Java keyword

Lingyan Xie

2/24/2022

For data types

* byte, short, int, long, float, double, char, boolean
* byte: byte is a keyword which designates the 8-bit signed integer primitive type.
* short: The short keyword is used to designate a field that can hold a 16-bit signed two's complement integer.
* int: The int keyword is used to designate a variable that can hold a 32-bit signed two's complement integer.
* long: long is a keyword which designates the 64-bit signed integer primitive type.
* float: The float keyword is a data type that can store fractional numbers from 3.4e−038 to 3.4e+038.
* double: The double keyword is used to designate a variable that can hold a 64-bit double precision IEEE 754 floating-point number
* char: char is a keyword that is used to declare a variable which store a character value from the range of +U0000 to U+FFFF.
* boolean: The boolean keyword is a data type that can only take the values true or false

flow control

* if, else, switch, case, default, for, do, while, break, continue, return
* if: The if statement specifies a block of Java code to be executed if a condition is true
* else: The else statement specifies a block of Java code to be executed if a condition is false in an if statement.
* switch: The switch keyword selects one of many code blocks to be executed
* case: The Java case keyword is a conditional label which is used with the switch statement.
* default: a) access modifier. b) default label in switch statement c) declare default methods in interface
* for: The for loop loops through a block of code a number of times.
* while: The while loop loops through a block of code as long as a specified condition is true.
* break: The break keyword is used to break out a for loop, a while loop or a switch block.
* continue: The continue keyword is used to end the current iteration in a for loop (or a while loop), and continues to the next iteration.
* return: The return keyword finished the execution of a method, and can be used to return a value from a method.

modifiers

* public, private, protected, static, final, abstract, synchronized, native, strictfp, transient, volatile
* public: The public keyword is an access modifier used for classes, attributes, methods and constructors, making them accessible by any other class.
* private: The private keyword is an access modifier used for attributes, methods and constructors, making them only accessible within the declared class.
* protected: The protected keyword is an access modifier used for attributes, methods and constructors, making them accessible in the same package and subclasses.
* static: The static keyword is a non-access modifier used for methods and attributes. Static methods/attributes can be accessed without creating an object of a class.
* final: The final keyword is a non-access modifier used for classes, attributes and methods, which makes them non-changeable (impossible to inherit or override).
* **abstract:** The abstract keyword is a non-access modifier, used for classes and methods.

**For class: An abstract class is a restricted class that cannot be used to create objects (to access it, it must be inherited from another class).**

**For method: An abstract method can only be used in an abstract class, and it does not have a body. The body is provided by the subclass (inherited from).**

* **synchronized: A piece of logic marked with synchronized becomes a synchronized block, allowing only one thread to execute at any given time.**
* **native**: The native keyword in Java is applied to a method to indicate that the method is implemented in native code using JNI (Java Native Interface). The native keyword is a modifier that is applicable only for methods, and we can't apply it anywhere else.
* **strictfp: strictfp is used to ensure that floating points operations give the same result on any platform.**
* **transient: The transient keyword in Java is used to avoid serialization. If any object of a data structure is defined as a transient, then it will not be serialized. Serialization is the ​process of converting an object into a byte stream.**
* **volatile: The Java volatile keyword is used to mark a Java variable as "being stored in main memory", making it thread safe**

exception handling

* try, catch, finally, throw, throws, assert
* try: The try keyword creates a try...catch statement.

The try statement allows you to define a block of code to be tested for errors while it is being executed.

* catch: The catch statement allows you to define a block of code to be executed, if an error occurs in the try block.
* finally: The finally keyword is used to execute code (used with exceptions - try..catch statements) no matter if there is an exception or not.
* **throw: Used to throw an exception for a method**
* **throws: Used to indicate what exception type may be thrown by a method**
* assert: assert is a Java keyword used to define an assert statement. An assert statement is used to declare an expected boolean condition in a program.

class related

* class, package, import, extends, implements, interface
* class: The class keyword is used to create a class.
* package: The package keyword creates a package.
* import: The import keyword is used to import a package, class or interface.
* extends: The extends keyword extends a class (indicates that a class is inherited from another class).
* implements: The implements keyword is used to implement an interface.
* interface: The interface keyword is used to declare a special type of class that only contains abstract methods.

Object related keywords,

* new, instanceof, super, this
* new: The new keyword creates new objects.
* instanceof: The instanceof keyword checks whether an object is an instance of a specific class or an interface.
* super: The super keyword refers to superclass (parent) objects. It is used to call superclass methods, and to access the superclass constructor.
* this: The this keyword refers to the current object in a method or constructor.