;

/*
 * Class Name :SampleInterface.java
 * Date :Mar 27, 2007
 * Author :Antony
 */

public interface SampleInterface
{
    int x=10;
    String str="Antony";

    //Interface method cannot have body
    public void getData();
}

```

15 .ARRAY

```

package com.java.syntax;
/**
 * @author Antony
 * Date : 23/12/06
 */
public class ArrayTest
{

    public static void main(String[] args)
    {
        /*
        * Array Declaration
        * You can use the following styntax for declaring array
        *         int[] a
        *     or
        *         int a[]
        *         or
        *         int[]a
        *         or
        *         int []a
        */
        int a[];

        // Various type of array initialization;
        a=new int[10];
        int[] b =new int[11];
        Object a1[][] = new Object[10][10];

        a1[0][0] ="Antony raj";
        System.out.println(a1[0][0]);
        int []c = {1,2,3,4,5,6,7,8,9,10};
        int[]d=new int[] {12,42,53,64,35,66};//Anonymous array

        //Array Copying
        a=d;
        b=c;

        System.out.println("The Values of Array a");
        for (int i=0;i<a.length;i++)
        {
            System.out.println(a[i]);
        }

        System.out.println("The Values of Array b");

        //for each loop since jdk1.5
        for(int x : b)
        {
            System.out.println(x);
        }

    }

}

```

