Tutorial 1 : Language Fundamentals

* Identifiers
* Reserve Words
* Data Types
* Literals
* Arrays
* Types of Variables
* Var-args Methods
* Main Methods
* Command Line Arguments
* Java Coding Standards

**IDENTIFIERS**

A name in java programming is called Identifier, which can be used for identification purposes. It can be method name or variable name, class name or label name

class Test

{

public static void main (String [] args)

{ int x;

// System.out.print( “ Hello ”);

}

}

Following are the identifiers in above program

* Test – class name
* Main - method name
* String - class name
* args – variable name (name of array)
* X - variable name

Rules for defining java identifiers:

* a to z
* A to Z
* 0 to 9
* $
* \_

Only allowed characters in java identifiers are above

**Rules**:- for identifiers

* If we are using any other character , we will get compile time error

Total\_number – allowed

Total# - not allowed

* Identifiers can’t starts with numerical values

Total123 – valid

123Total – invalid

* Java identifiers are **case sensitive** of course java languages itself is treated as case sensitive

Programming language

class Test

{

int number = 10;

int Number = 20;

int NUMBER = 30;

}

In this case we can differentiate with respect to case.

* In Java , there is no length limit for java identifiers, but it is not recommended to take too lengthy identifiers

Class Test

{

public static void main (String [] args)

{ int xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx = 10;

// System.out.print( “ Hello ”);

System.out.print(xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx);

}

}

This code is a valid code as java doesn’t have any limit on the length of identifiers,

But it is not a good practice to use a lengthy identifiers.

* We can’t use reserved words as identifiers

Int x = 10 - valid;

Int if = 10 – invalid; (because if is reserved words)

We can’t use reserve words as identifiers

* We can use already defined java class name or interface name as identifiers.

class Test{

public static void main (String [] args)

{

int String = 888;

int Runnable = 999;

System.out.println(String);

System.out.println(Runnable);

}

}

* All predefined class name and interface name we can use as a identifiers, but it is not a good practice to use it.
* Eventhough it is valid but it is not a good programming practice , because it reduces readability and creates confusion

Which of the following are valid java identifiers

* Total\_number - valid
* Total# - not valid
* 123total - not valid
* Total123 - valid
* ca$h - valid
* \_$\_$\_$\_$\_ - valid
* all@hands -not valid
* java2share -valid
* Integer - valid – class name
* Int - valid – not a reserve words
* Int - not valid - reserve words

In java some words are reserved words to represent some meaning or functionality such types of words are called reserved words (53) keywords are in java

<Goto - 35.12 >

Reserved Words (53) = Keywords (50) + Reserved Literals (3)

Keywords (50) – if the reserved word is assoicated with functionality

Reserved Literals (3) - [true, false, null]

Keyword(50) = used keywords(48) + nounsed keywords(2)

Keywords for dataTypes (8) - bit, short, int, long, float, double, boolean, char

Keywords for flow control (11)- if-else. switch (case, default ), while , do-while, for, break, continue, return

Keywords for modifiers (11) - public, private, protected, static, final, abstract, synchronized, native, strictfp, transient, volatile

keywords for exception handling (6)- try-catch, finally, throw, throws, assert

Keyword related to class (6)- class, interface, extends, implements, package, imports

Object related keywords - new , instanceof, super, this,